

BAB III

PELAKSANAAN KERJA

3.1 Kedudukan dan Koordinasi

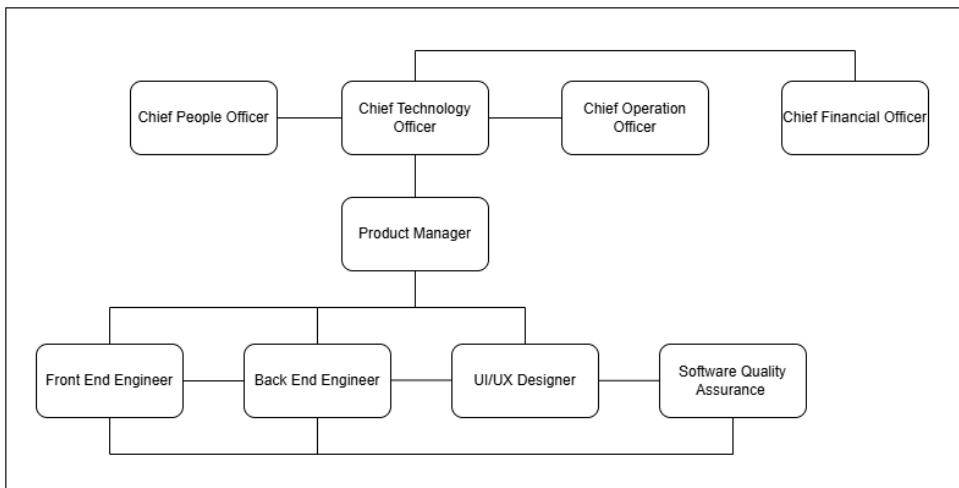
Bagian ini memuat keterangan dan informasi mengenai posisi yang ditempati selama periode praktik kerja serta alur koordinasi yang diterapkan. Dijelaskan pula bagaimana mekanisme koordinasi dilaksanakan dengan pembimbing lapangan pada saat pengerjaan proyek atau penugasan tertentu, termasuk struktur pelaporan dan interaksi tim yang dilakukan.

3.1.1 Kedudukan

Dalam *role* sebagai QA *Intern*, kedudukan yang dimiliki adalah setara dengan rekan-rekan *engineer* dan *designer* lainnya. Hal ini disebabkan tidak adanya senior atau manajer dengan *role* QA yang sama secara langsung dalam tim, sehingga QA *Intern* merupakan satu-satunya yang bertugas. Kondisi ini menuntut tingkat kemandirian yang tinggi dan tanggung jawab penuh terhadap keseluruhan proses pengujian dan kualitas produk. Oleh karena itu, koordinasi dan interaksi dilakukan secara langsung dengan berbagai *role* fungsional lain, seperti *Back-End Engineer*, *Front-End Engineer*, dan *UI/UX Designer*.

3.1.2 Koordinasi

Bagian ini memuat penjelasan terperinci mengenai informasi alur koordinasi pekerjaan yang dilaksanakan selama periode praktik kerja di perusahaan. Dijelaskan pula bagaimana mekanisme alur kerja tersebut divalidasi dan diikuti, yang dapat dilengkapi dengan bagan alur kerja visual untuk memberikan gambaran yang lebih jelas mengenai interaksi dan proses yang dilakukan:



Gambar 3. 1 Alur Koordinasi

Dalam *role* sebagai QA, kedudukan yang dimiliki adalah setara dengan rekan-rekan *engineer* dan *designer* lainnya seperti pada gambar 3.1. Hal ini disebabkan tidak terdapat senior atau manajer dengan *role* QA yang sama secara langsung dalam tim, sehingga QA merupakan satu-satunya yang bertugas. Kondisi ini menuntut tingkat kemandirian yang tinggi dan tanggung jawab penuh terhadap keseluruhan proses pengujian dan kualitas produk.

Struktur dan alur koordinasi pekerjaan dimulai dari tingkat kepemimpinan eksekutif. Kebutuhan spesifik divisi, seperti yang diajukan oleh CPO, COO, dan CFO, dilaporkan kepada CTO. Kebutuhan tersebut kemudian diturunkan kepada *Product Manager* untuk dianalisis dan diterjemahkan menjadi spesifikasi fitur dalam aplikasi. Setelah spesifikasi disepakati, pengembangan dilanjutkan oleh *UI/UX Designer*, *Back-End Engineer*, dan *Front-End Engineer*. *Product Manager* bertanggung jawab untuk memberikan penjelasan komprehensif mengenai keseluruhan konsep produk dan *system flow*. Penjelasan ini kemudian dijadikan dasar utama bagi penyusunan perencanaan pengujian yang akan dilaksanakan.

Apabila terdapat alur (*flow*) yang dinilai cukup rumit, konfirmasi dan pertanyaan dilakukan kepada *UI/UX Designer* untuk meminimalisir salah pemahaman sebelum pengujian dimulai. Selain itu, apabila selama proses pengujian teridentifikasi adanya alur yang janggal, tidak sesuai ekspektasi,

atau membingungkan, diskusi terlebih dahulu dilakukan dengan *Designer*. Tujuan dari langkah ini adalah untuk memverifikasi alur yang seharusnya sesuai dengan spesifikasi desain dan kebutuhan pengguna.

Pelaporan *bug* teknis diarahkan secara langsung kepada tim *Engineer* yang relevan. Jika pengujian menemukan *bug* terkait tampilan (*user interface*) atau fungsionalitas sisi klien, seperti elemen yang tidak berfungsi atau tata letak yang salah, pelaporan ditujukan kepada *Front End Engineer*. Sebaliknya, apabila terdeteksi *bug* pada sisi *server*, seperti *error* kode 500 atau masalah pada integritas data dan logika bisnis, pelaporan diarahkan kepada *Back End Engineer*. Setelah diperoleh konfirmasi alur yang benar dari *Designer* terkait alur yang janggal, pelaporan *bug* atau permintaan perubahan baru akan dilanjutkan kepada *Front End Engineer* untuk implementasi perbaikan.

3.2 Tugas yang Dilakukan

Selama program kerja magang, terdapat beberapa proyek yang dikerjakan, tabel 3.1 merupakan contoh proyek beserta keterangan yang dilakukan dari awal hingga akhir.

Tabel 3. 1 Contoh Proyek Setiap Minggu

No.	Proyek	Keterangan	Minggu	Tanggal Mulai Aktivitas	Tanggal Akhir Aktivitas
1	Perkenalan	Perkenalan Lingkungan Kerja	1	23 Juni 2025	27 Juni 2025
2	Fibrina	Pemahaman produk Fibrina.	2	30 Juni 2025	4 Juli 2025
3	Fibrina	Memperbarui dokumentasi <i>flow</i> sistem dan membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Request Expense</i> .	3	7 Juli 2025	11 Juli 2025

No.	Proyek	Keterangan	Minggu	Tanggal Mulai Aktivitas	Tanggal Akhir Aktivitas
4	Fibrina	Membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Approval</i> dan <i>Rejection Expense</i> .	4	14 Juli 2025	18 Juli 2025
5	Fibrina	Membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Payment</i> .	5 - 6	21 Juli 2025	1 Agustus 2025
6	Fibrina	Pelaksanaan <i>Testing</i> manual modul <i>Request</i> dan <i>Approval Expense</i> .	7 – 8	4 Agustus 2025	15 Agustus 2025
7	Fibrina	Pelaksanaan <i>Testing</i> manual modul <i>Payment</i> dan <i>Automation Testing</i> .	9 - 10	18 Agustus 2025	29 Agustus 2025
8	Alpha	Pemahaman produk Alpha, membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Request</i> dan <i>Approve RO, PO, dan DO</i> .	11 – 12	1 September 2025	12 September 2025
9	Alpha	Membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>GR</i> dan <i>Payment</i> .	13	15 September 2025	19 September 2025
10	Alpha	Pelaksanaan <i>Testing</i> modul <i>Request</i> dan <i>Approve RO, PO, dan DO</i> .	14 - 15	22 September 2025	3 Oktober 2025
11	Alpha	Pelaksanaan <i>Testing</i> modul <i>GR</i> dan <i>Payment</i> .	16 - 17	6 Oktober 2025	22 Oktober 2025

3.3 Uraian Pelaksanaan Kerja

Sub-bab ini menyajikan uraian rinci mengenai pelaksanaan kegiatan magang dalam *role QA Intern* selama periode penugasan. Pembahasan mencakup keseluruhan proses, mulai dari pengembangan keterampilan teknis dan non-teknis terkait QA, kolaborasi tim, kontribusi pada pengembangan dan optimalisasi sistem yang diuji, hingga identifikasi kendala yang muncul beserta solusi yang diterapkan. Uraian yang disajikan dilengkapi dengan detail mengenai *role* dan tanggung jawab yang diemban, metode atau alat pengujian yang digunakan, serta hasil konkret yang dicapai pada setiap bagian.

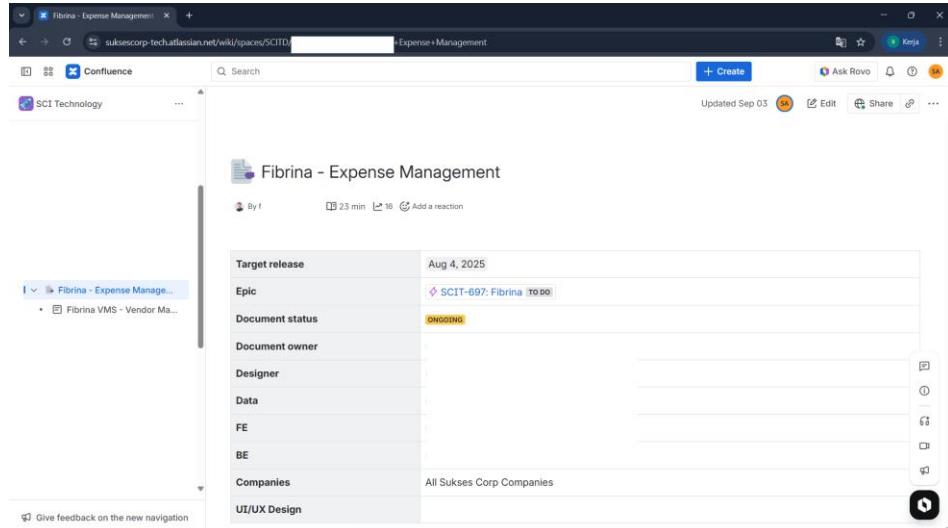
3.3.1 Proses Pelaksanaan

Pada proses ini menjelaskan secara rinci yang telah dilakukan selama kerja magang dari proses awal hingga akhir yang memberikan hasil yang relevan dengan *jobdesc* dan tanggung jawab sebagai *QA Intern* sebagaimana yang dijelaskan pada *point* terinci di bawah ini.

3.3.1.1 Perkenalan Lingkungan Kerja

Hari pertama memasuki program kerja magang dilakukan *onboarding* yang dibawakan oleh HR. Seluruh peserta magang yang tergabung pada hari itu dikumpulkan di salah satu ruang *meeting* yang bernama Ruang Sintia. Pada kesempatan ini, terdapat delapan peserta magang yang terbagi menjadi dalam masing-masing divisi mulai dari divisi *Sourcing, HR employee, Finance, Market Research, Legal* dan *QA*. Kegiatan *onboarding* menjadi pintu awal masing-masing peserta magang saling mengenal rekan kerja untuk beberapa bulan kedepan. Sesi ini tidak hanya diisi dengan perkenalan perusahaan, tetapi juga perkenalan masing-masing individu serta *mini games*. Setelah seluruh rangkaian telah dilaksanakan, kemudian peserta magang di ajak *office tour* oleh HR, mulai dari lantai 1 – 7. Setelah itu, masing-masing peserta akan kembali ke meja kerjanya untuk perkenalan lebih lanjut dengan user.

3.3.1.2 Pemahaman produk Fibrina.



The screenshot shows a Confluence page titled 'Fibrina - Expense Management'. The page includes a table with the following data:

Target release	Aug 4, 2025
Epic	SCIT-697: Fibrina TODO
Document status	ONGOING
Document owner	
Designer	
Data	
FE	
BE	
Companies	All Sukses Corp Companies
UI/UX Design	

Gambar 3. 2 Confluence Fibrina

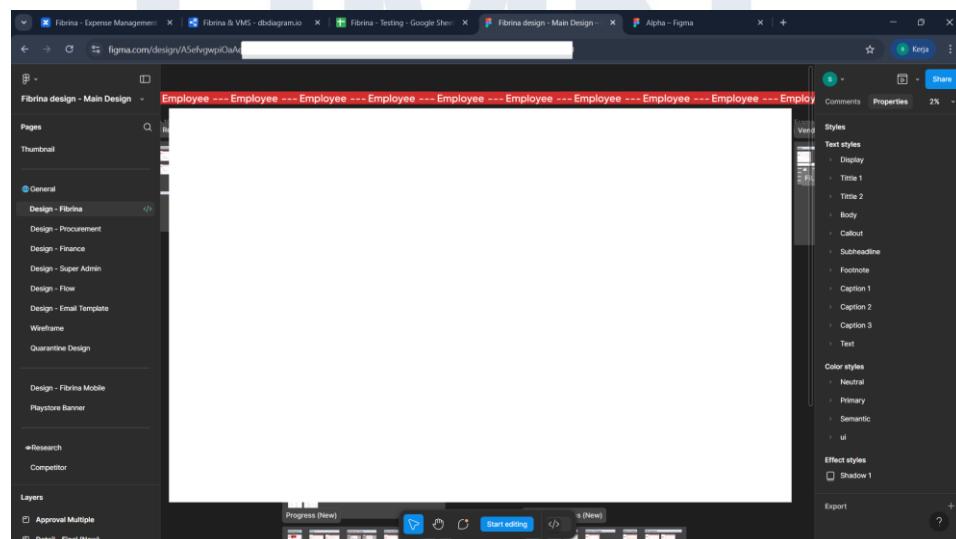
Pada minggu pertama pelaksanaan kegiatan magang, dilakukan upaya pemahaman yang komprehensif terhadap produk Fibrina, khususnya yang berkaitan dengan modul *Expense Management*. Pemahaman ini difokuskan pada analisis fungsionalitas sistem, alur kerja keseluruhan, dan lini masa rilis yang ditargetkan. Proses pemahaman mendalam ini wajib dilakukan sebagai langkah awal dan didasarkan pada tinjauan dokumentasi resmi yang telah tersedia dan tersimpan dalam *tools* Confluence pada gambar 3.2 perusahaan. Dokumentasi yang dipelajari berfungsi sebagai fondasi utama untuk seluruh aktivitas pengujian yang akan dilaksanakan kemudian.

Dokumentasi tersebut memuat *Product Overview* dan *General Requirements*, yang mencakup identifikasi tipe pengguna, aktivitas utama (*key activities*), serta perkiraan total angka pengguna sistem. Alur basis data aplikasi dijelaskan melalui *Entity-Relationship Diagram* (ERD) berdasarkan kebutuhan pengguna, dan alur antarmuka pengguna (*flow UI*) digambarkan melalui diagram

use case serta diagram alir. Selain itu, terdapat penjelasan rinci mengenai *Story Line* yang memetakan fitur dan menu spesifik yang dapat diakses, di mana akses ini disesuaikan berdasarkan perbedaan *role* atau jabatan. Kelengkapan dokumentasi ditunjukkan dengan adanya bagian *Expense Status* yang mendefinisikan arti dari setiap status pengeluaran, serta bagian *Expense Request Fields* yang merinci *fields* wajib pada halaman permintaan pengeluaran, disesuaikan setiap *role* yang ada.

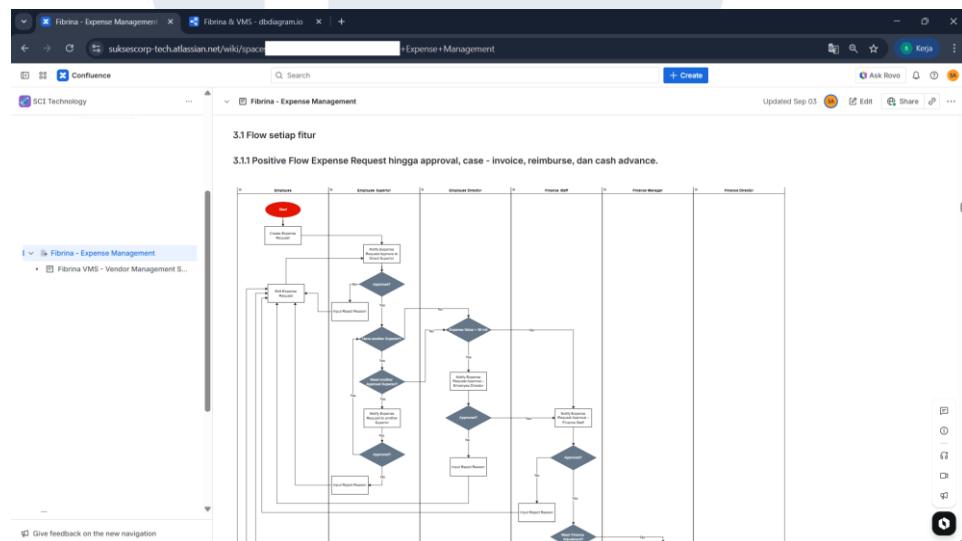
Pemahaman menyeluruh terhadap seluruh dokumentasi teknis dan fungsional yang tersedia ini menjadi landasan yang sangat penting bagi aktivitas *Software Quality Assurance*. Seluruh informasi tersebut digunakan sebagai acuan baku dalam penyusunan perencanaan pengujian, penentuan skenario, dan validasi fungsionalitas sistem Fibrina. Dengan fondasi pemahaman yang kuat, akurasi dan efektivitas pengujian di tahap-tahap selanjutnya dapat terjamin.

3.3.1.3 Memperbarui dokumentasi *flow* sistem dan membuat *Test Planning* dan *Skenario* modul *Request Expense*.



Gambar 3. 3 Figma Fibrina

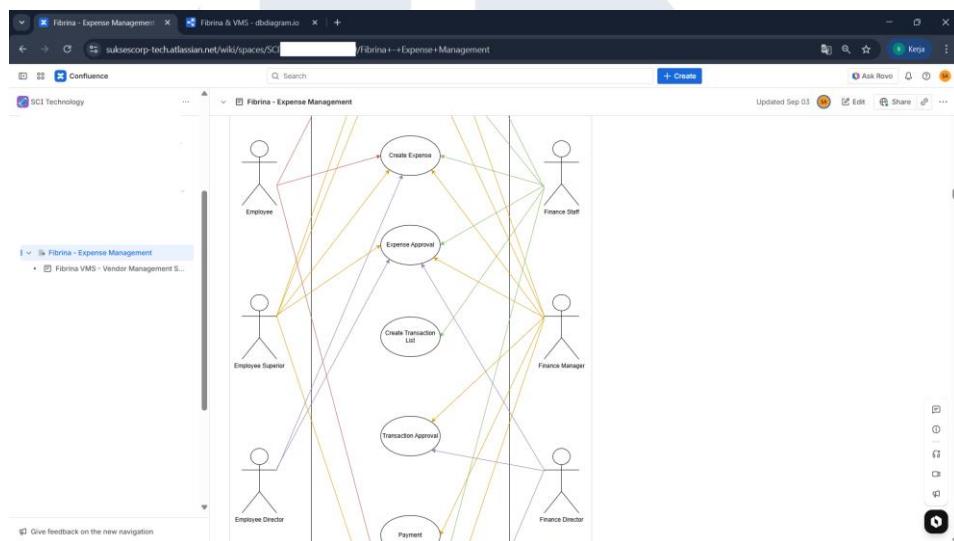
Pada minggu pertama pelaksanaan kegiatan magang, dilakukan upaya pemahaman yang komprehensif terhadap produk Fibrina, khususnya yang berkaitan dengan modul *Expense Management*. Pemahaman ini diawali dengan meninjau *design* antarmuka pengguna (*user interface*) secara terperinci melalui *tools* Figma seperti pada gambar 3.3, yang memuat seluruh rancangan yang telah dibuat oleh *UI/UX Designer*. Selanjutnya, pemahaman juga difokuskan pada analisis fungsionalitas sistem, alur kerja keseluruhan, dan lini masa rilis yang ditargetkan pada tanggal 4 Agustus 2025. Proses pemahaman mendalam ini didasarkan pada tinjauan dokumentasi resmi yang telah tersedia dan tersimpan dalam *tools* Confluence perusahaan.



Gambar 3.4 Diagram Flow Fibrina

Dokumentasi Confluence tersebut memuat *Product Overview* dan *General Requirements*, yang mencakup identifikasi tipe pengguna, aktivitas utama (*key activities*), serta perkiraan total angka pengguna sistem. Alur basis data aplikasi dijelaskan melalui *Entity-Relationship Diagram* (ERD) gambar 3.4 berdasarkan kebutuhan pengguna, dan alur design interface digambarkan melalui diagram use case serta diagram alur. Selain itu, terdapat penjelasan rinci mengenai

Story Line yang memetakan fitur dan menu spesifik yang dapat diakses, di mana akses ini disesuaikan berdasarkan perbedaan role atau jabatan. Kelengkapan dokumentasi ditunjukkan dengan adanya bagian *Expense Status* yang mendefinisikan arti dari setiap status pengeluaran, serta bagian *Expense Request Fields* yang merinci *fields* wajib pada halaman permintaan pengeluaran, disesuaikan setiap *role* yang ada.



Gambar 3. 5 Use Case Fibrina

Selain pembaruan *flow diagram*, pembuatan *use case* diagram juga dilaksanakan untuk mempermudah pemahaman mengenai fitur dan menu yang dapat diakses oleh setiap pengguna yang terlibat dalam sistem. Dalam aplikasi ini, sistem terdiri dari enam peran utama gambar 3.5, yaitu *Employee*, *Employee Superior*, *Employee Director*, *Finance Staff*, *Finance Manager*, dan *Finance Director*. Setiap aktor yang terlibat memiliki batasan fitur yang spesifik dalam sistem, dan diagram *use case* ini menjadi alat visual yang efektif untuk memetakan batasan akses tersebut. Dengan demikian, dapat dipastikan bahwa setiap peran hanya dapat mengakses fungsionalitas yang sesuai dengan tanggung jawabnya.

Batasan akses fitur yang spesifik diatur untuk setiap peran. Peran *Employee Staff* dibatasi hanya untuk dapat *request expense* dan mengajukan akses (*request access*). Peran *Employee Superior* dan *Employee Director* diberikan wewenang yang sama, yaitu dapat mengajukan *expense*, menyetujui (*approve*) dan menolak (*reject*) *expense*, serta menyetujui dan menolak permintaan akses. Peran *Finance Staff* memiliki wewenang untuk mengajukan *expense*, menyetujui dan menolak *expense*, serta membuat *transaction list*. Sementara itu, *Finance Manager* dan *Finance Director* memiliki wewenang yang serupa, yaitu dapat mengajukan *expense*, menyetujui dan menolak *expense*, serta menyetujui dan menolak *transaction list*.

Alur utama pemrosesan dimulai dari pembuatan *expense* oleh *Employee*, yang kemudian dilanjutkan ke tahap persetujuan oleh seluruh *approver* yang berwenang yaitu *Superior*, *Director*, *Finance Staff*, *Finance Manager*, dan *Finance Director*. Setelah disetujui, selanjutnya akan masuk ke tahap pembayaran dengan *transaction list* yang akan dibuat oleh *Finance Staff*, yang kemudian disetujui oleh *Finance Manager* dan *Finance Director*. Proses dilanjutkan dengan pengunggahan bukti pembayaran (*upload proof of payment*) oleh *Finance Staff* dan diakhiri dengan pengunggahan bukti penggunaan (*upload proof of usage*) oleh *Employee* sebagai *requestor*.

Fibria - Testing												
Test Plan - Fibria - 6												
ID	Name	Category	Attribute	Role	App	Model	SubModel	Test Scenario	Expected Result	Duration (Second)	Expected Result	Result
6	TP - Fibria - 6	Functional	Test	No	Superior	Web	Expense	Expense Approval	Expense Approval - Single	77 ms	TF - Fibria - 6.png	PASS
7	TP - Fibria - 7	Functional	Test	No	Director	Web	Expense	Expense Approval	Expense Approval - Single	44 ms	TF - Fibria - 7.png	PASS
8	TP - Fibria - 8	Functional	Test	No	Finance	Web	Expense	Expense Approval	Expense Approval - Single	96 ms	TF - Fibria - 8.png	PASS
9	TP - Fibria - 9	Functional	Test	No	Finance Manager	Web	Expense	Expense Approval	Expense Approval - Single	89 ms	TF - Fibria - 9.png	PASS
10	TP - Fibria - 10	Functional	Test	No	Finance Director	Web	Expense	Expense Approval	Expense Approval - Single	53 ms	TF - Fibria - 10.png	PASS

Gambar 3. 6 Potongan Lembar Kerja Test Planning

Setelah dilakukan pemahaman product, selanjutnya adalah pembuatan lembar kerja untuk *test planning*. Lembar kerja ini digunakan bukan hanya untuk dokumentasi QA dalam melakukan testing seperti gambar 3.6 contoh potongan lembar kerja *test planning*, tapi juga sebagai lembar yang dapat di akses oleh *frontend* dan *backend developer* untuk mengecheck apakah seluruh pengujian sudah menyeluruh atau terdapat bagian yang belum di jadikan planning.

Dalam penulisan *Test Planning*, sebanyak 14 kolom menjadi persyaratan utama yang harus dipenuhi oleh seorang *Quality Assurance*. Kolom pertama adalah *Naming*, yang berisi *ID* pengujian yang ditulis secara berurutan dengan format *TP – Fibrina – 1* mengacu pada *Test Planning* pertama untuk aplikasi Fibrina. Selanjutnya, terdapat kolom *Category*, yang berfungsi sebagai indikator pengujian dengan dua pilihan yaitu *Functional Test*, yang berfokus pada kegunaan setiap fitur dan alur dan *Regression Test*, yang merupakan pengujian ulang setelah adanya perbaikan temuan atau masalah. Kolom *Automate* digunakan untuk memberikan keterangan mengenai metode pengujian, apakah dibantu dengan alat otomasi atau dilakukan secara manual. Kolom *Role* menjelaskan peran pengguna yang menjalankan skenario , sementara kolom *Apps* memberikan penjelasan mengenai tipe aplikasi yang diuji, apakah berbasis *web* atau *mobile*.

Kolom *Modul* dan *Submodul* berfungsi untuk menjelaskan fungsi umum dan fungsi lebih rinci dari skenario yang dibuat. *Test Skenario* berisi tahapan yang harus dieksekusi dari awal hingga akhir, dan *Duration* mencatat hasil *end point* yang dikeluarkan oleh sistem untuk mengukur kecepatan *backend* dalam memproses data. Kolom *Expected Result* menjelaskan hasil yang seharusnya diberikan pada skenario tersebut, dan *Result* berisi hasil

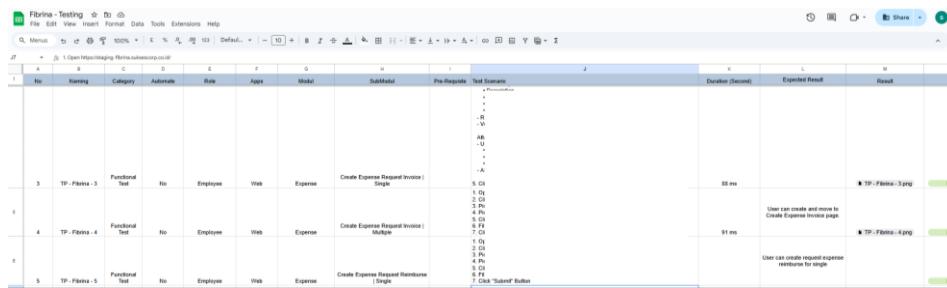
sesungguhnya dari skenario yang telah dilakukan. Kolom *Status* memberikan keterangan hasil pengujian dengan dua opsi *Pass* apabila pengujian berhasil dan *Not Pass* apabila pengujian gagal. Terakhir, kolom *PC Dev* dan *PC QA* mencatat nama *developer* yang mengerjakan dan *QA* yang melakukan pengujian.

	A	B	C	D	E	F	G	H	I	J	K	M
	Test Case ID	Modul	Sub Modul	Log	Log Type	Status	Raise Date	PIC Test	PIC Dev	Attachment	Dev Note	
6	TP - Fibrina - 06 - 01	Superior	Approve Expense Request	Unable to invoke	multiple lines not use date or page my times	Major	03/07/2025 - 3:45 PM	Sabrina Azmi	Done			
6	TP - Fibrina - 06 - 02	Superior	Approve Expense Request	Unable to invoke when I uses finance Emp	use date or page my times	Minor	03/07/2025 - 5:00 PM	Sabrina Azmi	Ready to Re-test			
7	TP - Fibrina - 07 - 01	Employee Director	Approve Expense Request	Unable to invoke when I uses finance Emp	use date or page my times	Minor	03/07/2025 - 5:15 PM	Sabrina Azmi	To Do			
8	TP - Fibrina - 10 - 01	Finance Director	Approve Expense Request	Unable to invoke when I uses finance Emp	use date or page my times	Minor	03/07/2025 - 5:35 PM	Sabrina Azmi	Pending			
9	TP - Fibrina - 11 - 01	Finance Staff	Transaction List	Date filter	any name	Minor	03/07/2025 - 5:45 PM	Sabrina Azmi	Done			

Gambar 3. 7 Potongan lembar kerja *Test Log*

Setelah *Test Planning* disusun, tahap selanjutnya adalah penulisan *Test Log* seperti gambar 3.7 contoh lembar kerja *Test Log*. Lembar kerja *Test Log* ini terdiri dari 12 kolom yang berfungsi untuk mendokumentasikan setiap temuan *bug*. Kolom pertama, *Test Case ID*, memiliki penamaan unik setiap barisnya, seperti *TP - Fibrina - 06 - 01*. Format penamaan ini menunjukkan bahwa permasalahan ditemukan dari *Test Planning* aplikasi *Fibrina*, berasal dari urutan skenario nomor 06, dengan dua angka terakhir mengindikasikan jumlah permasalahan yang muncul pada skenario tersebut. Kolom *Modul* dan *Submodul* digunakan untuk mengidentifikasi fungsi yang terpengaruh. Kolom *Log* berisi penjelasan rinci mengenai permasalahan atau *error* yang ditemukan. *Log Type* mengklasifikasikan tipe permasalahan, apakah termasuk dalam kategori minor (*non blocking*) atau permasalahan major (*blocking*). Kolom *Status* memiliki empat kategori yaitu *To Do*, yang berarti temuan telah dilaporkan oleh *QA*, *Pending*, yang mengindikasikan bahwa perbaikan *bug* harus ditunda karena adanya kepentingan lain dari *developer*, *Ready to re-test*, yang berarti *bug* sudah diperbaiki oleh *developer* dan siap untuk diuji kembali oleh *QA*, dan *Done*, yang mengartikan bahwa *bug* sudah selesai diuji kembali. *Raise Date* mencatat tanggal dilaporkannya *bug* oleh *QA*. Kolom *PIC QA*

mencantumkan penanggung jawab yang melaporkan *bug*, sementara *PIC Dev* menunjukkan kepada siapa *bug* harus diperbaiki, apakah *front end* atau *back end*. Terakhir, kolom *Attachment* digunakan untuk menyimpan bukti saat *QA* menemukan *bug* selama pengujian.



No	Flow	Category	Actions	Row	App	Module	SubModule	Pre-Request	Test Scenario			Duration (second)	Expected Result	Result
									1	2	3			
3	TP - Fibrina - 3	Functional Test	No	Employee	Web	Expense	Create Expense Request	Single	1. O1	2. O2	3. O3	60 ms	TP - Fibrina - 3.png	OK
4	TP - Fibrina - 4	Functional Test	No	Employee	Web	Expense	Create Expense Request	Multiple	1. O1	2. O2	3. O3	91 ms	User can create and review N Create Expense Invoice page	OK
5	TP - Fibrina - 5	Functional Test	No	Employee	Web	Expense	Create Expense Request Reimburse	Single	1. O1	2. O2	3. O3	91 ms	User can create request expense reimbursement for single	NG

Gambar 3. 8 Potongan Test Planning Fibrina Request Expense

Pada minggu yang sama dengan kegiatan pemahaman produk, penyusunan *Test Planning* dan *Test Skenario* secara rinci mulai dilaksanakan, berfokus pada modul request expense sseperti pada gambar 3.8. Modul ini diidentifikasi sebagai alur utama dari aplikasi Fibrina. Perencanaan pengujian dirancang untuk mencakup seluruh skenario pengguna, di mana pengguna diberikan kapabilitas untuk mengajukan pembayaran dengan tiga tipe pengeluaran utama, yaitu tipe *Invoice*, *Reimburse*, dan *Cash Advance*.

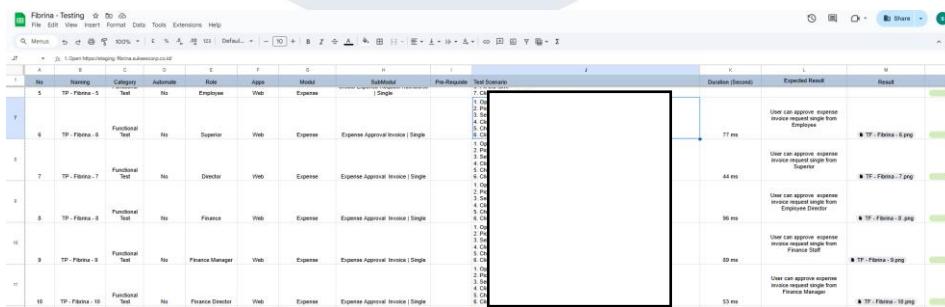
Skenario *request expense* dibagi menjadi dua jenis utama, yaitu *positive flow* dan *negative flow*. Alur positif merupakan kondisi ketika skenario pengujian dirancang berjalan maju, dimulai dari menekan tombol *create* hingga pengajuan berhasil disubmit. Sementara itu, alur negatif merupakan kondisi pengujian yang dibuat berjalan maju lalu mundur kembali, misalnya ketika pengajuan dibuat sebagai *draft* alih-alih langsung *di-submit*. Kemudian, dilakukan pengujian pada proses pengeditan *draft* pengeluaran yang telah disimpan sebelumnya.

Kompleksitas pengujian ditingkatkan karena setiap tipe pengeluaran memiliki dua variasi kategori, yaitu *single* dan *multiple*.

Kategori *single* digunakan ketika permintaan pengeluaran hanya untuk satu jenis beban biaya, sedangkan kategori *multiple* digunakan jika pengajuan melibatkan lebih dari satu tipe beban biaya. Oleh karena itu, *test planning* yang dilaksanakan menjadi sangat beragam.

Secara rinci, alur positif diuji untuk enam skenario yang meliputi *create* hingga *submit* pada kombinasi *invoice single* dan *multiple*, *reimburse single* dan *multiple*, serta *cash advance single* dan *multiple*. Selain itu, alur negatif diuji dengan dua variasi utama mulai dari *create* hingga *draft*, dan *edit draft* hingga *submit*. Karena setiap variasi alur negatif diterapkan pada enam kombinasi tipe dan kategori, total terdapat 12 skenario pengujian negatif. Dengan demikian, total keseluruhan *test plan* yang dibuat untuk modul *request expense* ini mencapai 18 skenario.

3.3.1.4 Membuat *Test Planning* dan *Skenario* modul *Approval* dan *Rejection Expense*.



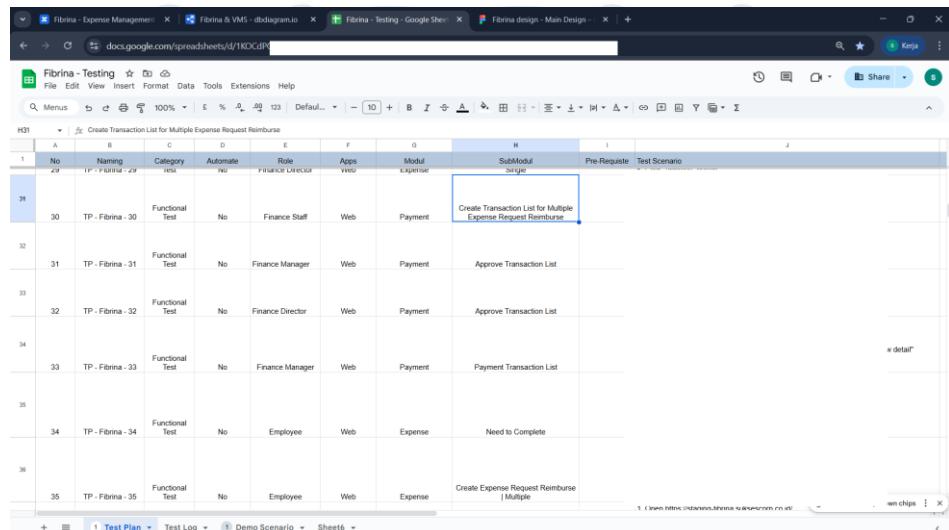
Gambar 3. 9 Potongan Test Planning Approval dan Rejection Fibrina

Pada minggu ini, penyusunan *Test Planning* dan *Test Skenario* difokuskan pada modul persetujuan (*approval flow*) seperti pada gambar 3.9. Proses persetujuan ini melibatkan lima *role* utama, yaitu *Employee Superior*, *Employee Director*, *Finance Staff*, *Finance Manager*, dan *Finance Director*. Skenario pengujian dirancang untuk mencakup alur positif (*approve*) dan alur negatif (*rejection*) dari setiap tipe pengeluaran.

Untuk alur *approve*, pengujian dilakukan pada enam skenario yang meliputi *approve expense* dengan tipe *invoice single* dan *multiple*, *reimburse single* dan *multiple*, serta *cash advance single* dan *multiple*. Sementara itu, alur *rejection* melibatkan pengujian pada enam skenario yang serupa, namun berfokus pada *rejection* pengeluaran. Dengan demikian, setiap *role approver* membutuhkan total 12 skenario pengujian untuk modul *approval*. Karena terdapat lima *role approver* yang berbeda, maka total keseluruhan *test planning* yang dibuat dan didokumentasikan mencapai 60 skenario .

Total keseluruhan test planning yang dibuat untuk pengujian aplikasi Fibrina adalah 135 skenario mulai dari flow positive hingga flow negative. Test planning ini mengakup dari role *employee*, *employee superior*, *employee director*, *finance staff*, *finance manager* dan *finance director*. Dengan total tiga modul utama modul *expense*, *approve expense* dan *payment*.

3.3.1.5 Membuat Test Planning dan Skenario modul *Payment*.



No	Test ID	Name	Category	Risk	Automate	Role	Age	Modul	SubModul	Pre-Request	Test Scenario
29	TP - Fibrina - 29	Functional Test	High	Low	No	Finance Director	Web	Expense	Expense	Create Transaction List for Multiple Expense Request Reimburse	
30	TP - Fibrina - 30	Functional Test	Medium	Medium	No	Finance Staff	Web	Payment	Expense		
31	TP - Fibrina - 31	Functional Test	Medium	Medium	No	Finance Manager	Web	Payment	Expense	Approve Transaction List	
32	TP - Fibrina - 32	Functional Test	Medium	Medium	No	Finance Director	Web	Payment	Expense	Approve Transaction List	
33	TP - Fibrina - 33	Functional Test	Medium	Medium	No	Finance Manager	Web	Payment	Expense	Payment Transaction List	
34	TP - Fibrina - 34	Functional Test	Medium	Medium	No	Employee	Web	Expense	Expense	Need to Complete	
35	TP - Fibrina - 35	Functional Test	Medium	Medium	No	Employee	Web	Expense	Expense	Create Expense Request Reimburse	

Gambar 3. 10 Potongan *Test Planning Payment* Fibrina

Pada modul pembayaran (*Payment*), terdapat dua jenis alur utama yang diuji, yaitu alur positif dan alur negatif. Alur positif

didefinisikan sebagai kondisi ketika seluruh pengeluaran yang terdaftar dalam satu daftar transaksi (*transaction list*) seperti gambar 3.10 disetujui oleh kedua *approver* transaksi, yaitu *Finance Manager* dan *Finance Director*. Sebaliknya, alur negatif akan terjadi jika terjadi penolakan (*rejection*) oleh salah satu dari kedua *approver* tersebut.

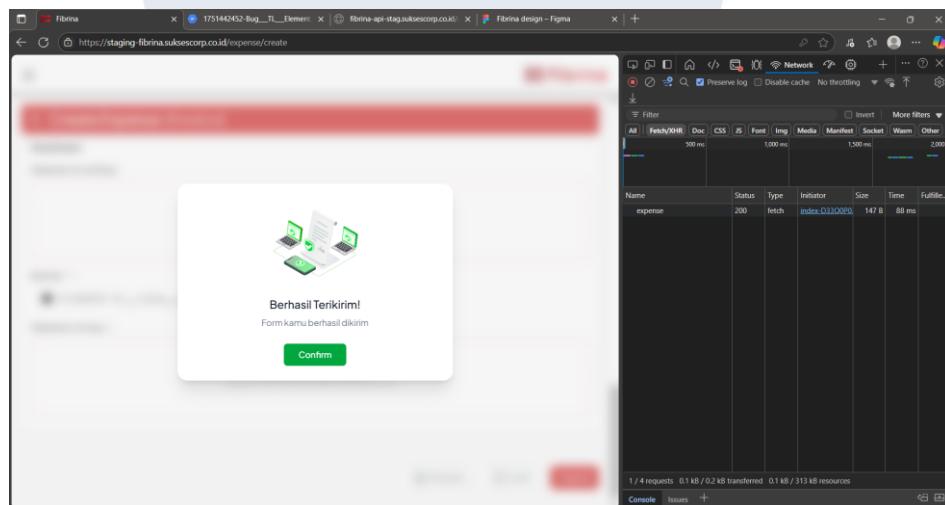
Penyusunan *Test Planning* untuk modul ini terbagi menjadi beberapa tahapan pengujian yang melibatkan berbagai *role*. Tahap awal, yaitu pembuatan daftar transaksi, dilakukan oleh *Finance Staff* untuk pengajuan pembayaran. Tahap *create* ini mencakup sembilan skenario pengujian, enam skenario untuk pembuatan daftar transaksi tunggal (*single transaction list*) yang memvalidasi *expense* bertipe *invoice single*, *invoice multiple*, *reimburse single*, *reimburse multiple*, *cash advance single*, dan *cash advance multiple*. Tiga skenario tambahan dibuat untuk skema *bulk transaction list* yang menguji penggabungan *expense single* dan *multiple* untuk tipe *invoice*, *reimburse*, dan *cash advance*. Tahap pembuatan ini menghasilkan subtotal sembilan *test planning*.

Setelah daftar transaksi dibuat, proses akan dilanjutkan ke tahap persetujuan oleh *Finance Manager*. Terdapat sembilan *test planning* yang serupa yang dirancang untuk tahap persetujuan ini. *Finance Manager* bertanggung jawab untuk menyetujui setiap daftar transaksi yang masuk, baik yang berisi satu *expense* (tunggal) maupun yang berisi banyak *expense* (*bulk*), sehingga menghasilkan subtotal sembilan *test planning* untuk persetujuan *Finance Manager*. Selanjutnya, daftar transaksi akan masuk ke tahap persetujuan *Finance Director*, yang juga memiliki subtotal sembilan *test planning* yang sama dengan *Finance Manager*.

Setelah disetujui, daftar transaksi akan kembali kepada *Finance Staff* untuk aktivitas unggah bukti pembayaran (*upload*

proof of payment). Bukti pembayaran wajib diinput per setiap *expense* secara individual, tidak dapat dilakukan secara *bulk*. Pengujian untuk tahap ini mencakup seluruh kombinasi dari transaksi tunggal dan *bulk*, sehingga total terdapat 24 *test planning*. Terakhir, *expense* akan kembali kepada *Requestor* untuk aktivitas unggah bukti penggunaan (*upload proof of usage*), dengan total 12 *test planning* yang telah disusun sehingga keseluruhan *test planning* yang dibuat untuk modul transaksi (*transaction module*) adalah 63 skenario .

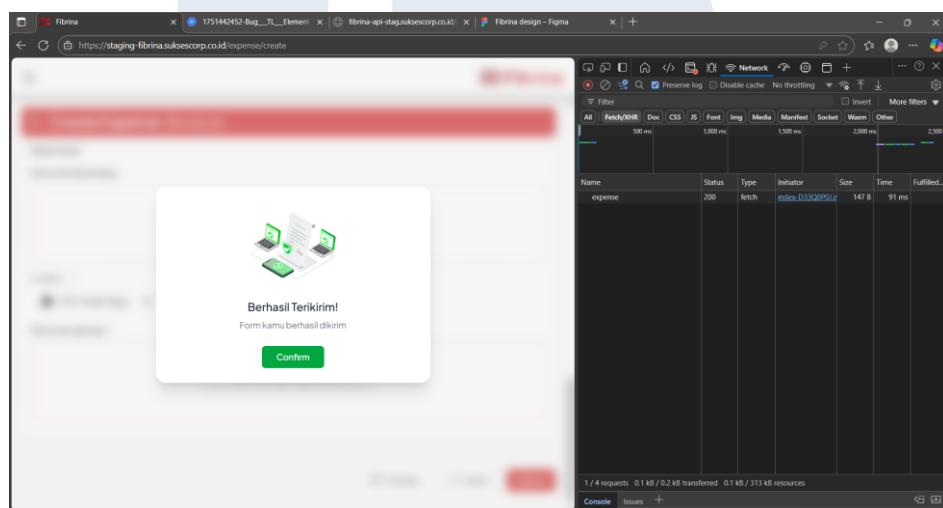
3.3.1.6 Pelaksanaan *Testing* manual modul *Request* dan *Approval Expense*.



Gambar 3. 11 Contoh Hasil *Testing Request Expense Single*

Pengujian fungsional modul *Request* dan *Approval Expense* dilakukan secara manual seperti pada gambar 3.11 untuk memvalidasi alur pembuatan (*create*) hingga pengiriman data. Salah satu hasil pengujian ditampilkan pada proses pembuatan *expense* dengan tipe *invoice single*. Hasil pengujian tersebut menunjukkan bahwa ketika tombol *submit* di antarmuka diklik, *end-point* yang dituju merespons dengan Status Code 200 (OK). Kode status ini mengindikasikan bahwa permintaan *request* yang dikirimkan oleh

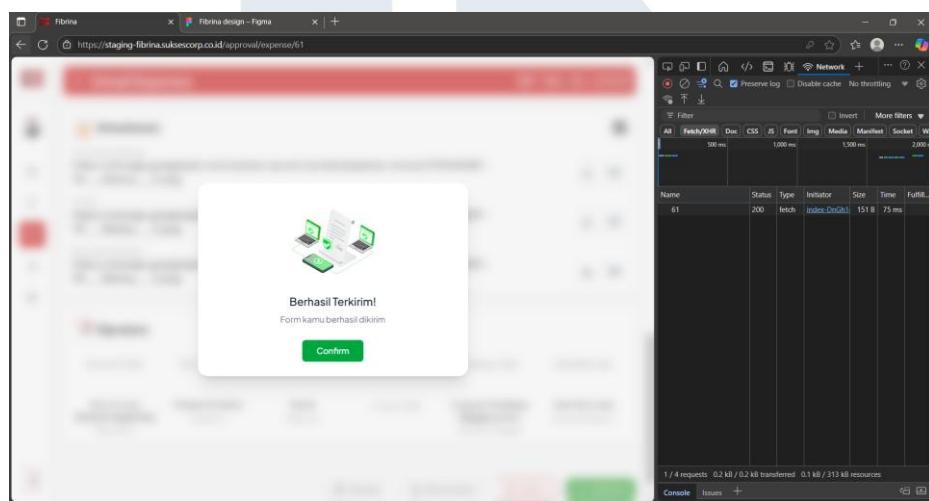
sistem (berisi data *expense* dalam format JSON) berhasil diterima dan diproses oleh server tanpa adanya kesalahan (*error*). Waktu respons yang tercatat adalah 88 milidetik (ms), yang menunjukkan bahwa pengiriman data dan respons dari server terjadi dalam durasi yang sangat cepat. Durasi respons yang singkat ini mengindikasikan kinerja *end-point* yang optimal saat memproses data *expense* tunggal. Dengan demikian, data JSON *expense* telah berhasil dikirim dan tersimpan di basis data, dan notifikasi "*Berhasil Terkirim!*" ditampilkan sebagai konfirmasi kepada pengguna.



Gambar 3. 12 Contoh Hasil *Testing Request Expense Invoice Multiple*

Pengujian fungsional modul *Request Expense* dan *Approval Expense* dilanjutkan dengan skenario pembuatan *expense* berjenis Invoice Multiple seperti pada gambar 3.12. Pengujian ini bertujuan untuk memvalidasi kemampuan sistem dalam memproses pengiriman data lebih dari satu *expense* dalam satu kali proses *submit*. Hasil pengujian yang diamati menunjukkan bahwa ketika data *multiple expense* dikirimkan dari antarmuka, *end-point* yang dituju merespons dengan Status Code 200 (OK). Kode status ini menegaskan bahwa permintaan *request* yang membawa *payload* data *multiple expense* berhasil diterima dan diproses oleh server tanpa adanya hambatan teknis. Waktu respons yang tercatat adalah

91 milidetik (ms), yang mengindikasikan bahwa kinerja *end-point* tetap berada dalam kondisi yang sangat baik meskipun memproses data *expense* dalam jumlah jamak. Kecepatan respons yang singkat ini sangat penting untuk mendukung pengalaman pengguna. Setelah proses selesai, notifikasi "Berhasil Terkirim!" ditampilkan sebagai konfirmasi bahwa data *multiple expense* telah berhasil disimpan ke basis data.

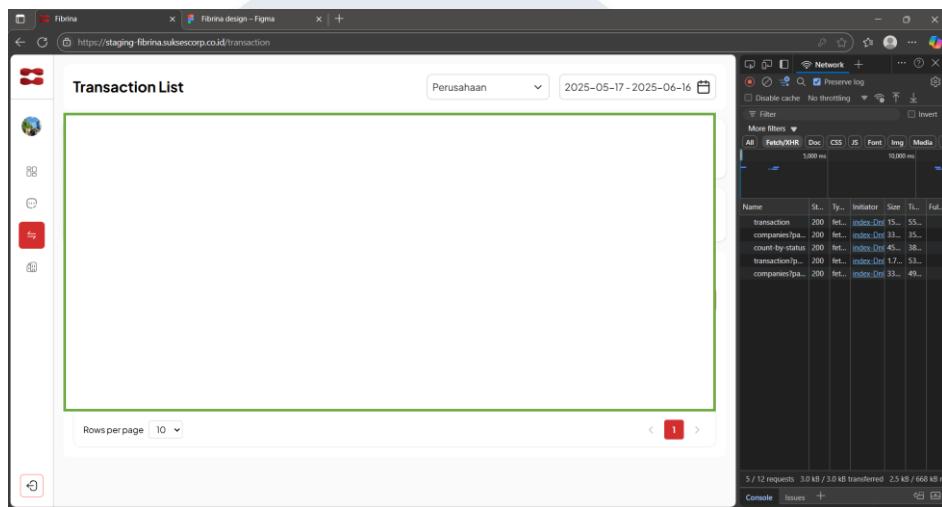


Gambar 3. 13 Contoh Hasil *Testing Approval Expense Invoice Multiple*

Pengujian fungsional modul *Request* dan *Approval Expense* dilanjutkan dengan skenario persetujuan (*approval*) *expense* yang berjenis Invoice Multiple seperti pada gambar 3.13. Pengujian ini memvalidasi kemampuan *approver* untuk menyetujui sekaligus beberapa *expense* yang terkait dalam satu permintaan. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani persetujuan (*approval*) merespons dengan Status Code 200 (OK). Kode status tersebut mengindikasikan bahwa permintaan *request* persetujuan *multiple expense* berhasil diterima dan diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat adalah 75 milidetik (ms), menunjukkan bahwa kinerja *end-point* sangat baik dan cepat dalam memproses persetujuan untuk data *expense* jamak. Kecepatan respons yang efisien ini penting untuk

memastikan kelancaran alur kerja persetujuan. Setelah proses selesai, notifikasi "Berhasil Terkirim!" ditampilkan sebagai konfirmasi bahwa status *multiple expense* telah berhasil diperbarui menjadi *approved* di basis data

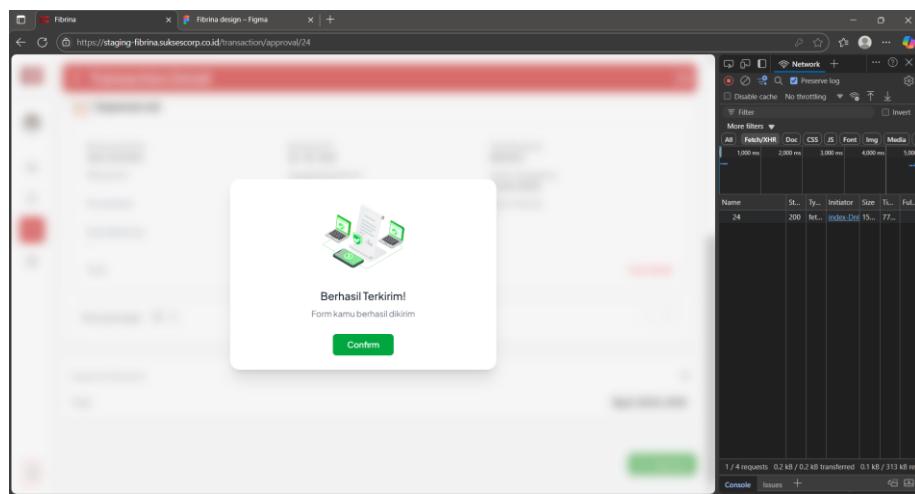
3.3.1.7 Pelaksanaan *Testing manual* modul *Transaction List* dan *Paid Expense* dan *Automation Testing*.



Gambar 3. 14 Contoh Hasil *Testing Create Transaction List*

Pengujian fungsional modul *Transaction List* dilakukan dengan skenario pembuatan daftar transaksi oleh *Finance Staff* seperti pada gambar 3.14. Modul ini bertujuan untuk mengelompokkan *expense* yang sudah *approved* menjadi satu daftar yang siap untuk proses pembayaran lebih lanjut. Hasil pengujian menunjukkan bahwa ketika perintah *Create Transaction List* dijalankan, *end-point* yang menangani pembuatan daftar tersebut merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* untuk membuat *transaction list* telah berhasil diproses oleh server tanpa kegagalan. Waktu respons yang tercatat adalah 53 milidetik (ms), mengindikasikan bahwa kinerja *end-point* sangat cepat dalam memproses *request* pembuatan data. Kecepatan ini memastikan efisiensi dalam alur kerja *Finance*. Setelah *transaction list* berhasil dibuat, data yang baru seperti pada

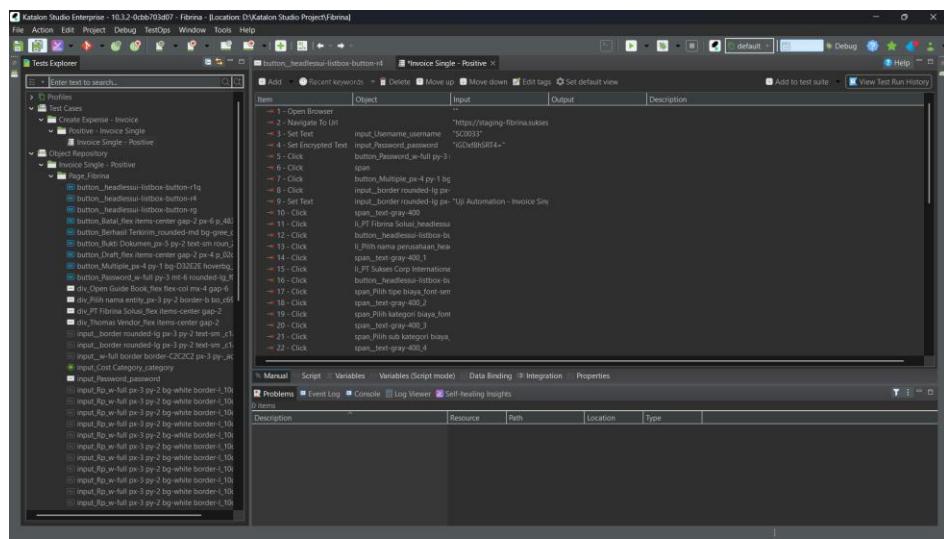
gambar 3.12 contohnya *Transaction ID* 24 pada gambar akan diperbarui pada *dashboard* dengan status awal *Need Finance Manager Approval*, menandakan bahwa alur persetujuan telah dimulai.



Gambar 3. 15 Contoh Hasil *Testing Approve Transaction List*

Pengujian fungsional modul *Transaction List* dilanjutkan dengan skenario persetujuan (*approval*) seperti pada gambar 3.15 daftar transaksi oleh *approver* yang berwenang merujuk pada *Finance Manager* atau *Finance Director* dalam alur. Pengujian ini memvalidasi kemampuan sistem untuk mengubah status *transaction list* dari *pending* menjadi *approved* melalui antarmuka persetujuan. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani aksi persetujuan merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* persetujuan yang dikirimkan oleh sistem telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat adalah 77 milidetik (ms), mengindikasikan bahwa kinerja *end-point* efisien dan responsif dalam memproses pembaruan status data. Kecepatan respons yang singkat ini memastikan kelancaran dan efektivitas alur persetujuan. Setelah proses selesai, notifikasi "*Berhasil Terkirim!*" ditampilkan, mengonfirmasi bahwa status

transaction list telah berhasil diperbarui dan dialihkan ke tahap proses berikutnya di basis data.



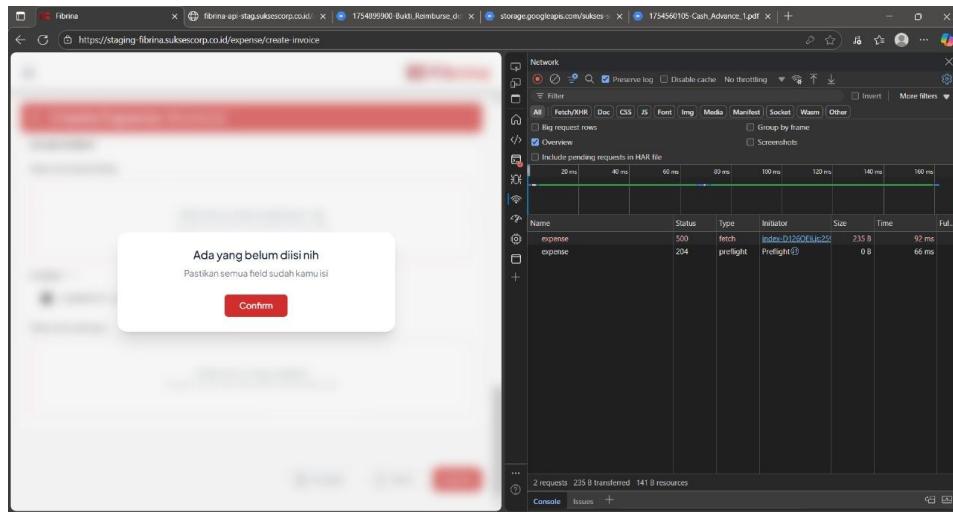
Gambar 3. 16 Hasil Testing Automation Katalon Studio

Pelaksanaan *testing automation* dilakukan menggunakan Katalon Studio Enterprise seperti pada gambar 3.16. Metode otomasi ini dipilih untuk mempermudah proses eksekusi *test case* yang bersifat pasti dan berulang. Pembuatan *test case* otomasi diawali dengan pemanfaatan fitur Web Recorder; fitur ini merekam setiap interaksi pengguna pada *browser*, yang kemudian secara otomatis menghasilkan skrip pengujian. Pada skenario ini, *testing automation* digunakan secara spesifik untuk percobaan *create expense* karena elemen antarmuka (*button* atau *element*) yang terlibat tidak banyak mengalami perubahan posisi. Katalon Studio bekerja dengan menangkap setiap item pada aplikasi sebagai objek yang unik, di mana sistem akan mendeteksi *input* yang dimasukkan oleh pengguna. Dengan demikian, serangkaian langkah *test case* yang kompleks dapat dieksekusi secara otomatis dan akurat, mengurangi waktu yang dibutuhkan untuk pengujian manual berulang.

Test Case ID	Module	Sub Module	Description	Log Type	Status	Raise Date	PC Test	PC Dev	Attachment	Dev Note
TP - Fibrina - 03 - 01	Employee	Create Expense - Invoice Single	Text box for Description expense cant fill with long text	Minor	Pass	02/07/2025 - 10:30 AM	Sabrina Azmi	Pass		
TP - Fibrina - 03 - 02	Employee	Create Expense - Invoice Single	Text box for Description of Edit Expense Detail cant fill with long text	Minor	Pass	02/07/2025 - 10:35 AM	Sabrina Azmi	Pass		
TL - Element - 1	Employee	Create Expense - Invoice Single	This text box in the Category form is required, but no alert message is shown for section Type, Category, Sub Category and Category	Minor	Pass	02/07/2025 - 2:45 PM	Sabrina Azmi	Pass		
TP - Fibrina - 06 - 01	Superior	Approve Expense Request	Unable to approve expense request for multiple invoices because the approval page does not display the list of invoices	Major	Pass	03/07/2025 - 3:45 PM	Sabrina Azmi	Pass		
TP - Fibrina - 06 - 02	Superior	Approve Expense Request	Unexpected section 'Descripti expense detail' appears on the Approval page even though the finance staff did not provide any notes	Minor	Pass	03/07/2025 - 5:00 PM	Sabrina Azmi	Pass		
TP - Fibrina - 07 - 01	Employee Director	Approve Expense Request	Unexpected section 'Descripti expense detail' appears on the Approval page even though the finance staff did not provide any notes	Minor	Pass	03/07/2025 - 5:15 PM	Sabrina Azmi	Pass		
TP - Fibrina - 10 - 01	Finance Director	Approve Expense Request	Unexpected section 'Descripti expense detail' appears on the Approval page even though the finance staff did not provide any notes (duplicate)	Minor	Pass	03/07/2025 - 5:30 PM	Sabrina Azmi	Pass		
TP - Fibrina - 11 - 01	Finance Staff	Transaction List	Date filter is still visible when the Company Name is selected	Minor	Pass	03/07/2025 - 5:50 PM	Sabrina Azmi	Pass		
TP - Fibrina - 11 - 02	Finance Staff	Transaction List	Transaction list can be created even though all form fields have been left empty	Minor	Pass	03/07/2025 - 5:55 PM	Sabrina Azmi	Pass		
TP - Fibrina - 04 - 01	Employee	Create Expense - Invoice Multiple	Alert 'Buatan baru berhasil' appears even though the expense can still be submitted	Minor	Pass	03/07/2025 - 6:40 PM	Sabrina Azmi	Pass		
TP - Fibrina - 04 - 02	Employee	Create Expense - Invoice Multiple	Unable to create expense even when all form fields have been left empty	Minor	Pass	03/07/2025 - 6:45 PM	Sabrina Azmi	Pass		
TP - Fibrina - 16 - 01	Employee Director	Approve Expense Request - Invoice Multiple	In the Attachment form, attachments from the previous expense request are still displayed even though the requestor identified a different expense	Minor	Pass	04/07/2025 - 11:35 AM	Sabrina Azmi	Pass		
TP - Fibrina - 17 - 01	Employee Staff	Transaction List	User can't provide expense because the document form is empty	Minor	Pass	04/07/2025 - 1:35 PM	Sabrina Azmi	Pass		
TP - Fibrina - 04 - 03	Employee Staff	Create Expense - Invoice Multiple	Text box for Description of Edit Expense Detail cant fill with long text	Minor	Pass	04/07/2025 - 1:40 PM	Sabrina Azmi	Pass		
TP - Fibrina - 06 - 03	Superior	Approve Expense Request - Invoice Multiple	Document form is empty even though the request page although it was provided by the requested	Minor	Pass	04/07/2025 - 3:35 PM	Sabrina Azmi	Pass		
TP - Fibrina - 06 - 04	Superior	Approve Expense Request - Invoice Multiple	The system shows 'Dokumen Surat Kerja dan Dokumen Surat Pengajuan' even though the requestor left them empty	Minor	Pass	04/07/2025 - 3:35 PM	Sabrina Azmi	Pass		

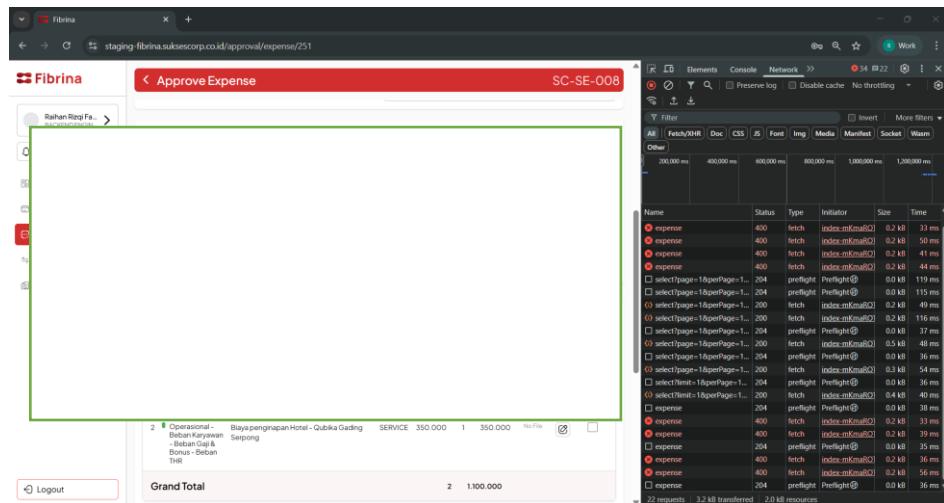
Gambar 3. 17 Hasil Test Log Fibrina

Dalam pengujian yang telah dilakukan mulai dari modul *request* hingga *payment* yang melibatkan seluruh role. Terdapat banyak temuan *error* atau *bug* yang muncul seperti pada gambar 3.17. Hal ini ditemukan bukan hanya dari pengujian berdasarkan interface tetapi juga pengujian data. Pengujian *interface* dilihat dengan acuan dari design yang telah dibuat oleh *UI/UX Designer* pada Figma seperti Gambar 3.2, menguji apakah setiap komponen sudah sesuai, tidak ada komponen yang hilang atau delay dalam penampilan data. Sedangkan pengujian data mengacu pada pelemparan data yang di input dan hasil yang diberikan, apakah data yang dimasukan dan di tampilkan sesuai, pengujian ini dilihat dari hasil response pada *end-point* yang diberikan. Pada proyek Fibrina terdapat 79 bug yang ditemukan, mulai dari *bug minor* hingga *major*.



Gambar 3. 18 Contoh Bug Major

Pada gambar 3.18 merupakan contoh bug major yang ditemukan ketika ingin menguji *create draft request expensse* menyebabkan *blocking* sehingga pengujian tidak dapat dilanjutkan ke tahap berikutnya. Hal ini dapat dilihat dari status yang dikeluarkan oleh *end-point* berupa 500 dengan permasalahan pada *backend* yang belum membuat development untuk *case draft*.

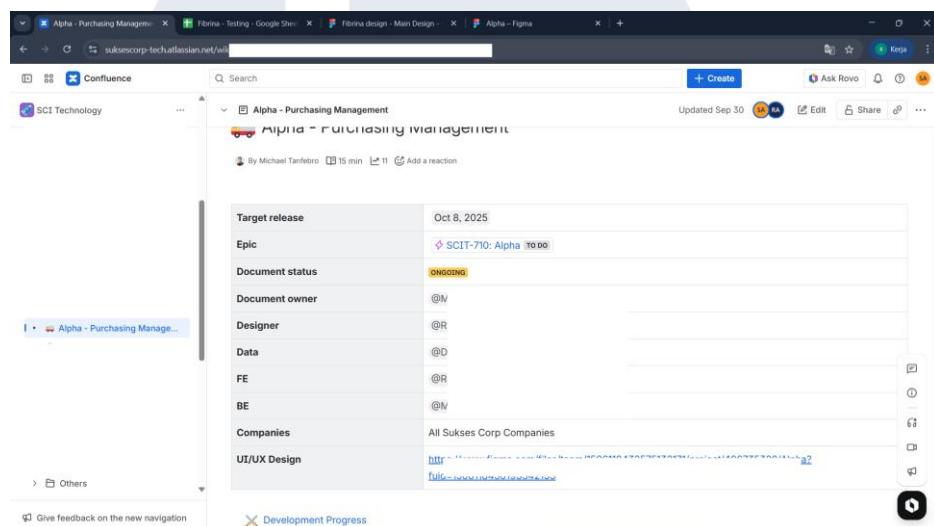


Gambar 3. 19 Contoh Bug Minor

Selain *bug major* yang ditemukan sebelumnya, terdapat pula temuan bug minor, seperti gambar 3.19. *Bug* jenis ini biasanya muncul tanpa mengakibatkan *blocking* pada alur kerja utama pengguna.

Meskipun demikian, temuan ini penting untuk diperbaiki karena seringkali terkait dengan *request* yang tidak diterima oleh *server*, seperti yang ditunjukkan oleh Status Code 400 (*Bad Request*). Kode status ini mengindikasikan adanya masalah pada permintaan yang dikirimkan oleh klien, seperti data yang tidak valid atau *format request* yang salah.

3.3.1.8 Pemahaman produk Alpha, membuat *Test Planning* dan *Skenario* modul *Request* dan *Approve RO, PO, dan DO*.

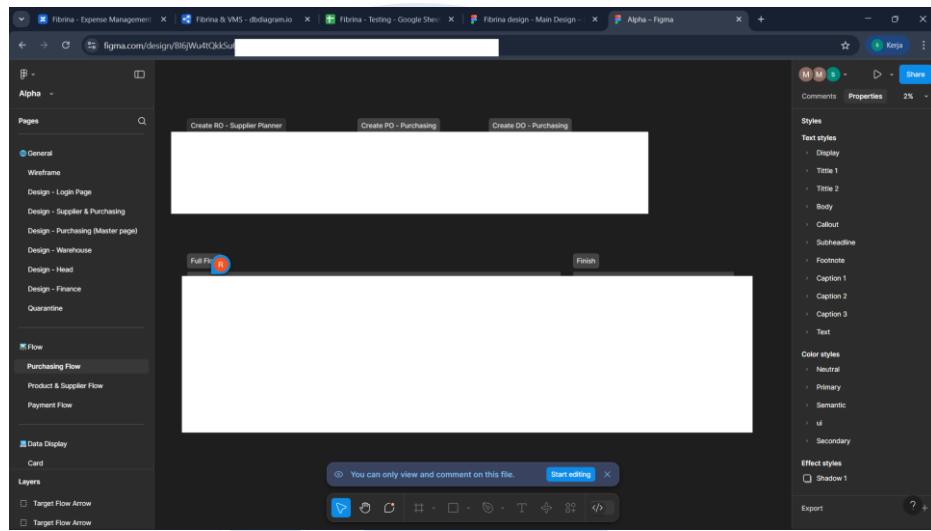


Target release	Oct 8, 2025
Epic	SCIT-710: Alpha
Document status	ONGOING
Document owner	@M
Designer	@R
Data	@D
FE	@R
BE	@M
Companies	All Sukses Corp Companies
UI/UX Design	http://fibrina.suksescorp-tech.azmi.net

Gambar 3. 20 Confluence Alpha

Setelah penyelesaian pengujian dokumentasi Fibrina, proses pemahaman produk baru dilakukan untuk proyek *Alpha* pada gambar 3.20. *Alpha* merupakan aplikasi yang dirancang khusus untuk kebutuhan manajemen pembelian (*Purchasing Management*) perusahaan, dengan *target release* pada tanggal 8 Oktober 2025. Aplikasi ini terdiri dari beberapa modul utama, yaitu *Request Order* (RO), *Purchase Order* (PO), *Delivery Order* (DO), *Good Receipt* (GR), dan *Payment*. Penggunaan aplikasi ini melibatkan *role* berjenjang, termasuk *Supplier Planner*, *Purchasing Staff*, *Purchasing Head*, *Finance Staff*, *Finance Manager*, *Finance Director*, dan *Warehouse*. Pemahaman mendalam ini mencakup analisis alur proses pembelian secara menyeluruh, mulai dari

pembuatan dokumen awal hingga proses persetujuan dan pembayaran yang memiliki kompleksitas bersyarat. Berdasarkan pemahaman tersebut, *Test Planning* dan *Test Skenario* mulai disusun untuk memvalidasi seluruh skenario bisnis dan batasan *role* di setiap modul.



Gambar 3. 21 Figma Alpha

Selain peninjauan dokumentasi di Confluence, penjelasan lebih rinci terkait aplikasi juga dipelajari melalui *tools* Figma, seperti yang ditunjukkan pada Gambar 3.21. Pemahaman desain ini sangat penting karena di dalamnya dimuat seluruh detail visual *user interface*. Hal ini meliputi perancangan tata letak halaman, peletakan tombol (*button*), penggunaan komponen, serta konsistensi visual di seluruh modul. Pemahaman desain di Figma ini berfungsi sebagai acuan utama bagi pengujian *UI* dan *UX*, memastikan bahwa implementasi yang dilakukan oleh tim *Engineer* sesuai dengan spesifikasi visual yang telah disepakati oleh *UI/UX Designer*.

No	Naming	Category	Automate	Role	Apps	Model	SubModel	Total Scenario	Duration (Second)	Expected Result	Result	Status	PC Dev
4	TP - Alpha - 4 Functional Test	No	Supplier	Web	Request Order	Create Request Order	1. Create Request Order which includes	3. All	17 ms	User can create request order by fill all fields	■ TP - Alpha - 04 (1).jpg	PASS	PASS
5	TP - Alpha - 5 Functional Test	No	Supplier	Web	Request Order	Draft Request Order	1. Draft Request Order which includes	1. Draft	19 ms	User can drafting Request Order	■ TP - Alpha - 05 (1).jpg	PASS	PASS
6	TP - Alpha - 6 Functional Test	No	Supplier	Web	Request Order	Edit Request Order	1. Edit Request Order which includes	3. All	29 ms	User can edit RO from Draft status	■ TP - Alpha - 06 (1).jpg	PASS	PASS
7	TP - Alpha - 7 Functional Test	No	Supplier	Web	Request Order	Cancel Request Order	1. Cancel Request Order which includes	3. All	41 ms	User can cancel RO when no one PO selected	■ TP - Alpha - 07 (1).jpg	PASS	PASS
8	TP - Alpha - 8 Functional Test	No	Supplier	Web	Request Order	Delete Request Order - Cancelling	3. All	41 ms	User can delete Request Order which already submitted before with status 'Cancelled'	■ TP - Alpha - 08 (1).jpg	NOT PASS	FAIL	FAIL
							1. Cancel Request Order which includes						

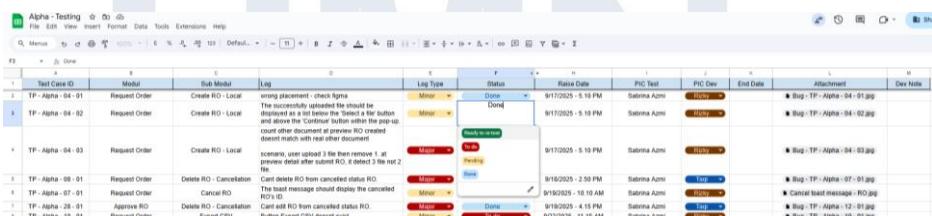
Gambar 3. 22 Potongan Lembar Kerja *Test Planning*.

Setelah dilakukan pemahaman produk, selanjutnya dilaksanakan pembuatan lembar kerja untuk *Test Planning* seperti pada gambar 3.22. Lembar kerja ini digunakan sebagai dokumentasi strategis bagi *QA* dalam melaksanakan *testing*, serta berfungsi sebagai sumber informasi yang dapat diakses oleh *front end* dan *back end developer* untuk memverifikasi kelengkapan cakupan pengujian.

Dalam penulisan *Test Planning*, sebanyak 14 kolom menjadi persyaratan utama yang harus dipenuhi. Kolom pertama adalah *Naming*, yang berisi *ID* pengujian yang ditulis secara berurutan dengan format *TP - Alpha - 4* mengacu pada *Test Planning* untuk aplikasi Alpha dan angka terakhir menunjukkan bahwa skenario tersebut ada dalam urutan keberapa. Selanjutnya, terdapat kolom *Category*, yang berfungsi sebagai indikator pengujian dengan dua pilihan yaitu *Functional Test*, yang berfokus pada kegunaan setiap fitur dan alur dan *Regression Test*, yang merupakan pengujian ulang setelah adanya perbaikan temuan atau masalah. Kolom *Automate* digunakan untuk memberikan keterangan mengenai metode pengujian, apakah dibantu dengan alat otomasi atau dilakukan secara manual. Kolom *Role* menjelaskan peran pengguna yang menjalankan skenario, sementara kolom *Apps* memberikan

penjelasan mengenai tipe aplikasi yang diuji, apakah berbasis *web* atau *mobile*.

Kolom *Modul* dan *Submodul* berfungsi untuk menjelaskan fungsi umum dan fungsi lebih rinci dari skenario yang dibuat. *Test Skenario* berisi tahapan yang harus dieksekusi dari awal hingga akhir, dan *Duration* mencatat hasil *end point* yang dikeluarkan oleh sistem untuk mengukur kecepatan *back end* dalam memproses data. Kolom *Expected Result* menjelaskan hasil yang seharusnya diberikan pada skenario tersebut, dan *Result* berisi hasil sesungguhnya dari skenario yang telah dilakukan. Kolom Status memberikan keterangan hasil pengujian yang terdiri dari empat opsi: *Pass* memiliki arti bahwa pengujian berhasil, *Not Pass* memiliki arti bahwa pengujian gagal, *Skip-Remove* memiliki arti bahwa pengujian dilewati karena skenario dihapus atau tidak relevan, dan *Skip-Double* memiliki arti bahwa pengujian dilewati karena merupakan skenario duplikasi. Terakhir, kolom *PC Dev* dan *PC QA* mencatat nama *developer* yang mengerjakan dan *QA* yang melakukan pengujian.



	A	B	C	D	E	F	G	H	I	J	K	L	M
	Test Case ID	Modul	Sub Modul	Log	Log Type	Raise Date	PIC Test	PIC Dev	End Date	Attachment	Dev Note		
1	TP - Alpha - 04 - 01	Request Order	Create RO - Local	Wrong placement of file type. User accidentally uploaded the invalid file displayed as a list below the 'Select a file' button and 'Choose file' button. The uploaded file is not a valid file type.	Minor	9/17/2025 - 5:10 PM	Sabrina Azmi	Pass		■ Bug - TP - Alpha - 04 - 01.jpg			
2	TP - Alpha - 04 - 02	Request Order	Create RO - Local		Minor	9/17/2025 - 5:10 PM	Sabrina Azmi	Pass		■ Bug - TP - Alpha - 04 - 02.jpg			
3	TP - Alpha - 04 - 03	Request Order	Create RO - Local	Scenario: user uploaded 3 file then remove 1 at the same time. The system detail after submit RO, it deleted 3 file not 2 file.	Minor	9/17/2025 - 5:10 PM	Sabrina Azmi	Pass		■ Bug - TP - Alpha - 04 - 03.jpg			
4	TP - Alpha - 08 - 01	Request Order	Delete RO - Cancellation	Can't delete RO from canceled status RO.	Minor	9/18/2025 - 2:50 PM	Sabrina Azmi	Pass		■ Bug - TP - Alpha - 07 - 01.jpg			
5	TP - Alpha - 07 - 01	Request Order	Cancel RO	Cancel last message should display the canceled RO ID.	Minor	9/18/2025 - 10:10 AM	Sabrina Azmi	Pass		■ Cancel last message - RO.jpg			
6	TP - Alpha - 28 - 01	Agarve RO	Delete RO - Cancellation	Can't edit RO from canceled status RO.	Minor	9/18/2025 - 4:15 PM	Sabrina Azmi	Pass		■ Bug - TP - Alpha - 07 - 01.jpg			
7	TP - Alpha - 14 - 01	Blancard Order	Blancard Order	Blancard Order file is invalid.	Minor	9/18/2025 - 4:15 PM	Sabrina Azmi	Pass		■ Bug - TP - Alpha - 14 - 01.jpg			

Gambar 3.23 Potongan Lembar Kerja *Test Log*

Seluruh temuan *bug* dan *error* didokumentasikan dalam lembar kerja *Test Log*. Lembar kerja ini memiliki 12 kolom yang berfungsi untuk memonitor siklus hidup setiap *bug* seperti pada gambar 3.23. Kolom *Test Case ID* memiliki penamaan unik yang mengacu pada *ID* dari *Test Planning*. Sebagai contoh, *TP - Alpha - 08 - 01* mengindikasikan bahwa itu adalah *bug* pertama yang ditemukan saat pelaksanaan skenario kedelapan dari *Test Plan*.

Alpha. Selanjutnya, kolom *Modul* dan *Submodul* digunakan untuk mengidentifikasi fungsionalitas yang terpengaruh. Kolom *Log* berisi deskripsi terperinci mengenai *error* yang ditemukan, termasuk penjelasan tentang kondisi *bug* saat ini dan bagaimana hasil yang seharusnya ditampilkan. Kolom *Log Type* mengklasifikasikan tingkat keparahan permasalahan, apakah termasuk kategori *Minor* atau *Major*.

Kolom Status pada *Test Log* memiliki empat kategori yang menunjukkan progres perbaikan. Status *To Do* mengindikasikan bahwa *bug* telah dilaporkan oleh *QA* dan sedang menunggu perbaikan oleh *developer*. Status *Pending* berarti perbaikan *bug* akan ditunda oleh *developer* karena adanya prioritas atau kepentingan lain. Status *Ready to re-test* mengartikan bahwa *bug* sudah selesai diperbaiki oleh *developer* dan siap untuk diuji kembali oleh *QA*. Terakhir, status *Done* menunjukkan bahwa *QA* telah selesai memverifikasi perbaikan *bug*. Kolom *Raise Date* mencatat tanggal *bug* ditemukan dan dilaporkan oleh *QA*. *PIC Test* mencantumkan penanggung jawab yang melakukan pengujian, dan *PIC Dev* mengarahkan *bug* tersebut kepada tim yang bertanggung jawab untuk perbaikan, apakah *front end* atau *back end*. Kolom *Attachment* digunakan untuk melampirkan bukti visual saat *QA* menemukan *bug* selama pengujian, dan kolom *Dev Note* berfungsi untuk mencatat catatan atau komentar dari *developer* terkait proses perbaikan.

Test ID	Test Name	Category	Automate	Role	App	Modul	Submodul	Test Scenario	Duration (Second)	Expected Result
4 3	TP - Alpha - 3	Functional Test	No	Supplier	Web	Request Order	Create Request Order - Import att	wide		User can create request order by in csv file type.
5 4	TP - Alpha - 4	Functional Test	No	Supplier	Web	Request Order	Create Request Order	wide	117 ms	User can create request order by file.
6 5	TP - Alpha - 5	Functional Test	No	Supplier	Web	Request Order	Draft Request Order	wide	133 ms	User can drafting Request Order
7 6	TP - Alpha - 6	Functional Test	No	Supplier	Web	Request Order	Edit Request Order	button	779 ms	User can edit RO from Draft state
8 7	TP - Alpha - 7	Functional Test	No	Supplier	Web	Request Order	Cancel Request Order	button	96 ms	User can cancel RO when no one created.
9 8	TP - Alpha - 8	Functional Test	No	Supplier	Web	Request Order	Delete Request Order - Cancelled	wide		User can delete Request Order if all submitted before with status 'Cancel'.
10 9	TP - Alpha - 9	Functional Test	No	Supplier	Web	Request Order	Delete Request Order - Draft	wide	87 ms	User can delete Request Order from Draft state.
11 10	TP - Alpha - 10	Functional Test	No	Supplier	Web	List Overview	Export CSV	wide		User can Export data to CSV file
										User can sort request RO based on

Gambar 3. 24 Potongan Test Planning RO, PO, dan DO

Penyusunan *Test Planning* dan *Test Skenario* seperti pada gambar 3.24 untuk modul RO, PO, dan DO dilaksanakan setelah pemahaman terhadap *product requirement* selesai dilakukan. Skenario pengujian dirancang untuk mencakup *positive case* dan *negative case* guna memvalidasi seluruh fungsionalitas dan integrasi logis antarmodul. Secara keseluruhan, 29 *Test Planning* telah disusun untuk ketiga modul ini.

Modul RO diuji melalui total 12 skenario. Alur pembuatan dan manajemen RO diuji melalui delapan skenario, yang terdiri dari *create local* dan *import*, *cancel RO local* dan *import*, *draft RO*, dan *edit RO*. Alur negatif pembatalan RO diatur dengan syarat ketat yaitu apabila RO telah dibuatkan PO oleh purchasing staff, PO terkait wajib dibatalkan terlebih dahulu, memastikan tidak ada PO yang berjalan saat RO dibatalkan. Selain itu, empat *Test Plan* disiapkan untuk tahap persetujuan oleh *Procurement Head*, yang meliputi pengujian *approve* dan *reject* untuk RO *local* dan *import*.

Pengujian untuk PO juga disusun menjadi 12 skenario. Sebanyak delapan *Test Plan* dialokasikan untuk pembuatan dan manajemen PO, yang terdiri dari *create local* dan *import*, *cancel PO local* dan *import*, *draft PO*, dan *edit PO*. Pembatalan PO dapat

dilakukan dengan prosedur berjenjang: PO hanya dapat dibatalkan apabila belum dibuatkan DO, dan apabila DO telah ada, maka DO tersebut harus dibatalkan terlebih dahulu. Empat *Test Plan* disiapkan untuk tahap persetujuan oleh *Purchasing Head*, meliputi *approve* dan *reject* untuk PO *local* dan *import*.

Terakhir, modul DO diuji menggunakan lima *Test Plan* yang difokuskan pada variasi tipe pembuatan DO. Lima skenario *positive flow* telah disusun berdasarkan jenis pengiriman, yaitu tiga skenario *local* dengan pemilihan menggunakan ekspedisi, *pick-up*, dan *supplier driver* dan dua skenario *import* dengan pemilihan menggunakan *sea* dan *air*. Mekanisme pembatalan juga diuji secara berjenjang. Pembatalan DO hanya dapat dilaksanakan apabila tidak terdapat GR yang melekat. Apabila GR telah dibuat, pembatalan GR wajib dilakukan terlebih dahulu sebelum pembatalan DO dapat diproses. Seluruh skenario ini dirancang untuk memvalidasi batasan *role* dan integrasi logis antarmodul agar sistem dapat beroperasi sesuai dengan kebutuhan bisnis.

3.3.1.9 Membuat Test Planning dan Skenario modul *GR* dan *Payment*.

	No	Name	Category	Automate	Rate	Apps	Module	SubModule	Test Scenario	Duration (second)	Expected Result
W	55	TP - Alpha - 55	Functional Test	No	Purchasing	Web	Payment	Create Payment - Purchase Order	1. ARI 2. ARI 3. ARI 4. ARI 5. ARI 6. ARI 7. ARI 8. ARI 9. ARI 10. ARI 11. ARI 12. ARI 13. ARI 14. ARI 15. ARI 16. ARI 17. ARI 18. ARI 19. ARI 20. ARI 21. ARI 22. ARI 23. ARI 24. ARI 25. ARI 26. ARI 27. ARI 28. ARI 29. ARI 30. ARI 31. ARI 32. ARI 33. ARI 34. ARI 35. ARI 36. ARI 37. ARI 38. ARI 39. ARI 40. ARI 41. ARI 42. ARI 43. ARI 44. ARI 45. ARI 46. ARI 47. ARI 48. ARI 49. ARI 50. ARI 51. ARI 52. ARI 53. ARI 54. ARI 55. ARI 56. ARI 57. ARI 58. ARI 59. ARI 60. ARI 61. ARI 62. ARI 63. ARI 64. ARI 65. ARI 66. ARI 67. ARI 68. ARI 69. ARI 70. ARI 71. ARI 72. ARI 73. ARI 74. ARI 75. ARI 76. ARI 77. ARI 78. ARI 79. ARI 80. ARI 81. ARI 82. ARI 83. ARI 84. ARI 85. ARI 86. ARI 87. ARI 88. ARI 89. ARI 90. ARI 91. ARI 92. ARI 93. ARI 94. ARI 95. ARI 96. ARI 97. ARI 98. ARI 99. ARI 100. ARI 101. ARI 102. ARI 103. ARI 104. ARI 105. ARI 106. ARI 107. ARI 108. ARI 109. ARI 110. ARI 111. ARI 112. ARI 113. ARI 114. ARI 115. ARI 116. ARI 117. ARI 118. ARI 119. ARI 120. ARI 121. ARI 122. ARI 123. ARI 124. ARI 125. ARI 126. ARI 127. ARI 128. ARI 129. ARI 130. ARI 131. ARI 132. ARI 133. ARI 134. ARI 135. ARI 136. ARI 137. ARI 138. ARI 139. ARI 140. ARI 141. ARI 142. ARI 143. ARI 144. ARI 145. ARI 146. ARI 147. ARI 148. ARI 149. ARI 150. ARI 151. ARI 152. ARI 153. ARI 154. ARI 155. ARI 156. ARI 157. ARI 158. ARI 159. ARI 160. ARI 161. ARI 162. ARI 163. ARI 164. ARI 165. ARI 166. ARI 167. ARI 168. ARI 169. ARI 170. ARI 171. ARI 172. ARI 173. ARI 174. ARI 175. ARI 176. ARI 177. ARI 178. ARI 179. ARI 180. ARI 181. ARI 182. ARI 183. ARI 184. ARI 185. ARI 186. ARI 187. ARI 188. ARI 189. ARI 190. ARI 191. ARI 192. ARI 193. ARI 194. ARI 195. ARI 196. ARI 197. ARI 198. ARI 199. ARI 200. ARI 201. ARI 202. ARI 203. ARI 204. ARI 205. ARI 206. ARI 207. ARI 208. ARI 209. ARI 210. ARI 211. ARI 212. ARI 213. ARI 214. ARI 215. ARI 216. ARI 217. ARI 218. ARI 219. ARI 220. ARI 221. ARI 222. ARI 223. ARI 224. ARI 225. ARI 226. ARI 227. ARI 228. ARI 229. ARI 230. ARI 231. ARI 232. ARI 233. ARI 234. ARI 235. ARI 236. ARI 237. ARI 238. ARI 239. ARI 240. ARI 241. ARI 242. ARI 243. ARI 244. ARI 245. ARI 246. ARI 247. ARI 248. ARI 249. ARI 250. ARI 251. ARI 252. ARI 253. ARI 254. ARI 255. ARI 256. ARI 257. ARI 258. ARI 259. ARI 260. ARI 261. ARI 262. ARI 263. ARI 264. ARI 265. ARI 266. ARI 267. ARI 268. ARI 269. ARI 270. ARI 271. ARI 272. ARI 273. ARI 274. ARI 275. ARI 276. ARI 277. ARI 278. ARI 279. ARI 280. ARI 281. ARI 282. ARI 283. ARI 284. ARI 285. ARI 286. ARI 287. ARI 288. ARI 289. ARI 290. ARI 291. ARI 292. ARI 293. ARI 294. ARI 295. ARI 296. ARI 297. ARI 298. ARI 299. ARI 300. ARI 301. ARI 302. ARI 303. ARI 304. ARI 305. ARI 306. ARI 307. ARI 308. ARI 309. ARI 310. ARI 311. ARI 312. ARI 313. ARI 314. ARI 315. ARI 316. ARI 317. ARI 318. ARI 319. ARI 320. ARI 321. ARI 322. ARI 323. ARI 324. ARI 325. ARI 326. ARI 327. ARI 328. ARI 329. ARI 330. ARI 331. ARI 332. ARI 333. ARI 334. ARI 335. ARI 336. ARI 337. ARI 338. ARI 339. ARI 340. ARI 341. ARI 342. ARI 343. ARI 344. ARI 345. ARI 346. ARI 347. ARI 348. ARI 349. ARI 350. ARI 351. ARI 352. ARI 353. ARI 354. ARI 355. ARI 356. ARI 357. ARI 358. ARI 359. ARI 360. ARI 361. ARI 362. ARI 363. ARI 364. ARI 365. ARI 366. ARI 367. ARI 368. ARI 369. ARI 370. ARI 371. ARI 372. ARI 373. ARI 374. ARI 375. ARI 376. ARI 377. ARI 378. ARI 379. ARI 380. ARI 381. ARI 382. ARI 383. ARI 384. ARI 385. ARI 386. ARI 387. ARI 388. ARI 389. ARI 390. ARI 391. ARI 392. ARI 393. ARI 394. ARI 395. ARI 396. ARI 397. ARI 398. ARI 399. ARI 400. ARI 401. ARI 402. ARI 403. ARI 404. ARI 405. ARI 406. ARI 407. ARI 408. ARI 409. ARI 410. ARI 411. ARI 412. ARI 413. ARI 414. ARI 415. ARI 416. ARI 417. ARI 418. ARI 419. ARI 420. ARI 421. ARI 422. ARI 423. ARI 424. ARI 425. ARI 426. ARI 427. ARI 428. ARI 429. ARI 430. ARI 431. ARI 432. ARI 433. ARI 434. ARI 435. ARI 436. ARI 437. ARI 438. ARI 439. ARI 440. ARI 441. ARI 442. ARI 443. ARI 444. ARI 445. ARI 446. ARI 447. ARI 448. ARI 449. ARI 450. ARI 451. ARI 452. ARI 453. ARI 454. ARI 455. ARI 456. ARI 457. ARI 458. ARI 459. ARI 460. ARI 461. ARI 462. ARI 463. ARI 464. ARI 465. ARI 466. ARI 467. ARI 468. ARI 469. ARI 470. ARI 471. ARI 472. ARI 473. ARI 474. ARI 475. ARI 476. ARI 477. ARI 478. ARI 479. ARI 480. ARI 481. ARI 482. ARI 483. ARI 484. ARI 485. ARI 486. ARI 487. ARI 488. ARI 489. ARI 490. ARI 491. ARI 492. ARI 493. ARI 494. ARI 495. ARI 496. ARI 497. ARI 498. ARI 499. ARI 500. ARI 501. ARI 502. ARI 503. ARI 504. ARI 505. ARI 506. ARI 507. ARI 508. ARI 509. ARI 510. ARI 511. ARI 512. ARI 513. ARI 514. ARI 515. ARI 516. ARI 517. ARI 518. ARI 519. ARI 520. ARI 521. ARI 522. ARI 523. ARI 524. ARI 525. ARI 526. ARI 527. ARI 528. ARI 529. ARI 530. ARI 531. ARI 532. ARI 533. ARI 534. ARI 535. ARI 536. ARI 537. ARI 538. ARI 539. ARI 540. ARI 541. ARI 542. ARI 543. ARI 544. ARI 545. ARI 546. ARI 547. ARI 548. ARI 549. ARI 550. ARI 551. ARI 552. ARI 553. ARI 554. ARI 555. ARI 556. ARI 557. ARI 558. ARI 559. ARI 560. ARI 561. ARI 562. ARI 563. ARI 564. ARI 565. ARI 566. ARI 567. ARI 568. ARI 569. ARI 570. ARI 571. ARI 572. ARI 573. ARI 574. ARI 575. ARI 576. ARI 577. ARI 578. ARI 579. ARI 580. ARI 581. ARI 582. ARI 583. ARI 584. ARI 585. ARI 586. ARI 587. ARI 588. ARI 589. ARI 590. ARI 591. ARI 592. ARI 593. ARI 594. ARI 595. ARI 596. ARI 597. ARI 598. ARI 599. ARI 600. ARI 601. ARI 602. ARI 603. ARI 604. ARI 605. ARI 606. ARI 607. ARI 608. ARI 609. ARI 610. ARI 611. ARI 612. ARI 613. ARI 614. ARI 615. ARI 616. ARI 617. ARI 618. ARI 619. ARI 620. ARI 621. ARI 622. ARI 623. ARI 624. ARI 625. ARI 626. ARI 627. ARI 628. ARI 629. ARI 630. ARI 631. ARI 632. ARI 633. ARI 634. ARI 635. ARI 636. ARI 637. ARI 638. ARI 639. ARI 640. ARI 641. ARI 642. ARI 643. ARI 644. ARI 645. ARI 646. ARI 647. ARI 648. ARI 649. ARI 650. ARI 651. ARI 652. ARI 653. ARI 654. ARI 655. ARI 656. ARI 657. ARI 658. ARI 659. ARI 660. ARI 661. ARI 662. ARI 663. ARI 664. ARI 665. ARI 666. ARI 667. ARI 668. ARI 669. ARI 670. ARI 671. ARI 672. ARI 673. ARI 674. ARI 675. ARI 676. ARI 677. ARI 678. ARI 679. ARI 680. ARI 681. ARI 682. ARI 683. ARI 684. ARI 685. ARI 686. ARI 687. ARI 688. ARI 689. ARI 690. ARI 691. ARI 692. ARI 693. ARI 694. ARI 695. ARI 696. ARI 697. ARI 698. ARI 699. ARI 700. ARI 701. ARI 702. ARI 703. ARI 704. ARI 705. ARI 706. ARI 707. ARI 708. ARI 709. ARI 710. ARI 711. ARI 712. ARI 713. ARI 714. ARI 715. ARI 716. ARI 717. ARI 718. ARI 719. ARI 720. ARI 721. ARI 722. ARI 723. ARI 724. ARI 725. ARI 726. ARI 727. ARI 728. ARI 729. ARI 730. ARI 731. ARI 732. ARI 733. ARI 734. ARI 735. ARI 736. ARI 737. ARI 738. ARI 739. ARI 740. ARI 741. ARI 742. ARI 743. ARI 744. ARI 745. ARI 746. ARI 747. ARI 748. ARI 749. ARI 750. ARI 751. ARI 752. ARI 753. ARI 754. ARI 755. ARI 756. ARI 757. ARI 758. ARI 759. ARI 760. ARI 761. ARI 762. ARI 763. ARI 764. ARI 765. ARI 766. ARI 767. ARI 768. ARI 769. ARI 770. ARI 771. ARI 772. ARI 773. ARI 774. ARI 775. ARI 776. ARI 777. ARI 778. ARI 779. ARI 780. ARI 781. ARI 782. ARI 783. ARI 784. ARI 785. ARI 786. ARI 787. ARI 788. ARI 789. ARI 790. ARI 791. ARI 792. ARI 793. ARI 794. ARI 795. ARI 796. ARI 797. ARI 798. ARI 799. ARI 800. ARI 801. ARI 802. ARI 803. ARI 804. ARI 805. ARI 806. ARI 807. ARI 808. ARI 809. ARI 810. ARI 811. ARI 812. ARI 813. ARI 814. ARI 815. ARI 816. ARI 817. ARI 818. ARI 819. ARI 820. ARI 821. ARI 822. ARI 823. ARI 824. ARI 825. ARI 826. ARI 827. ARI 828. ARI 829. ARI 830. ARI 831. ARI 832. ARI 833. ARI 834. ARI 835. ARI 836. ARI 837. ARI 838. ARI 839. ARI 840. ARI 841. ARI 842. ARI 843. ARI 844. ARI 845. ARI 846. ARI 847. ARI 848. ARI 849. ARI 850. ARI 851. ARI 852. ARI 853. ARI 854. ARI 855. ARI 856. ARI 857. ARI 858. ARI 859. ARI 860. ARI 861. ARI 862. ARI 863. ARI 864. ARI 865. ARI 866. ARI 867. ARI 868. ARI 869. ARI 870. ARI 871. ARI 872. ARI 873. ARI 874. ARI 875. ARI 876. ARI 877. ARI 878. ARI 879. ARI 880. ARI 881. ARI 882. ARI 883. ARI 884. ARI 885. ARI 886. ARI 887. ARI 888. ARI 889. ARI 890. ARI 891. ARI 892. ARI 893. ARI 894. ARI 895. ARI 896. ARI 897. ARI 898. ARI 899. ARI 900. ARI 901. ARI 902. ARI 903. ARI 904. ARI 905. ARI 906. ARI 907. ARI 908. ARI 909. ARI 910. ARI 911. ARI 912. ARI 913. ARI 914. ARI 915. ARI 916. ARI 917. ARI 918. ARI 919. ARI 920. ARI 921. ARI 922. ARI 923. ARI 924. ARI 925. ARI 926. ARI 927. ARI 928. ARI 929. ARI 930. ARI 931. ARI 932. ARI 933. ARI 934. ARI 935. ARI 936. ARI 937. ARI 938. ARI 939. ARI 940. ARI 941. ARI 942. ARI 943. ARI 944. ARI 945. ARI 946. ARI 947. ARI 948. ARI 949. ARI 950. ARI 951. ARI 952. ARI 953. ARI 954. ARI 955. ARI 956. ARI 957. ARI 958. ARI 959. ARI 960. ARI 961. ARI 962. ARI 963. ARI 964. ARI 965. ARI 966. ARI 967. ARI 968. ARI 969. ARI 970. ARI 971. ARI 972. ARI 973. ARI 974. ARI 975. ARI 976. ARI 977. ARI 978. ARI 979. ARI 980. ARI 981. ARI 982. ARI 983. ARI 984. ARI 985. ARI 986. ARI 987. ARI 988. ARI 989. ARI 990. ARI 991. ARI 992. ARI 993. ARI 994. ARI 995. ARI 996. ARI 997. ARI 998. ARI 999. ARI 1000. ARI 1001. ARI 1002. ARI 1003. ARI 1004. ARI 1005. ARI 1006. ARI 1007. ARI 1008. ARI 1009. ARI 1010. ARI 1011. ARI 1012. ARI 1013. ARI 1014. ARI 1015. ARI 1016. ARI 1017. ARI 1018. ARI 1019. ARI 1020. ARI 1021. ARI 1022. ARI 1023. ARI 1024. ARI 1025. ARI 1026. ARI 1027. ARI 1028. ARI 1029. ARI 1030. ARI 1031. ARI 1032. ARI 1033. ARI 1034. ARI 1035. ARI 1036. ARI 1037. ARI 1038. ARI 1039. ARI 1040. ARI 1041. ARI 1042. ARI 1043. ARI 1044. ARI 1045. ARI 1046. ARI 1047. ARI 1048. ARI 1049. ARI 1050. ARI 1051. ARI 1052. ARI 1053. ARI 1054. ARI 1055. ARI 1056. ARI 1057. ARI 1058. ARI 1059. ARI 1060. ARI 1061. ARI 1062. ARI 1063. ARI 1064. ARI 1065. ARI 1066. ARI 1067. ARI 1068. ARI 1069. ARI 1070. ARI 1071. ARI 1072. ARI 1073. ARI 1074. ARI 1075. ARI 1076. ARI 1077. ARI 1078. ARI 1079. ARI 1080. ARI 1081. ARI 1082. ARI 1083. ARI 1084. ARI 1085. ARI 1086. ARI 1087. ARI 1088. ARI 1089. ARI 1090. ARI 1091. ARI 1092. ARI 1093. ARI 1094. ARI 1095. ARI 1096. ARI 1097. ARI 1098. ARI 1099. ARI 1100. ARI 1101. ARI 1102. ARI 1103. ARI 1104. ARI 1105. ARI 1106. ARI 1107. ARI 1108. ARI 1109. ARI 1110. ARI 1111. ARI 1112. ARI 1113. ARI 1114. ARI 1115. ARI 1116. ARI 1117. ARI 1118. ARI 1119. ARI 1120. ARI 1121. ARI 1122. ARI 1123. ARI 1124. ARI 1125. ARI 1126. ARI 1127. ARI 1128. ARI 1129. ARI 1130. ARI 1131. ARI 1132. ARI 1133. ARI 1134. ARI 1135. ARI 1136. ARI 1137. ARI 1138. ARI 1139. ARI 1140. ARI 1141. ARI 1142. ARI 1143. ARI 1144. ARI 1145. ARI 1146. ARI 1147. ARI 1148. ARI 1149. ARI 1150. ARI 1151. ARI 1152. ARI 1153. ARI 1154. ARI 1155. ARI 1156. ARI 1157. ARI 1158. ARI 1159. ARI 1160. ARI 1161. ARI 1162. ARI 1163. ARI 1164. ARI 1165. ARI 1166. ARI 1167. ARI 1168. ARI 1169. ARI 1170. ARI 1171. ARI 1172. ARI 1173. ARI 1174. ARI 1175. ARI 1176. ARI 1177. ARI 1178. ARI 1179. ARI 1180. ARI 1181. ARI 1182. ARI 1183. ARI 1184. ARI 1185. ARI 1186. ARI 1187. ARI 1188. ARI 1189. ARI 1190. ARI 1191. ARI 1192. ARI 1193. ARI 1194. ARI 1195. ARI 1196. ARI 1197. ARI 1198. ARI 1199. ARI 1200. ARI 1201. ARI 1202. ARI 1203. ARI 1204. ARI 1205. ARI 1206. ARI 1207. ARI 1208. ARI 1209. ARI 1210. ARI 1211. ARI 1212. ARI 1213. ARI 1214. ARI 1215. ARI 1216. ARI 1217. ARI 1218. ARI 1219. ARI 1220. ARI 1221. ARI 1222. ARI 1223. ARI 1224. ARI 1225. ARI 1226. ARI 1227. ARI 1228. ARI 1229. ARI 1230. ARI 1231. ARI 1232. ARI 1233. ARI 1234. ARI 1235. ARI 1236. ARI 1237. ARI 1238. ARI 1239. ARI 1240. ARI 1241. ARI 1242. ARI 1243. ARI 1244. ARI 1245. ARI 1246. ARI 1247. ARI 1248. ARI 1249. ARI		

Modul *Payment* seperti pada gambar 3.25 diuji melalui alur berjenjang yang melibatkan beberapa *role*. Proses *positive case* dimulai dari *Purchasing Staff* yang membuat *payment voucher*, dilanjutkan dengan persetujuan oleh *Purchasing Head*. Setelah disetujui, *Purchasing Staff* membuat *handover payment* yang kemudian disetujui oleh *Finance Staff*. *Payment voucher* tersebut lalu masuk ke *Finance Manager* untuk dibuatkan *transaction list*, dan *transaction list* ini disetujui oleh *Finance Director*. Proses diakhiri dengan *Finance Staff* yang mengunggah *proof of payment* dari *transaction list* tersebut, yang kemudian diterima oleh *Purchasing Staff*. Jika terdapat *voucher type* yang menggunakan dokumen *hard copy*, *Purchasing Staff* wajib membuat *additional handover* yang disetujui oleh *Finance Staff*.

Secara keseluruhan, 24 *Test Planning* telah disusun untuk seluruh alur modul *Payment*. Tahap pembuatan *payment voucher* diuji melalui empat *test planning* untuk kondisi *PO local* dan *import*. Tahap persetujuan oleh *Purchasing Head* diuji dengan empat skenario *approve* dan empat skenario *reject*. Selanjutnya, satu *test planning* disiapkan untuk pembuatan *handover*, satu untuk *approve handover* oleh *Finance Staff*, dan satu untuk *reject handover* oleh *Finance Staff*. Alur *transaction list* diuji dengan satu skenario *create* oleh *Finance Manager*, satu skenario *approve* oleh *Finance Director*, dan satu skenario *reject* oleh *Finance Director*. Empat *test case* dibuat untuk proses *upload proof of payment* oleh *Finance Staff*. Terakhir, proses *additional handover* diuji dengan satu skenario *submit* oleh *Purchasing Staff*, satu skenario *approve*, dan satu skenario *reject* oleh *Finance Staff*.

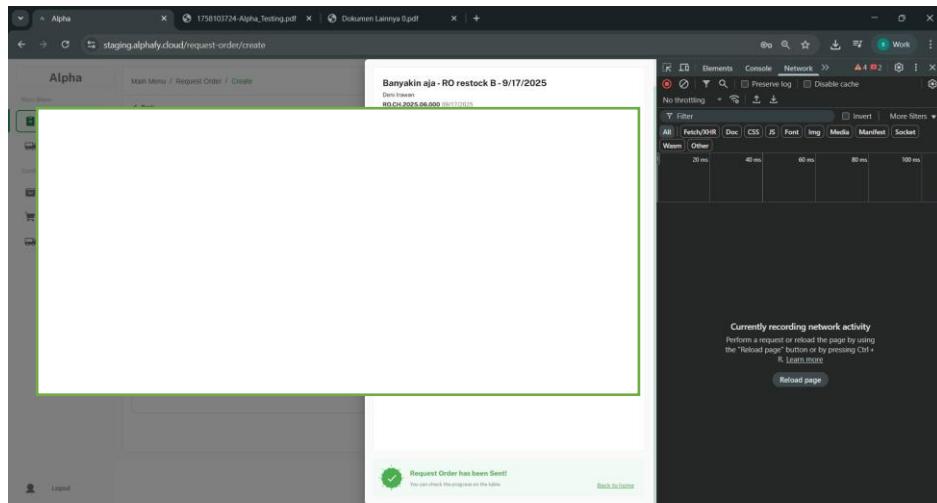
No	Naming	Category	Automate	Role	Apps	Modul	SubModul	Test Scenario	Action	Duration (Second)	Expected Result
62	TP - Alpha - 62	Functional Test	No	Warehouse	Web	Good Receipt	Create Good Receipt - Local type	7. A to P 1. C 2. L 3. A 4. S 'Create' 5. A 6. A 7. A 8. A 9. A 10. A 11. A 12. A 13. A 14. A 15. A 16. A 17. A 18. A 19. A 20. A 21. A 22. A 23. A 24. A 25. A 26. A 27. A 28. A 29. A 30. A 31. A 32. A 33. A 34. A 35. A 36. A 37. A 38. A 39. A 40. A 41. A 42. A 43. A 44. A 45. A 46. A 47. A 48. A 49. A 50. A 51. A 52. A 53. A 54. A 55. A 56. A 57. A 58. A 59. A 60. A 61. A 62. A 63. A 64. A 65. A 66. A	click Back		
63	TP - Alpha - 63	Functional Test	No	Warehouse	Web	Good Receipt	Create Good Receipt - Import type	1. o 2. o 3. o 4. o 5. o 6. o 7. o 8. o 9. o 10. o 11. o 12. o 13. o 14. o 15. o 16. o 17. o 18. o 19. o 20. o 21. o 22. o 23. o 24. o 25. o 26. o 27. o 28. o 29. o 30. o 31. o 32. o 33. o 34. o 35. o 36. o 37. o 38. o 39. o 40. o 41. o 42. o 43. o 44. o 45. o 46. o 47. o 48. o 49. o 50. o 51. o 52. o 53. o 54. o 55. o 56. o 57. o 58. o 59. o 60. o 61. o 62. o 63. o 64. o 65. o 66. o	click to click to	101 ms	User can create good receipt for local delivery type
64	TP - Alpha - 64	Functional Test	No	Supplier	Web	Request Order	Delete Rejection	1. o 2. o 3. o 4. o 5. o 6. o 7. o 8. o 9. o 10. o 11. o 12. o 13. o 14. o 15. o 16. o 17. o 18. o 19. o 20. o 21. o 22. o 23. o 24. o 25. o 26. o 27. o 28. o 29. o 30. o 31. o 32. o 33. o 34. o 35. o 36. o 37. o 38. o 39. o 40. o 41. o 42. o 43. o 44. o 45. o 46. o 47. o 48. o 49. o 50. o 51. o 52. o 53. o 54. o 55. o 56. o 57. o 58. o 59. o 60. o 61. o 62. o 63. o 64. o 65. o 66. o	click to	73 ms	User can create good receipt for import delivery type
65	TP - Alpha - 65	Functional Test	No	Head Dept of Purchase	Web	Purchase Order	Send Email	1. o 2. o 3. o 4. o 5. o 6. o 7. o 8. o 9. o 10. o 11. o 12. o 13. o 14. o 15. o 16. o 17. o 18. o 19. o 20. o 21. o 22. o 23. o 24. o 25. o 26. o 27. o 28. o 29. o 30. o 31. o 32. o 33. o 34. o 35. o 36. o 37. o 38. o 39. o 40. o 41. o 42. o 43. o 44. o 45. o 46. o 47. o 48. o 49. o 50. o 51. o 52. o 53. o 54. o 55. o 56. o 57. o 58. o 59. o 60. o 61. o 62. o 63. o 64. o 65. o 66. o	if button	5.16 s	User can sending email to Vendor after PO Approved
66	TP - Alpha - 66	Functional Test	No	Purchasing	Web	Purchase Order	Create Purchase Order	1. o 2. o 3. o 4. o 5. o 6. o 7. o 8. o 9. o 10. o 11. o 12. o 13. o 14. o 15. o 16. o 17. o 18. o 19. o 20. o 21. o 22. o 23. o 24. o 25. o 26. o 27. o 28. o 29. o 30. o 31. o 32. o 33. o 34. o 35. o 36. o 37. o 38. o 39. o 40. o 41. o 42. o 43. o 44. o 45. o 46. o 47. o 48. o 49. o 50. o 51. o 52. o 53. o 54. o 55. o 56. o 57. o 58. o 59. o 60. o 61. o 62. o 63. o 64. o 65. o 66. o	call the button Confirm'	107 ms	User can create more than one PO (bulk)

Gambar 3. 26 Potongan Test Planning GR

Modul GR dibuat oleh *role Warehouse* ketika DO yang telah dibuat oleh Purchasing Staff sudah tersedia. Test planning dirancang untuk mencakup seluruh variasi alur pengiriman DO seperti pada gambar 3.26, di mana lima test case disiapkan untuk kondisi DO dari pengiriman *local* dengan pilihan menggunakan ekspedisi, *pick-up*, dan *supplier driver* serta pengiriman *import* dengan pilihan menggunakan *sea* dan *air*.

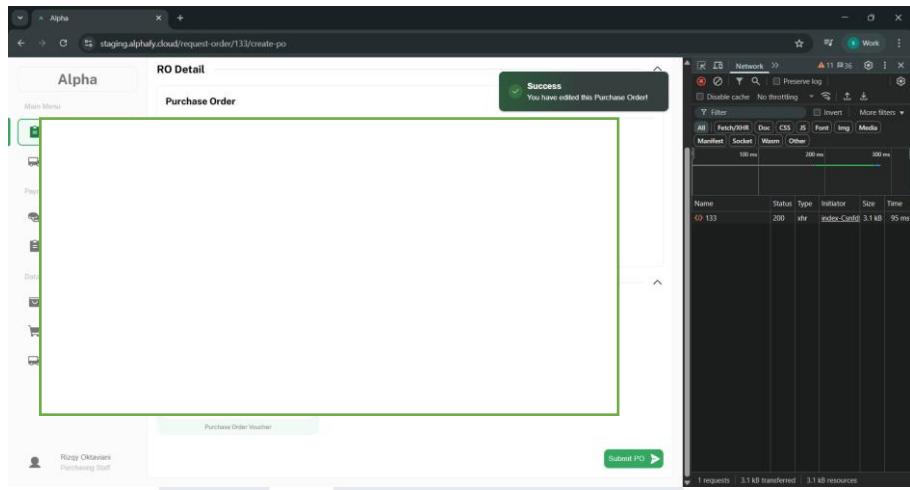
Modul GR menjadi modul terakhir dalam test planning karena menjadi tahap terakhir dalam flow *procurement*. Total test planning yang telah dibuat adalah sebanyak 153 rencana. Total ini mencakup seluruh role yang bertanggung jawab pada setiap tahap, seperti *supplier planner*, *procurement staff*, *procurement head*, *finance staff*, *finance manager*, *finance director* dan *warehouse*. Dengan total lima modul utama yaitu RO, PO, DO, *Payment* dan GR.

3.3.1.10 Pelaksanaan Testing modul *Request* dan *Approve RO, PO, dan DO*.



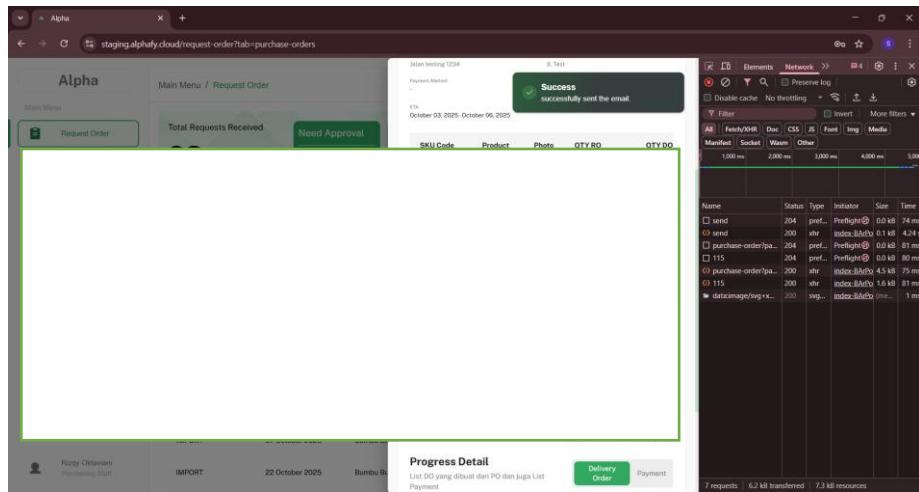
Gambar 3. 27 Contoh Hasil *Testing RO*

Pengujian fungsional modul RO dimulai dengan skenario pembuatan RO oleh *Supplier Planner* seperti pada gambar 3.27. Pengujian ini dilakukan untuk memvalidasi alur pengiriman data RO yang mencakup detail produk (*SKU Code*), jumlah pesanan, dan dokumen pendukung yang diunggah. Hasil pengujian yang diamati menunjukkan bahwa ketika RO telah diajukan, *end-point* yang menangani pembuatan RO merespons dengan kode status yang ideal. Status ini mengindikasikan bahwa permintaan *request* yang berisi seluruh detail RO berhasil diterima dan diproses oleh server tanpa adanya kegagalan teknis. Notifikasi “*Request Order has been sent!*” ditampilkan sebagai konfirmasi bahwa data RO telah berhasil tersimpan dan dialihkan ke tahap proses berikutnya. Validasi ini memastikan bahwa *Supplier Planner* dapat mengajukan permintaan order secara efisien sesuai alur yang ditetapkan.



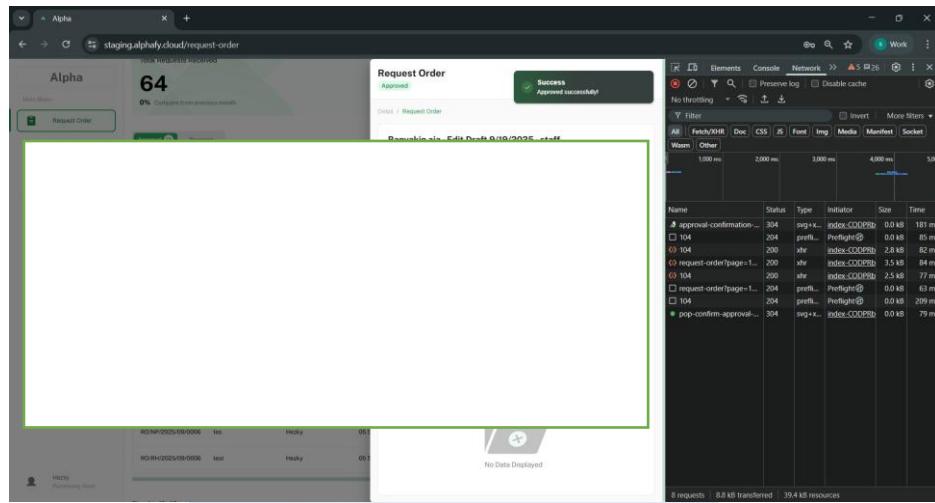
Gambar 3. 28 Contoh Hasil *Testing PO*

Pengujian fungsional modul PO dilanjutkan dengan skenario pembuatan PO oleh *Purchasing Staff*. Pembuatan PO dilakukan berdasarkan RO yang telah disetujui, seperti yang terlihat dari detail RO *New Order 2 (Import)* yang ditampilkan. Pengujian ini bertujuan untuk memvalidasi alur pembuatan PO seperti pada gambar 3.28, termasuk pemrosesan data *Supplier*, nominal transaksi, dan status PO baru. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani pembuatan PO merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* untuk membuat PO telah berhasil diproses oleh server tanpa kegagalan. Waktu respons yang tercatat adalah 95 milidetik (ms), mengindikasikan bahwa kinerja *end-point* sangat efisien dan responsif dalam memproses *request* yang kompleks. Setelah PO berhasil dibuat, notifikasi “*Success. You have edited the Purchase Order!*” ditampilkan, mengonfirmasi bahwa PO telah tersimpan dan siap dialihkan ke tahap persetujuan berikutnya.



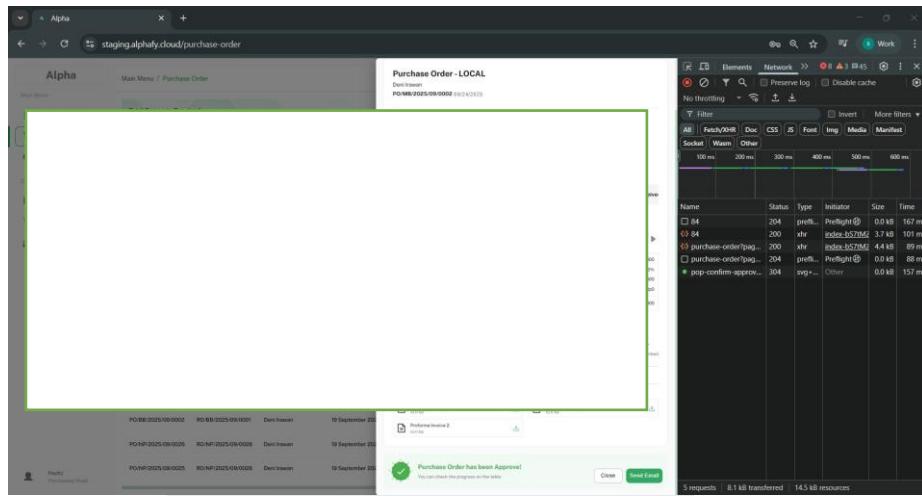
Gambar 3. 29 Contoh Hasil Testing DO

Pengujian fungsional modul DO dilanjutkan dengan skenario pembuatan DO oleh *Purchasing Staff*. Pembuatan DO ini seperti pada gambar 3.29 merupakan tindak lanjut dari PO yang telah disetujui, dan bertujuan untuk memproses pengiriman barang dari pemasok ke gudang. Hasil pengujian yang diamati menunjukkan bahwa *end-point* utama yang menangani pengiriman data DO merespons dengan Status Code 200 (OK). Kode status tersebut memverifikasi bahwa *request* untuk membuat DO telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat bervariasi antara 45 hingga 74 milidetik (ms) untuk berbagai *end-point* terkait (*purchase-order*, *send email*). Kinerja yang cepat ini mengindikasikan bahwa sistem mampu memproses pembuatan dokumen DO dan mengirimkan notifikasi terkait secara efisien. Setelah proses selesai, notifikasi “*Success. successfully sent the email.*” ditampilkan, mengonfirmasi bahwa DO telah tersimpan di basis data dan notifikasi pengiriman telah berhasil dikirimkan.



Gambar 3. 30 Contoh Hasil *Testing Approve RO*

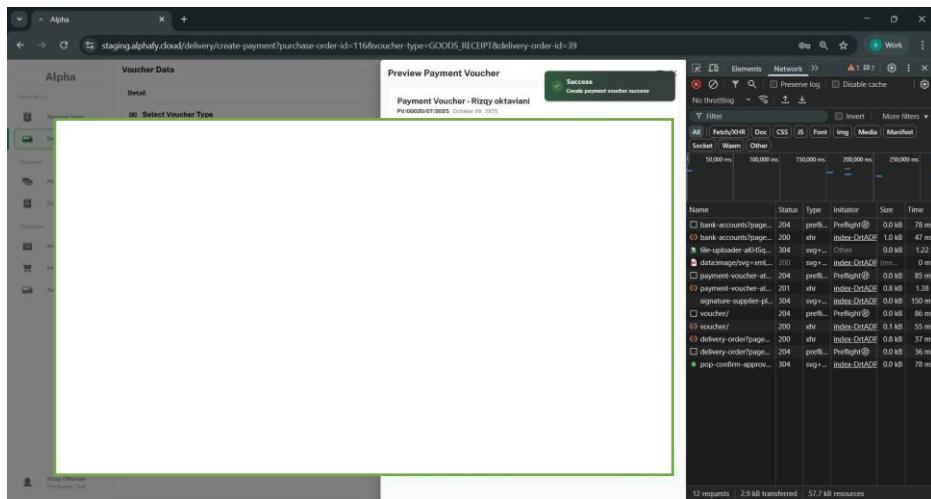
Pengujian fungsional modul RO dilanjutkan dengan skenario *approval* RO oleh *Procurement Head*. Pengujian ini bertujuan untuk memvalidasi alur persetujuan akhir RO seperti pada gambar 3.30, yang akan mengubah status dokumen menjadi *Approved* dan memungkinkan proses dilanjutkan ke pembuatan PO. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani aksi *approval* merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* persetujuan RO telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat bervariasi antara 84 milidetik (ms) untuk *end-point* yang terkait dengan pembaruan status dan konfirmasi persetujuan. Kinerja yang efisien ini mengindikasikan bahwa sistem mampu memproses pembaruan status RO dengan cepat dan responsif. Setelah persetujuan berhasil dilakukan, notifikasi “*Success. Approved successfully!*” ditampilkan, mengonfirmasi bahwa status RO telah berhasil diperbarui dan dialihkan ke tahap proses berikutnya di basis data.



Gambar 3. 31 Contoh Hasil Testing Approve PO

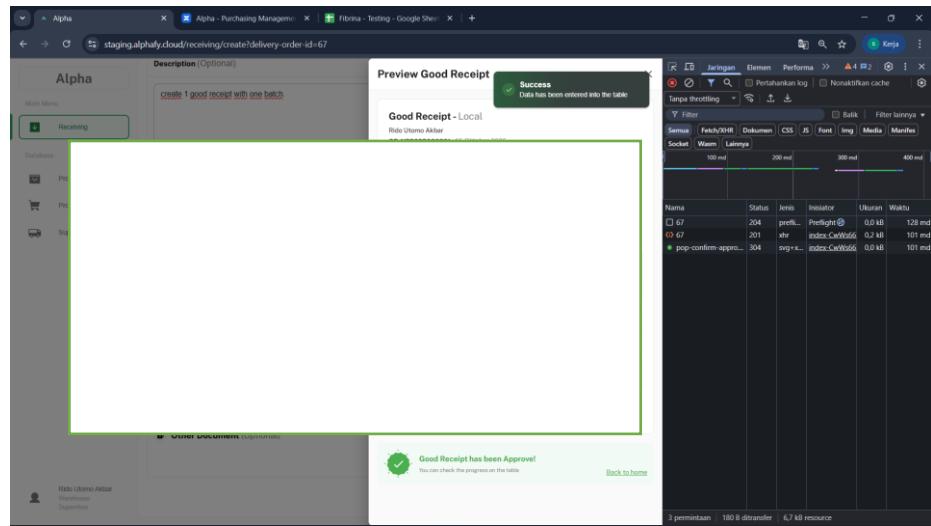
Pengujian fungsional modul PO dilanjutkan dengan skenario *approval* PO oleh *Procurement Head*. Pengujian ini bertujuan untuk memvalidasi alur persetujuan akhir PO, yang akan mengubah status dokumen menjadi *Approved* dan memungkinkan proses dilanjutkan ke pembuatan DO. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani aksi *approval* merespons dengan Status Code 200 (OK). Kode status tersebut memverifikasi bahwa *request* persetujuan PO telah berhasil diproses oleh server tanpa adanya kesalahan teknis seperti pada gambar 3.31. Waktu respons yang tercatat bervariasi antara 89 milidetik (ms) untuk *end-point* yang terkait dengan pembaruan status dan pengiriman notifikasi, menunjukkan kinerja *end-point* yang efisien dan responsif dalam memproses pembaruan status data. Setelah persetujuan berhasil dilakukan, notifikasi “*Purchase Order has been Approved!*” ditampilkan, mengonfirmasi bahwa status PO telah berhasil diperbarui dan notifikasi *email* telah berhasil dikirimkan.

3.3.1.11 Pelaksanaan Testing modul *GR* dan *Payment*



Gambar 3. 32 Contoh Hasil Testing Payment

Pengujian fungsional modul Payment dimulai dengan skenario pembuatan *Payment Voucher* oleh *Purchasing Staff*. Pengujian ini bertujuan untuk memvalidasi alur pembuatan dokumen pembayaran, termasuk pemrosesan *Voucher Type*, detail *Beneficiary*, dan dokumen faktur yang terkait. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani pembuatan *Payment Voucher* merespons dengan Status Code 200 (OK) seperti pada gambar 3.32. Kode status tersebut memverifikasi bahwa *request* untuk membuat *voucher* telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat menunjukkan durasi yang efisien, dengan *end-point* utama *payment-voucher* merespons dalam 78 milidetik (ms), sementara *end-point* konfirmasi persetujuan merespons dalam 37 milidetik (ms). Kinerja yang cepat ini mengindikasikan bahwa sistem mampu memproses pembuatan dokumen pembayaran yang kompleks secara responsif. Setelah proses selesai, notifikasi “Success. Create payment voucher success” dan “Payment voucher has been sent!” ditampilkan, mengonfirmasi bahwa *voucher* telah tersimpan dan berhasil dikirimkan ke alur persetujuan berikutnya.



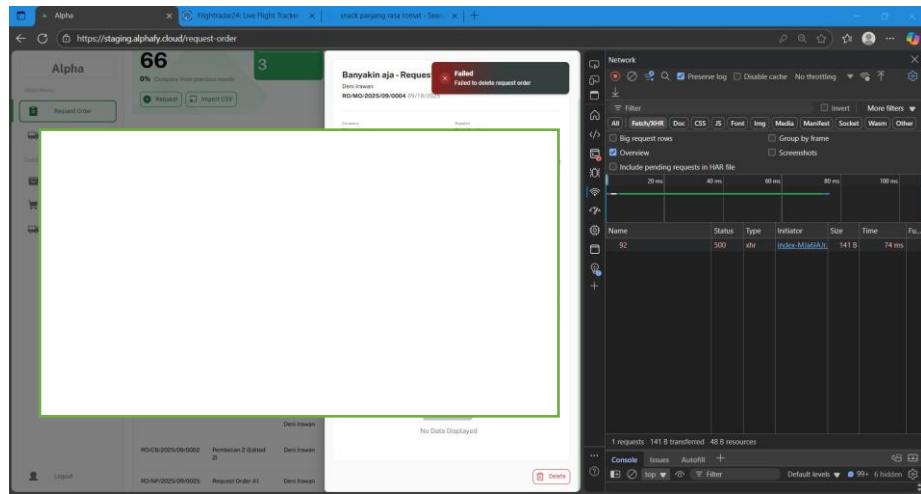
Gambar 3. 33 Contoh Hasil Testing Good Receipt

Pengujian fungsional modul GR dilaksanakan oleh peran *Warehouse* setelah DO tersedia. Pengujian ini bertujuan untuk memvalidasi kemampuan sistem dalam memproses penerimaan barang ke gudang dan pembaruan inventaris. Skenario yang diuji, seperti terlihat pada gambar, adalah pembuatan *good receipt* dengan satu *batch* (*create 1 good receipt with one batch*) seperti pada gambar 3.33. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani pembuatan GR merespons dengan Status Code 200 (OK). Kode status tersebut memverifikasi bahwa *request* untuk membuat GR telah berhasil diproses oleh *server* tanpa adanya kegagalan teknis. Waktu respons yang tercatat menunjukkan durasi yang efisien, dengan *end-point* utama *receiving/create* merespons dalam waktu sekitar 101 milidetik (ms). Kinerja yang cepat ini memastikan efisiensi dalam alur kerja *Warehouse*. Setelah proses selesai, notifikasi “*Data has been entered into the table*” dan “*Good Receipt has been Approved!*” ditampilkan, mengonfirmasi bahwa data penerimaan barang telah tersimpan dan telah melalui alur persetujuan.

Test Case ID	Modul	Sub Modul	Log	Log Type	Status	Raise Date	Pic Test	Pic Dev	End Date	Attachment	Dev Note
2 TP - Alpha - 04 - 01	Request Order	Create RO - Local	wrong placement - check figma	Minor	Done	9/17/2025 - 5:10 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 04 - 01.jpg	
3 TP - Alpha - 04 - 02	Request Order	Create RO - Local	The successfully updated message should be displayed when the user click the 'Save' button and above the 'Continue' button within the pop-up	Minor	Done	9/17/2025 - 5:10 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 04 - 02.jpg	
4 TP - Alpha - 04 - 03	Request Order	Create RO - Local	After selecting one of the RO created, click on the 'Delete' button and then click on the 'Delete' button again. The system should display a confirmation message before deleting the RO.	Minor	Done	9/17/2025 - 5:10 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 04 - 03.jpg	
5 TP - Alpha - 08 - 01	Request Order	Delete RO - Cancellation	Can't delete RO from cancelled status RO. The toast message should display the cancelled status.	Minor	Done	9/18/2025 - 2:59 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 07 - 01.jpg	
6 TP - Alpha - 07 - 01	Request Order	Cancel RO	The toast message should display the cancelled status.	Minor	Done	9/19/2025 - 10:10 AM	Sabrina Azmi	Ready		Cancel toast message - RO.jpg	
7 TP - Alpha - 28 - 01	Approve RO	Delete RO - Cancellation	Can't edit RO from cancelled status RO.	Minor	Done	9/19/2025 - 4:15 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 12 - 01.jpg	
8 TP - Alpha - 19 - 01	Request Order	Export CSV	Button Export CSV doesn't work.	Minor	To Do	9/20/2025 - 11:30 AM	Sabrina Azmi	Ready		Bug - TP - Alpha - 10 - 01.jpg	
9 TP - Alpha - 12 - 01	Request Order	Sort by Company	After selecting one of the RO created, click on the 'Delete' button and then click on the 'Delete' button again. The system should display a confirmation message before deleting the RO.	Minor	Done	9/20/2025 - 11:30 AM	Sabrina Azmi	Ready		Bug - TP - Alpha - 11 - 01.jpg	
10 TP - Alpha - 22 - 01	Request Order	Selected by name	After selecting one of the requestor name, the result is not correct.	Minor	To Do	9/23/2025 - 1:45 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 22 - 01.jpg	
11 TP - Alpha - 26 - 01	Approve RO	Reject notes	When the user click on the 'Reject' button, the reason should show and then click on the 'Save' button.	Minor	Done	9/22/2025 - 3:45 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 26 - 01.jpg	
12 TP - Alpha - 23 - 01	List Overview	Selected by status	Should be hyperlinks.	Minor	Done	9/23/2025 - 2:45 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 23 - 01.jpg	
13 TP - Alpha - 03 - 01	Request Order	Import CSV	Should be hyperlinks.	Minor	To Do	9/23/2025 - 2:45 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 03 - 01.jpg	
14 TP - Alpha - 64 - 01	Request Order	Delete RO - Rejection	Can't edit RO from rejected status RO. error 400, response only can delete RO from rejected status.	Minor	To Do	9/23/2025 - 3:05 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 54 - 01.jpg	
15 TP - Alpha - 08 - 02	Request Order	Delete RO - Cancellation	Can't delete RO from cancelled status RO. 400 error.	Minor	Done	9/23/2025 - 4:20 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 08 - 02.jpg	
16 TP - Alpha - 11 - 01	List Overview	Sort by date - range menu	Data shown in 1 from date selected	Minor	To Do	9/23/2025 - 4:30 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 11 - 01.jpg	
17 TP - Alpha - 17 - 01	List Overview	Sort by date - Oldest to Newest	When the date filter is applied and saved, the data displayed in the table does not change.	Minor	To Do	9/23/2025 - 4:30 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 17 - 01.jpg	
18 TP - Alpha - 16 - 01	List Overview	Sort by branch	When the date filter is applied and saved, the data displayed in the table does not change.	Minor	To Do	9/23/2025 - 4:55 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 16 - 01.jpg	
19 TP - Alpha - 26 - 02	Request Order	Reaction detail preview	no need delete and revise button after RO	Minor	Done	9/24/2025 - 1:35 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 26 - 02.jpg	
20 TP - Alpha - 94 - 03	Request Order	Create RO - Local	At previous PO, column product should be name of product, not the vector. Ital label, cant create PO if quantity more than RO quantity should be shown.	Minor	Done	9/24/2025 - 3:00 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 94 - 03.jpg	
21 TP - Alpha - 48 - 01	Purchase Order	Create PO - Local	At previous PO error SKU Code, Product, and quantity. It should be quantity and not the vector. signature of purchasing staff	Minor	Done	9/24/2025 - 3:05 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 48 - 01.jpg	
22 TP - Alpha - 48 - 02	Purchase Order	Create PO - Local	At previous PO error SKU Code, Product, and quantity. It should be quantity and not the vector. signature of purchasing staff	Minor	Done	9/24/2025 - 3:05 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 48 - 02.jpg	
23 TP - Alpha - 48 - 03	Purchase Order	Create PO - local	error design at create PO - check figma, submit	Minor	Pending	9/24/2025 - 3:45 PM	Sabrina Azmi	Ready		Bug - TP - Alpha - 48 - 03.jpg	

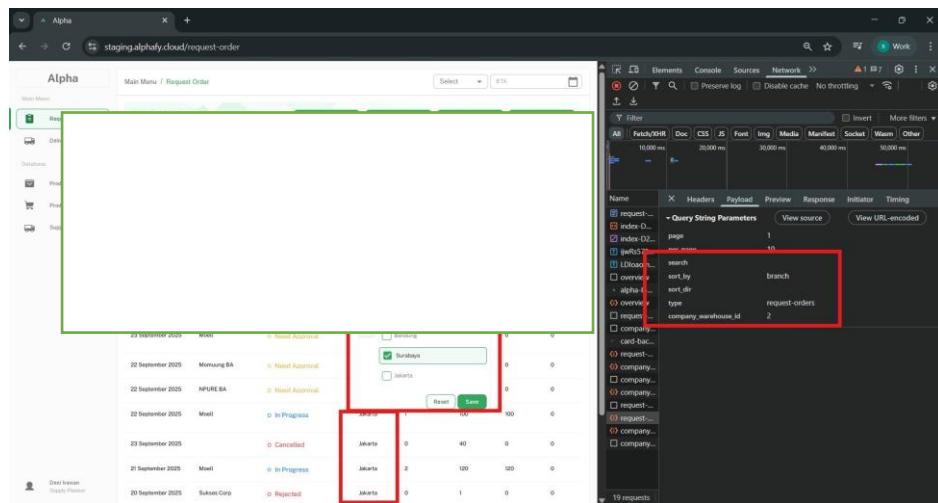
Gambar 3. 34 Hasil *Test Log Alpha*

Dalam pengujian yang telah dilakukan mulai dari modul RO hingga *payment* yang melibatkan seluruh role. Terdapat banyak temuan *error* atau *bug* yang muncul. Hal ini ditemukan bukan hanya dari pengujian berdasarkan interface tetapi juga pengujian data seperti pada gambar 3.34. Pengujian *interface* dilihat dengan acuan dari design yang telah dibuat oleh *UI/UX Designer* pada Figma seperti Gambar 3.19, menguji apakah setiap komponen sudah sesuai, tidak ada komponen yang hilang atau delay dalam penampilan data. Sedangkan pengujian data mengacu pada pelemparan data yang di input dan hasil yang diberikan, apakah data yang dimasukan dan di tampilkan sesuai, pengujian ini dilihat dari hasil response pada *end-point* yang diberikan. Pada proyek Alpha terdapat 64 bug yang ditemukan, mulai dari *bug minor* hingga *major*.



Gambar 3. 35 Contoh *Bug Major*

Pada gambar 3.35 merupakan contoh *bug major* yang ditemukan ketika ingin menguji *cancel RO* menyebabkan *blocking* sehingga pengujian tidak dapat dilanjutkan ke tahap berikutnya. Hal ini dapat dilihat dari status yang dikeluarkan oleh end-point berupa 500 dengan permasalahan pada backend yang belum membuat *development* untuk *cancel RO*.



Gambar 3. 36 Contoh *Bug Minor*

Selain *bug major* yang ditemukan sebelumnya, terdapat pula temuan *bug minor*, seperti yang gambar 3.36. *Bug* jenis ini biasanya muncul tanpa mengakibatkan *blocking* pada alur kerja utama

pengguna. Pada gambar 3.36 menunjukkan adanya *error* saat pelemparan data, *action* yang dilakukan tidak memberikan hasil yang sesuai meskipun response 200. Jika hal ini terjadi, perlu diperhatikan *payload* dari end-point yang ada untuk mengidentifikasi permasalahan yang ada.

3.3.2 Kendala yang Ditemukan

Selama proses praktik kerja magang sebagai QA *Intern* di PT. Sukses Corp International, ditemukan beberapa tantangan dan kendala yang menjadi bagian penting dari proses adaptasi dan pembelajaran. Kendala-kendala ini diatasi melalui inisiatif pribadi, komunikasi aktif, dan adaptasi cepat terhadap lingkungan kerja yang dinamis.

1. Tantangan Struktur Organisasi dan Pembimbingan Fungsional

Kendala utama teridentifikasi dari struktur tim yang unik, di mana posisi yang diemban adalah satu-satunya peran QA *Intern*. Karena tidak tersedianya atasan atau senior langsung dengan peran QA yang serupa dalam tim, acuan pembimbingan fungsional harian yang spesifik dari seorang QA senior tidak dimiliki. Konsekuensi dari situasi ini adalah diharuskannya pembentukan standar kerja secara mandiri. Hal tersebut menuntut adanya inisiatif yang tinggi dan kemampuan pembelajaran mandiri yang kuat untuk setiap pengambilan keputusan teknis yang berkaitan dengan metodologi pengujian, prosedur pelaporan *bug*, dan dokumentasi.

2. Keterbatasan Dokumentasi dan Standar Pengujian

Kesulitan signifikan dihadapi pada fase awal proses pengujian karena ketiadaan dokumentasi pengujian (*testing documentation*) yang terstruktur dari proyek-proyek terdahulu. Selain itu, standar pengujian baku yang jelas untuk QA, seperti format penulisan *test case* atau kriteria penentuan tingkat keparahan *bug*, belum ditetapkan secara formal. Kondisi ini memperpanjang waktu adaptasi yang diperlukan pada proses pengujian

di tahap awal. Oleh sebab itu, serangkaian diskusi intensif perlu dilakukan bersama *UI/UX Designer* dan tim *Engineer* untuk mendefinisikan serta menyepakati mekanisme pengujian yang paling efektif untuk aplikasi Fibrina dan Alpha.

3. Adaptasi Awal dan Kurva Pembelajaran Teknis

Kurva pembelajaran yang signifikan teridentifikasi selama periode magang, mengingat bahwa peran ini merupakan pengalaman pertama sebagai QA profesional. Proses adaptasi yang diperlukan mencakup pemahaman mendalam mengenai alur bisnis perusahaan yang kompleks, khususnya terkait produk Fibrina dan Alpha, dan penguasaan alat bantu pengujian. Secara spesifik, penguasaan alat ini meliputi keterampilan dalam *scripting* dan implementasi otomatisasi pengujian menggunakan Katalon Studio Enterprise. Kondisi tersebut menuntut adanya upaya proaktif dalam pencarian sumber daya belajar dan penyediaan waktu tambahan untuk menguasai keterampilan teknis yang dibutuhkan secara mandiri.

4. Isu Koordinasi Lintas Divisi

Dalam kolaborasi tim yang intensif, komunikasi yang kurang efektif terjadi beberapa kali di antara pihak *UI/UX Designer*, *Back-End Developer*, dan *Front-End Developer*. Ketidakselarasan komunikasi ini umumnya berkaitan dengan perbedaan interpretasi terhadap spesifikasi fungsional atau alur desain yang telah ditetapkan. Hal ini kemudian berakibat pada terhambatnya alur pengujian yang sedang dilaksanakan. Untuk memitigasi isu tersebut, inisiatif diambil untuk memverifikasi ulang alur yang terindikasi janggal dengan *Designer* sebelum setiap *bug* dilaporkan kepada tim *Developer*. Upaya verifikasi ini berfungsi untuk memastikan bahwa masalah yang disampaikan telah divalidasi berdasarkan spesifikasi desain yang akurat.

3.3.3 Solusi atas Kendala yang Ditemukan

Setiap kendala yang dihadapi selama praktik kerja magang dijadikan peluang untuk menunjukkan inisiatif, adaptasi, dan tanggung jawab. Berikut adalah solusi proaktif yang diterapkan untuk mengatasi tantangan yang muncul:

1. Mengambil Alih Pembelajaran Mandiri

Pelaksanaan pembelajaran mandiri dilakukan secara aktif terkait sistematika dan *testing flow*, serta penyesuaianya dengan kebutuhan aplikasi internal perusahaan. Kegiatan yang dijalankan mencakup pendalaman penguasaan alat bantu otomatisasi pengujian. Upaya ini dilaksanakan untuk mengatasi tantangan yang timbul akibat tidak adanya atasan langsung dengan *role* yang serupa di dalam tim, sehingga berkontribusi pada percepatan proses adaptasi teknis yang dibutuhkan.

2. Menciptakan Dokumentasi Standar Perusahaan

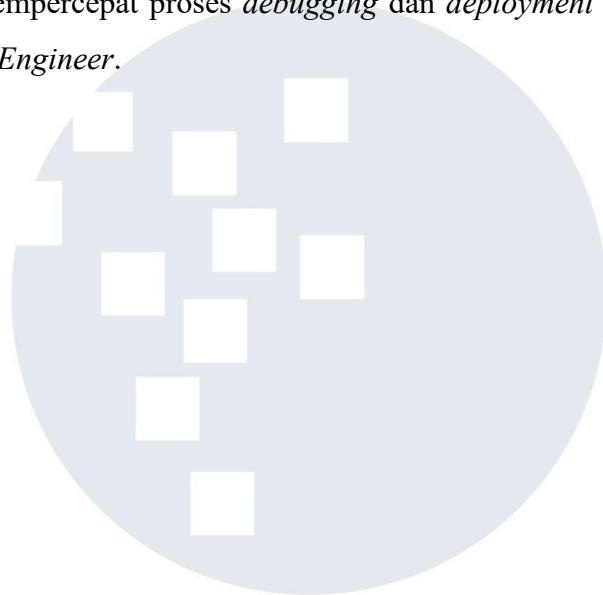
Inisiatif penyusunan dokumentasi yang terstruktur dilakukan guna mengatasi ketiadaan dokumentasi pengujian sebelumnya. Dokumentasi yang dihasilkan mencakup *Test Plan*, *Test Skenario*, dan *Test Log* yang rinci untuk setiap modul yang diuji. Dokumentasi yang telah dibuat tersebut kemudian berfungsi sebagai acuan standar pengujian resmi yang dapat digunakan oleh QA di masa mendatang.

3. Mengoptimalkan Komunikasi Fungsional

Perbaikan dilakukan pada mekanisme komunikasi dengan tim *Designer* dan *Developer* dengan tujuan meminimalkan potensi miskomunikasi. Perbaikan tersebut diimplementasikan melalui prosedur verifikasi ulang alur yang terindikasi jangkal dengan *Designer* sebelum setiap *bug* dilaporkan kepada *Developer*. Prosedur ini dilaksanakan untuk memastikan bahwa setiap masalah yang dilaporkan telah divalidasi dan sesuai dengan spesifikasi yang benar.

4. Memperbaiki Efektivitas Pelaporan

Perbaikan kualitas pada pelaporan *bug* menjadi fokus utama. Hal ini diwujudkan dengan memastikan bahwa setiap laporan yang disampaikan kepada *Back-End Developer* atau *Front-End Developer* memiliki sasaran yang tepat, serta dilengkapi dengan langkah reproduksi dan bukti pendukung yang jelas. Peningkatan kualitas pelaporan tersebut ditujukan untuk mempercepat proses *debugging* dan *deployment* yang dilaksanakan oleh tim *Engineer*.



UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA