

BAB III

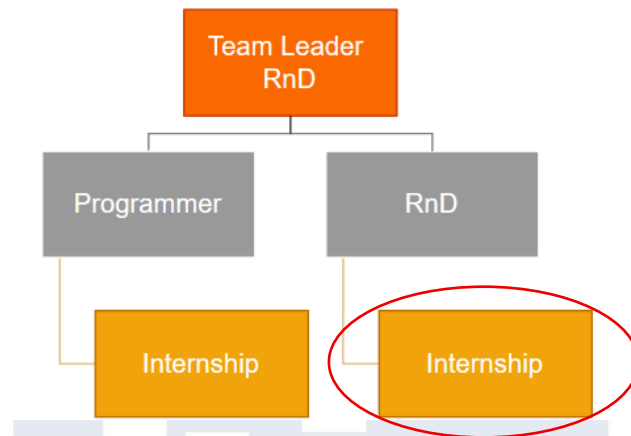
PELAKSANAAN KERJA

3.1 Kedudukan dan Koordinasi

3.1.1 Kedudukan

Selama melaksanakan magang sebagai Quality Assurance Engineer di PT Qtera Mandiri, peserta magang menempati kedudukan di bawah koordinasi langsung divisi Research and Development (RnD). Dalam struktur organisasi tersebut, peserta magang terlibat dalam alur kerja teknis pengujian sistem dan berada di bawah supervisi Team Leader RnD, serta bekerja sama dengan tim pengembang yang mencakup Frontend Developer, Backend Developer, dan pihak terkait lainnya. Kedudukan ini memungkinkan peserta magang untuk terlibat langsung dalam proses pengembangan sistem, khususnya pada tahap verifikasi dan validasi kualitas perangkat lunak. Berdasarkan kedudukan tersebut, peserta magang bertanggung jawab dalam melakukan pengujian fitur pada Website Asset Management sebagai salah satu produk utama perusahaan. Tanggung jawab yang dilaksanakan meliputi penyusunan test case, pelaksanaan pengujian manual, serta pengujian otomatis menggunakan Katalon Studio. Selain itu, peserta magang melakukan pencatatan dan pelaporan bug, serta berkoordinasi dengan supervisor dan tim pengembang untuk memastikan setiap proses pengujian berjalan sesuai dengan alur bisnis dan spesifikasi sistem yang telah ditetapkan. Dalam praktiknya, peserta magang melakukan koordinasi rutin dengan supervisor dan anggota tim lain yang terlibat dalam pengembangan proyek. Koordinasi ini penting dilakukan untuk memastikan setiap proses pengujian berjalan sejalan dengan rancangan awal, serta mencegah terjadinya kesalahan interpretasi atau penyimpangan dalam implementasi. Melalui komunikasi yang terstruktur dan kolaboratif, proses pengembangan dapat berjalan lebih efisien dan hasil pengujian dapat

mendukung kualitas produk secara optimal. Gambar 3.1 dibawah akan menggambarkan kedudukan dari peserta magang di PT. Qtera Mandiri

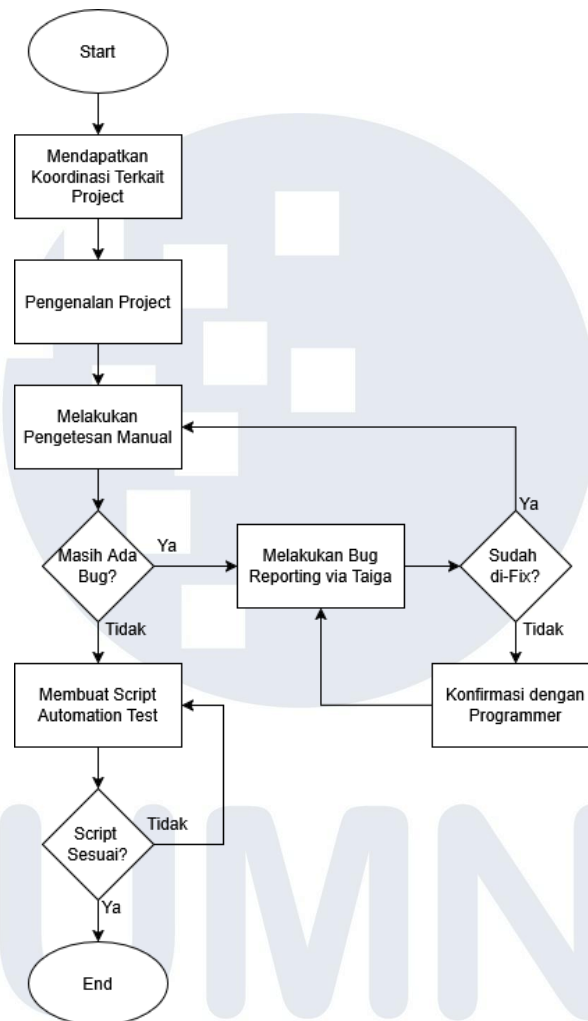


Gambar 3.1 Kedudukan Peserta Magang dalam Perusahaan

Struktur organisasi perusahaan menempatkan peserta magang *Quality Assurance Engineer* di bawah koordinasi langsung divisi *Research and Development (RnD)*. Divisi ini bertanggung jawab terhadap proses perancangan sistem yang dilaksanakan oleh tim *UI/UX Designer* yang beranggotakan 6 orang, serta proses pengujian sistem yang dilakukan oleh tim *Quality Assurance* yang terdiri dari 5 anggota, yaitu 3 karyawan tetap (*full-time employee*) dan 2 peserta magang. Kedudukan tersebut menempatkan peserta magang pada alur kerja teknis yang terarah dengan supervisi langsung dari *Team Leader RnD*, serta dukungan dari divisi *Programmer* yang terbagi menjadi beberapa tim, yaitu *Frontend Developer* yang terdiri dari 7 anggota, *Backend Developer* yang terdiri dari 9 anggota, *Mobile Application Developer* yang terdiri dari 6 anggota. Dalam pelaksanaan tugas, peserta magang berkomunikasi secara langsung dengan *Team Leader* dan tim *Programmer* untuk memastikan setiap proses pengujian berjalan sesuai standar serta selaras dengan kebutuhan proyek. Interaksi tersebut mencakup penyelarasan kebutuhan pengujian, verifikasi hasil pengembangan serta diskusi mengenai temuan dan penyesuaian yang diperlukan.

3.1.2 Koordinasi

Selain itu, peserta magang turut mengikuti alur kerja yang berlaku dalam divisi seperti yang digambarkan pada gambar 3.2 dibawah.



Gambar 3.2 Alur Kerja Magang dalam Perusahaan

Tahap pertama dalam proses pengujian adalah berkoordinasi dan berkonsultasi dengan *supervisor* untuk memastikan bahwa peserta magang dapat mengerti alur bisnis dan penugasan yang diberikan. Peserta magang mendapatkan bantuan mengenai tujuan pengembangan, fitur yang sedang dikerjakan, dan urutan pengujian yang akan dilakukan oleh peserta magang. Tahap ini dilakukan oleh *Team Leader* dengan cara membagikan penugasan kepada beberapa anggota tim secara per modul, dimana setiap modulnya

akan memiliki penanggungjawabnya masing-masing dalam tim *Quality Assurance* maupun *Developer*. Setelah penugasan diberikan, peserta magang akan mendapatkan penjelasan lebih detail tentang cara kerja fitur yang akan diuji dan menetapkan tahapan untuk pengujian manual. Tahap *testing manual* dilaksanakan menggunakan berbagai *testing scenario* yang telah dibuat oleh *team leader* yang disimpan dalam Google Spreadsheet terhadap setiap modul yang ada pada *website asset management*. Dalam pelaksanaan pengujian manual, apabila peserta magang menemukan kesalahan yang ternyata tidak sesuai dengan alur bisnis yang telah dibangun, peserta magang akan melakukan *bug reporting* kepada *programmer* yang mengerjakan modul atau fitur tersebut. Setelah tim *programmer* melakukan perubahan, peserta magang akan melakukan pemeriksaan terhadap perbaikan *bug* tersebut. Pemeriksaan ini sangat penting untuk memastikan bahwa masalah telah diperbaiki dan tidak mempengaruhi fungsi lainnya. Langkah selanjutnya adalah menggunakan teknologi otomasi untuk membuat skrip pengujian otomatis setelah semua pengujian manual dilakukan dan tidak ada yang diperbaiki. Peserta magang kemudian dapat memanfaatkan temuan dari skrip ini dalam pengujian selanjutnya karena didasarkan pada kasus pengujian yang telah ditentukan sebelumnya. Setelah skrip selesai dibuat, skrip akan di-*execute* untuk memastikan berfungsi dan hasilnya benar.

3.2 Tugas yang Dilakukan

Selaras dengan tanggungjawab yang diberikan dalam menjalani tugasnya sebagai *Quality Assurance Engineer*, peserta magang berpartisipasi aktif dalam 2 project PT. Qtera Mandiri. Project yang pertama adalah website penjualan internal perusahaan yang mencatat segala jenis pembelian dan penjualan yang dilakukan oleh perusahaan sebagai distributor dan juga partner dari beberapa perusahaan seperti Heimdal Security, Panda Security dan Watchguard. Peserta magang diminta untuk mengenali dan mempelajari konsep utama dari pelaksanaan tanggungjawab sebagai *Quality Assurance Engineer*. Pada project kedua, peserta magang diminta untuk berpartisipasi aktif dalam melakukan testing secara manual dan *automation*

terhadap *website asset management*, yang menjadi salah satu produk utama perusahaan. Setiap tahapan pengerjaan dari project akan disusun dan dijabarkan pada Tabel 3.1

Tabel 3.1 Uraian Kegiatan Selama Pelaksanaan Magang

No.	Kegiatan	Tanggal Mulai	Tanggal Selesai
1.	Masa pengenalan lingkungan kerja dan <i>project</i> kantor		
1a.	Pengenalan lingkungan kerja magang di PT. Qtera Mandiri	21 Juli 2025	21 Juli 2025
2.	Mulai pelaksanaan tugas <i>project</i> internal perusahaan		
2a.	Pengenalan <i>project</i> internal perusahaan	22 Juli 2025	22 Juli 2025
2b.	Membuat <i>test case</i> positif dan <i>Project Requirement Document</i> (PRD) untuk beberapa modul	22 Juli 2025	1 Agustus 2025
3.	Transisi ke <i>project website asset management</i>		
3a.	Perpindahan ke <i>development website asset management</i>	1 Agustus 2025	1 Agustus 2025
3b.	Pengenalan <i>project website asset management</i>	4 Agustus 2025	4 Agustus 2025
4.	Menjalankan <i>testing</i> untuk <i>website asset management</i> (<i>Basic Pricing</i>)		
4a.	Menjalankan <i>manual testing</i> untuk <i>basic pricing</i>	5 Agustus 2025	18 Agustus 2025
4b.	Menjalankan <i>automation testing</i> untuk <i>basic pricing</i>	19 Agustus 2025	19 September 2025
5.	Menjalankan <i>testing</i> untuk <i>website asset management</i> (<i>Enterprise Pricing</i>)		
5a.	Menjalankan <i>manual testing</i> untuk <i>enterprise pricing</i>	22 September 2025	9 Januari 2026
5b.	Menjalankan <i>automation testing</i> untuk <i>enterprise pricing</i>	1 Oktober 2025	30 Januari 2026

Tabel 3.1 menjelaskan timeline pelaksanaan tugas yang dilakukan oleh peserta magang dalam memenuhi tanggungjawabnya dalam PT. Qtera Mandiri. Tabel tersebut menunjukkan urutan penugasan yang dilakukan oleh peserta magang selama durasi pelaksanaan magang di PT. Qtera Mandiri, yang mencakup runtutan kegiatan yang dilaksanakan beserta dengan durasinya yang terpecah

menjadi tanggal mulai dan tanggal selesai dari setiap kegiatan. Tabel tersebut menggambarkan bahwa kegiatan paruh pertama merupakan pengenalan lingkungan kerja, dimana peserta belajar tentang budaya perusahaan, cara berbicara satu sama lain, dan aturan-aturan yang berlaku dalam pekerjaannya. Setelah peserta magang terbiasa dengan pekerjaan barunya dan mempelajari cara melakukannya, mereka diberikan proyek internal perusahaan untuk dikerjakan. Hal ini termasuk membuat kasus uji dan menyusun *Project Requirement Document* yang akan digunakan sebagai dasar pengembangan. Setelah peserta magang menyelesaikan tahap pembelajaran, mereka disuruh mengerjakan produk inti perusahaan yaitu *website Asset Management*. Pada titik ini, peserta mempelajari struktur aplikasi, fitur yang sedang dikerjakan, dan cara tim pengembangan bekerja sama. Sebagai bagian dari *Quality Assurance*, peserta memulai dengan menguji secara manual modul-modul yang ada di *Basic Pricing*. Kemudian mereka beralih ke pengujian otomatis untuk memastikan pengujian bekerja lebih lancar dan konsisten. Setelah modul yang ada pada *Basic Pricing* telah selesai di-*testing*, peserta magang melanjutkan proses serupa pada modul Enterprise Pricing, kembali melalui tahapan manual dan automation testing. Kegiatan tersebut mencakup seluruh tanggungjawab yang telah diberikan dan disepakati sebagai peserta magang *Quality Assurance Engineer* di PT. Qtera Mandiri.

3.3 Uraian Pelaksanaan Kerja

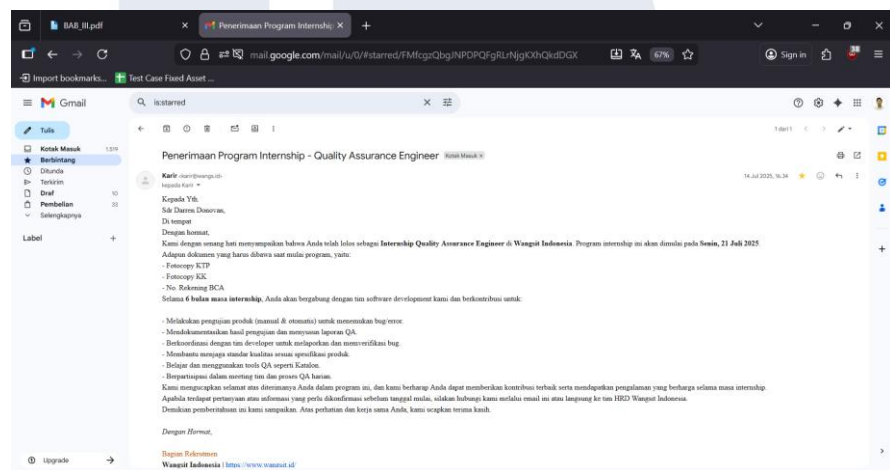
Berdasarkan runtutan kegiatan yang telah dijabarkan pada tabel 3.1, berikut adalah penjelasan detail dari masing-masing kegiatan yang dilakukan oleh peserta magang di PT. Qtera Mandiri.

3.3.1 Proses Pelaksanaan

Proses pelaksanaan magang yang dijalani peserta terdiri atas beberapa tahapan yang mencerminkan alur kegiatan dari awal hingga akhir masa magang. Setiap tahapan dilakukan secara bertahap sesuai arahan perusahaan dan kebutuhan divisi terkait. Bagian berikut menjelaskan rangkaian proses tersebut secara runtut mulai dari masa pengenalan lingkungan kerja hingga pelaksanaan *project*

3.3.1.1 Masa Pengenalan Lingkungan Kerja dan Project Kantor

Sebelum berlangsungnya kegiatan magang, peserta magang telah dihubungi terlebih dahulu melalui email pribadi oleh divisi *Human Resource* (HR) PT. Qtera Mandiri. Pada email tersebut, dilampirkan beberapa informasi terkait detail pekerjaan yang akan dilakukan oleh peserta magang selama 6 bulan masa kerja dan beberapa dokumen yang harus dibawa pada hari dimulainya kegiatan magang pada tanggal 21 Juli 2025. Detail dari email yang dikirimkan tersebut terlampir pada Gambar 3.3.



Gambar 3.3 *Letter of Acceptance* Peserta Magang

Sesampainya peserta magang pada lokasi kantor, peserta magang disambut oleh salah satu perwakilan HR yang bertanggung jawab atas program *internship*, bersama *Project Manager* dari tim yang akan menjadi tempat penugasan peserta magang. Pada sesi awal, peserta magang diberikan beberapa penjelasan umum terkait perusahaan, termasuk beberapa aturan yang harus ditaati oleh peserta magang serta ketentuan yang harus dipenuhi seperti perizinan. Penjelasan tersebut kemudian diakhiri dengan proses penandatanganan dokumen kontrak kerja magang. Setelah pengenalan awal, peserta magang diperkenalkan dengan lingkungan kantor, termasuk ruangan-ruangan yang ada beserta dengan beberapa divisi

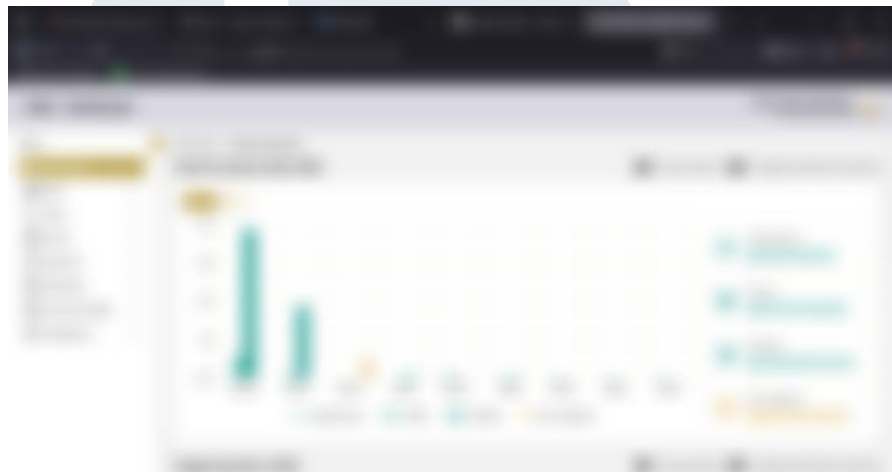
yang bertanggung jawab atas kegiatan operasional PT. Qtera Mandiri. Setelah itu, peserta magang kemudian diperkenalkan kembali dengan supervisor yang akan bertanggungjawab sebagai mentor dari peserta magang. Selanjutnya, peserta magang diperkenalkan dengan beberapa project yang sedang sedang berjalan di perusahaan, termasuk *project* internal perusahaan dan *website asset management*. Pada tahap ini, *supervisor* dan *Project Manager* memberikan penjelasan menyeluruh mengenai tujuan project, alur kerja pengembangan, teknologi yang digunakan serta peran setiap tim yang terlibat dalam proses tersebut. Peserta magang juga diajarkan mengenai tugas yang akan dilaksanakan oleh peserta magang, termasuk kontribusi apa saja yang diharapkan selama proses pelaksanaan magang. Peserta magang akan belajar tentang lingkungan kerja pada kegiatan magang, bagaimana tim dalam perusahaan bekerja sama dan bagaimana magang akan membantu perusahaan menjalankan bisnisnya dan meningkatkan sistemnya selama kegiatan magang.

3.3.1.2 Pelaksanaan Tugas *Project* Internal Perusahaan

Peserta magang ditugaskan untuk membantu proyek website pencatatan penjualan internal perusahaan yang berlangsung mulai tanggal 22 Juli 2025 hingga 1 Agustus 2025. Website pencatatan penjualan internal perusahaan merupakan proyek untuk membangun sistem yang membantu perusahaan dalam melacak dan mencatat aktivitas penjualan produk dalam satu tempat. Proyek ini sebagian besar bertujuan untuk membuat proses pendokumentasian transaksi menjadi digital sehingga departemen yang tepat dapat mengelola data penjualan secara lebih terorganisir, akurat dan mudah ditemukan.

Dalam prosesnya, peserta magang diperkenalkan pada alur bisnis yang menjadi dasar pengembangan sistem, mulai dari proses input data penjualan, pengelolaan informasi produk hingga alur penyusunan laporan penjualan yang digunakan sebagai bahan evaluasi

internal melalui *website* tersebut. Peserta magang juga diberikan pemahaman mengenai bagaimana sistem ini dirancang untuk mengurangi ketergantungan pada pencatatan manual, meminimalkan potensi kesalahan serta mendukung penyajian data penjualan secara real time guna mempercepat proses pengambilan keputusan di level manajemen. Gambar 3.4 akan menampilkan gambar halaman *Dashboard* dari *website* internal perusahaan yang akan menampilkan beberapa informasi terkait penjualan yang telah dilakukan.



Gambar 3.4 Tampilan Utama *Website* Internal Perusahaan

Peserta magang mengerjakan proyek internal perusahaan sebagai salah satu cara untuk membiasakan diri dengan perusahaan dan sebagai latihan sebelum masuk ke dalam proyek utama yaitu *website asset management* yang memiliki ruang lingkup yang lebih kompleks. Pada fase pertama ini, peserta magang harus mempelajari cara kerja divisi *Quality Assurance*, standar operasional yang diterapkan dan cara menguji berbagai *test case scenario* di lingkungan perusahaan. Sebagai bagian dari pelatihan, peserta magang ditugaskan untuk melakukan pengujian di lingkungan pengembangan (*dev*). Ini adalah lingkungan pengujian pertama yang digunakan untuk memeriksa kualitas sistem sebelum dirilis ke lingkungan produksi (*production*). Peserta magang bertanggung jawab untuk membuat *test*

case scenario untuk beberapa modul di *website* internal perusahaan sebagai bagian dari pekerjaan mereka. Agar peserta magang dapat melihat bagaimana sistem berperilaku dalam situasi yang berbeda, pembuatan *test case scenario* melibatkan penempatan kasus uji ke dalam kelompok positif dan negatif. Selain itu, proses penyusunan *test case* menuntut pemahaman mendalam terhadap fungsionalitas modul, alur bisnis yang mendasari pengoperasian sistem, serta parameter yang berpotensi memengaruhi hasil pengujian. Komponen-komponen utama yang digunakan dalam penyusunan *test case* dijabarkan pada Tabel 3.2.

Tabel 3.2 Kolom dalam *Test Case Scenario*

Nama Kolom	Fungsi
<i>Test Case</i>	Berisi judul atau nama skenario pengujian yang menggambarkan tujuan dari test case tersebut.
<i>Type</i>	Menjelaskan kategori pengujian yang dilakukan, seperti <i>Positive Test</i> atau <i>Negative Test</i> . <ul style="list-style-type: none"> - <i>Positive Test</i> memastikan sistem bekerja sesuai alur normal. - <i>Negative Test</i> membuat validasi terhadap bagaimana sistem merespons input atau aksi yang tidak valid.
<i>Steps Description</i>	Memuat langkah-langkah yang harus diikuti dalam menjalankan test case. Instruksi pada bagian ini dibuat secara terstruktur agar proses pengujian dapat dilakukan dengan konsisten dan mudah direplikasi oleh penguji lain.
<i>Expected Result</i>	Berisi hasil yang seharusnya ditampilkan oleh sistem ketika langkah pengujian dilaksanakan dengan benar. Expected Result menjadi acuan untuk menilai apakah proses yang diuji sudah sesuai standar atau masih memerlukan perbaikan.
<i>Precondition</i>	Menjelaskan kondisi atau persiapan yang harus dipenuhi sebelum test case dijalankan. Contohnya data apa saja yang dibutuhkan sebelum

Nama Kolom	Fungsi
	melaksanakan langkah-langkah yang tercantum pada kolom <i>Steps Description</i>
<i>Post Condition</i>	Menjelaskan kondisi yang muncul setelah seluruh <i>steps</i> dijalankan. Perbedaannya dengan <i>actual result</i> adalah konten yang tercatat di dalamnya, dimana <i>post condition</i> akan menampilkan hasil dari <i>test case</i> secara <i>cross-module</i> , sedangkan <i>actual result</i> akan menampilkan hasil yang muncul pada modul yang dites
<i>Actual Result</i>	Mencatat hasil nyata yang muncul ketika <i>test case</i> dijalankan. Informasi pada bagian ini digunakan untuk membandingkan apakah output yang dihasilkan sistem sesuai dengan <i>Expected Result</i> atau tidak.
<i>Status</i>	Status digunakan untuk memberikan kesimpulan terhadap hasil pengujian jenis tertentu, yang terdiri dari: <ul style="list-style-type: none"> - Passed. Sebuah <i>test case</i> dapat dinyatakan berhasil jika <i>Actual Result</i> sesuai dengan <i>Expected Result</i>. - Bug. Sebuah proses pengujian menemukan ketidaksesuaian antara hasil aktual dan hasil yang diharapkan sehingga memerlukan tindak lanjut berupa perbaikan melalui bug reporting yang dilakukan oleh peserta magang - Not Checked. Proses pengujian belum dijalankan atau belum dapat dilakukan karena kendala tertentu.

Setelah semua kasus uji disusun dengan cermat dan bagian-bagiannya dijelaskan, peserta magang melanjutkan pengujian dengan melakukan pengujian manual berdasarkan petunjuk langkah demi langkah yang diberikan untuk setiap skenario. Pengujian dilakukan selangkah demi selangkah untuk memastikan bahwa setiap langkah

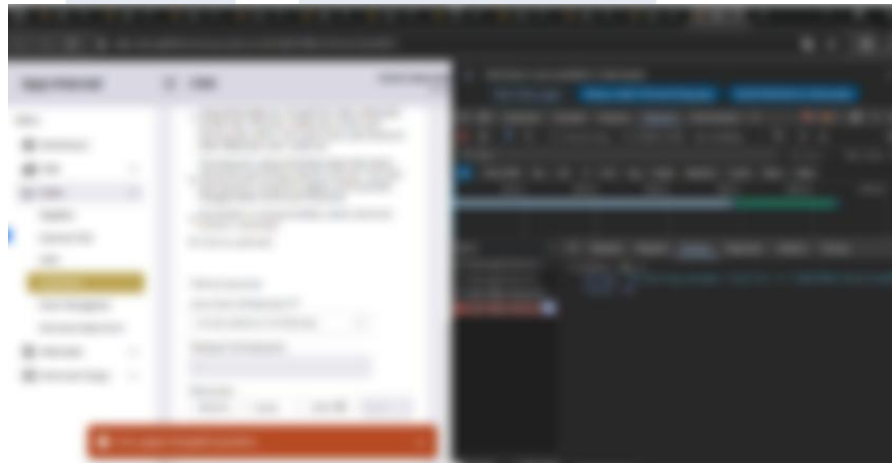
memeriksa cara kerja sistem dengan benar. Setelah pengujian selesai, hasil nyata dicatat dan dibandingkan dengan hasil yang diharapkan untuk melihat apakah sistem berfungsi sebagaimana mestinya. Jika pekerja magang menemukan perbedaan atau perilaku yang tidak sesuai spesifikasi, mereka harus menulis laporan bug terperinci yang mencakup deskripsi masalah, prosedur untuk mereproduksinya, dan bukti, seperti tangkapan layar. Tim pengembangan dapat menggunakan dokumentasi ini untuk membantu mereka memperbaiki masalah yang ditemukan. Gambar 3.5 menunjukkan bagaimana Google Spreadsheets menempatkan semua bagian hasil akhir ke dalam sebuah tabel.

The image shows a screenshot of a Google Spreadsheet interface. The spreadsheet contains a table with several columns and rows. The columns appear to be organized into sections, possibly for test case details, steps, and results. The rows contain text-based data, likely representing individual test cases or scenarios. The interface includes the standard Google Sheets toolbar at the top and a sidebar on the right.

Gambar 3.5 *Test Case Scenario Website Internal Perusahaan*

Setelah mendokumentasikan semua temuan pengujian di Google Spreadsheet, pekerja magang secara konsisten berunding dengan supervisor mereka untuk memverifikasi bahwa hasil yang diharapkan selaras dengan spesifikasi fungsi sistem. Selama pembicaraan ini, peserta magang memperoleh pemahaman yang lebih mendalam tentang logika bisnis yang berkaitan dengan setiap modul, yang mencakup kondisi batas, proses validasi, dan koneksi antar-fungsi yang sering tidak terlihat dari antarmuka. Penjelasan ini membantu peserta magang dalam memahami betapa sedikitnya

modifikasi pada suatu komponen dapat mempengaruhi perilaku sistem secara keseluruhan, khususnya dalam modul yang saling terkait dengan proses lain. Apabila metode pengujian menunjukkan bahwa perilaku sistem tidak sejalan dengan hasil yang diharapkan, peserta magang diwajibkan untuk membuat *bug reporting*. Selama proses ini, peserta magang diinstruksikan untuk menyertakan informasi penting seperti detail kesalahan dari *console*, *server response*, dan informasi tambahan yang diperoleh melalui fitur *Inspect Element* pada browser. Sebagaimana ditunjukkan pada Gambar 3.6, dokumentasi teknis ini digunakan oleh tim pengembangan untuk melaksanakan analisis dan perbaikan.



Gambar 3.6 Dokumentasi untuk *Bug Reporting*

3.3.1.3 Transisi ke *Project Website Asset Management*

Setelah peserta magang menyelesaikan pegetesan pada proyek internal perusahaan, *supervisor* dari peserta magang menilai bahwa peserta magang memiliki pemahaman yang cukup baik tentang proses dan alur kerja dari *Quality Assurance*. Peserta magang dinilai mampu membuat test case scenario yang terorganisir, melakukan pengujian manual berdasarkan langkah-langkah yang ditetapkan, menemukan ketidaksesuaian fungsional dan menulis *bug reporting* yang jelas dan terorganisir. Berdasarkan penilaian ini, *supervisor* kemudian

mengatur agar peserta magang pindah ke project website asset management yang merupakan salah satu produk perusahaan yang kini sedang melalui tahap transisi desain sistem strategis dan membutuhkan lebih banyak staf pengujian.

Perubahan arsitektur *website asset management* merupakan langkah yang sangat penting yang dimaksudkan untuk membuat program lebih terukur dan meningkatkan kinerjanya secara keseluruhan. Website yang awalnya dibuat untuk penggunaan internal menjadi lebih rumit dan perusahaan ingin menggunakan produk tersebut sebagai solusi manajemen aset jangka panjang untuk bisnis lain. Oleh karena itu, sistem memerlukan basis arsitektur yang lebih fleksibel dan efisien yang dapat menangani lebih banyak data dan lebih banyak pengguna di masa depan. Perubahan ini juga dimaksudkan untuk meningkatkan kinerja aplikasi, seperti seberapa cepat aplikasi dapat memproses data, seberapa cepat modul merespons satu sama lain, dan seberapa stabilnya selama operasi operasional yang melibatkan banyak transaksi. Arsitektur baru ini dimaksudkan untuk mengurangi kemacetan yang biasanya terjadi pada berbagai tahap proses, membuat struktur kode lebih modular, dan mempermudah penggunaan teknologi yang membantu aplikasi berfungsi lebih baik.

Tim menghasilkan peningkatan kinerja tidak hanya dalam aplikasi itu sendiri, namun juga dalam cara mereka bekerja sama untuk mengembangkannya. Dalam arsitektur lama, sulit bagi tim frontend dan backend untuk bekerja sama karena hal-hal seperti ketergantungan kode yang tinggi, masalah sinkronisasi perubahan, dan masalah dengan debugging dan integrasi. Perusahaan ingin membuat lingkungan pengembangan lebih efisien dengan membuat area kerja lebih jelas, menggunakan standar API yang lebih formal, dan menggunakan metode pengembangan yang memungkinkan

pengembangan secara bersamaan. Hal ini memungkinkan tim front-end dan back-end bekerja lebih mandiri, cepat, dan dalam skala yang lebih besar.

Selama masa transisi ini, perusahaan menekankan aktivitas pengujian, karena modifikasi apa pun pada arsitektur berpotensi mengakibatkan regresi pada fungsionalitas saat ini. Banyaknya penyesuaian dan persyaratan untuk validasi menyeluruh memerlukan kapasitas tambahan bagi tim QA. Oleh karena itu, peserta pelatihan berpartisipasi langsung dalam pengujian dalam lingkungan pengembangan yang menggabungkan arsitektur baru. Mereka memvalidasi proses bisnis penting, menilai integrasi antar modul, memastikan konsistensi di seluruh fungsi, dan mendokumentasikan semua masalah yang teridentifikasi secara menyeluruh untuk memfasilitasi penyelesaian yang cepat dan tepat oleh tim pengembangan.

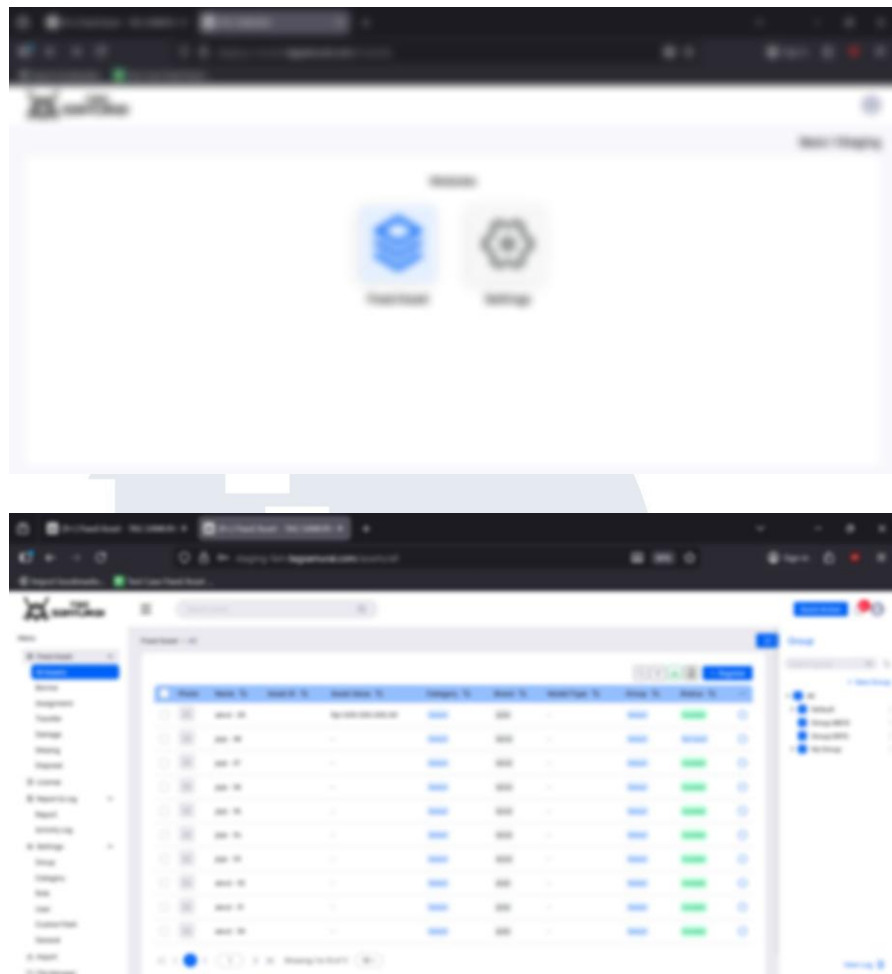
Melalui keterlibatan dalam project yang sedang berada dalam fase transformasi teknis ini, peserta magang memperoleh pengalaman yang lebih luas dibandingkan pengujian pada sistem stabil. Peserta dapat memahami bagaimana perubahan arsitektur berdampak pada performa aplikasi, proses pengembangan, hingga koordinasi lintas divisi. Peserta juga mengalami secara langsung dinamika kerja pada project prioritas dimana ketelitian, dokumentasi yang akurat dan komunikasi teknis merupakan hal yang sangat penting. Tahap ini menjadi pengalaman penting yang memperkuat kemampuan teknis dan pemahaman peserta magang mengenai pengembangan sistem dalam perusahaan.

Secara singkat, *website asset management* yang ditawarkan oleh perusahaan memiliki dua tingkatan *subscription*, yakni *Basic Pricing* dan *Enterprise Pricing*. *Basic Pricing* menyediakan fungsi-fungsi fundamental yang diperlukan untuk proses pencatatan dan

pemantauan aset dalam skala terbatas, yang mencakup fungsi pemantauan aset, pencatatan aktivitas dasar serta akses penggunaan untuk jumlah pengguna yang terbatas. Sementara itu, *Enterprise Pricing* menawarkan rangkaian fitur dan modul yang lebih lengkap, termasuk penambahan modul baru serta peningkatan kapabilitas dibandingkan dengan *basic pricing*. Paket ini dirancang dengan tingkat kompleksitas yang lebih tinggi sehingga lebih sesuai untuk kebutuhan organisasi berskala menengah hingga besar yang memerlukan pengelolaan aset secara terintegrasi dan lebih komprehensif.

3.3.1.4 Menjalankan *Testing* untuk *Basic Pricing*

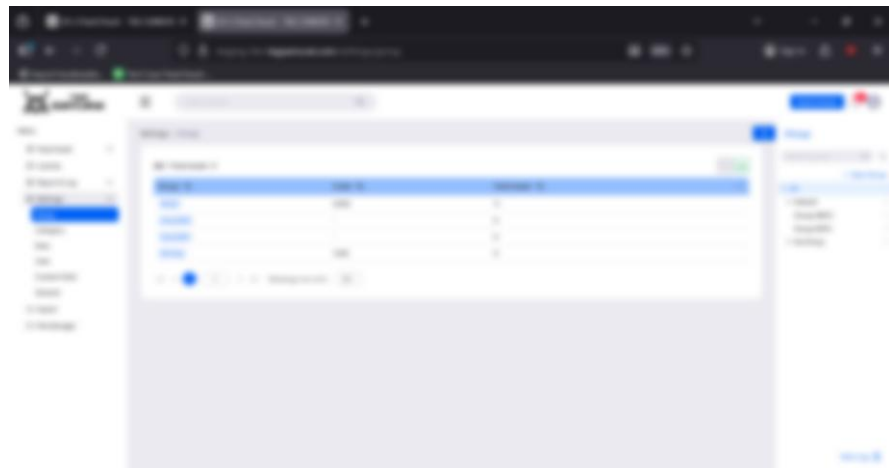
Setelah menjalani masa pembelajaran awal pada *project website* internal pencatatan penjualan, peserta magang kemudian diarahkan untuk mulai terlibat dalam pengujian pada project utama perusahaan yaitu *website asset management*, terutama pada *subscription basic pricing*. *Basic pricing* dari *website asset management* yang dikembangkan oleh PT. Qtera Mandiri dirancang sebagai paket fungsional dasar yang memberikan kemampuan inti bagi perusahaan untuk mengatur dan mengelola seluruh aset secara terintegrasi. Secara singkat, tujuan utama dari dibuatnya *website* ini adalah untuk menyediakan sarana operasional yang terpusat sehingga perusahaan dapat memonitor, mendokumentasikan, dan mengendalikan berbagai aset maupun kegiatan yang berkaitan dengannya secara lebih efisien dan akurat. Gambar 3.7 akan menampilkan tampilan utama dan *homepage* berisi daftar modul dari *website asset management* pada *basic pricing*.



Gambar 3.7 Tampilan Utama *Website Asset Management Basic Pricing*

Gambar 3.7 menampilkan tampilan utama dari website asset management khususnya untuk yang Basic Pricing, dimana setelah melakukan login, terdapat 2 pilihan menu yang dapat dipilih dan ketika memilih menu pengelolaan aset, maka akan menampilkan daftar aset yang terdaftar di perusahaan. Fokus utama yang dilakukan oleh peserta magang adalah melakukan pengetesan terhadap modul-modul yang ada di dalam *website* tersebut, maupun itu secara manual ataupun *automation*. Hal ini dilakukan untuk memastikan bahwa produk yang akan dijual oleh perusahaan, terlepas dari *bug* atau kesalahan fatal yang berpotensi dapat menghambat operasional *client*. Untuk mencapai itu, peserta magang menjalani beberapa tahapan utama. Pada tahapan awal, peserta magang dikoordinasi untuk

melakukan pengetesan terhadap beberapa modul, dengan kondisi menggunakan scenario yang sudah pernah dibuat sebelumnya oleh *supervisor*. Gambar dibawah akan menampilkan beberapa tampilan dari modul yang ditugaskan kepada peserta magang untuk dites.



Gambar 3.8 Tampilan Fitur *Group* pada *Basic Pricing*

Gambar 3.8 menunjukkan fitur *group*, yang merupakan tempat utama pengelolaan unit lokasi perusahaan. Fitur ini dimaksudkan untuk menunjukkan struktur lokasi hierarkis seperti, ruangan dengan area besar dan ruang yang lebih kecil seperti area kerja khusus. Fitur ini juga memiliki alat bagi pengguna untuk mengelola data di ruang-ruang tersebut.



Gambar 3.9 Tampilan Fitur *Category* pada *Basic Pricing*

Gambar 3.9 menampilkan tampilan dari fitur *category* yang berfungsi sebagai pusat pengaturan klasifikasi aset di dalam sistem. Fitur ini memungkinkan pengelolaan kategori dari aset yang akan didaftarkan agar lebih terorganisir.



Gambar 3.10 Tampilan Fitur *Role* pada *Basic Pricing*

Gambar 3.10 mengilustrasikan fitur *role*, yang berfungsi untuk mengatur alokasi peran di antara pengguna sistem. Fitur ini memungkinkan perusahaan untuk memastikan siapa yang memiliki wewenang untuk mengakses atau mengontrol bagian-bagian tertentu dari aplikasi, sesuai dengan peran dan kewajiban yang telah ditetapkan. Setiap posisi dapat diatur dengan hak akses yang berbeda, memungkinkan pengguna untuk mengamati atau menjalankan tindakan yang relevan dengan fungsi operasionalnya.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.11 Tampilan Fitur *User* pada *Basic Pricing*

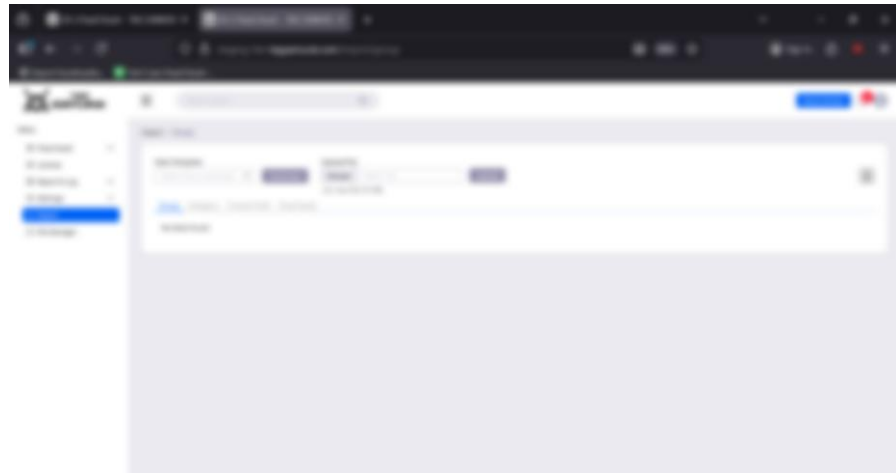
Gambar 3.11 menampilkan sebuah fitur *user* yang berfungsi sebagai pusat administrasi untuk mengelola seluruh pengguna yang terdaftar dalam sistem. Melalui bagian ini, perusahaan dapat mengelola pengguna yang akan menggunakan *website* tersebut berdasarkan kebutuhan operasional. Setiap pengguna dicatat secara individual sehingga seluruh aktivitas dan keterlibatan mereka dalam sistem dapat dipantau dengan lebih jelas.



Gambar 3.12 Tampilan Fitur *Custom Field* pada *Basic Pricing*

Gambar 3.12 menunjukkan fitur *custom field* yang memungkinkan pengguna memasukkan informasi tambahan yang tidak tersedia di kolom reguler saat mendaftarkan aset. Bagian ini

memungkinkan bisnis untuk membuat kolom unik yang sesuai dengan kebutuhan mereka. Ini mungkin mencakup segala hal mulai dari detail teknis dan informasi administratif hingga informasi tambahan yang terkait dengan kualitas aset.



Gambar 3.13 Tampilan Fitur *Import* pada *Basic Pricing*

Gambar 3.13 menampilkan sebuah fitur *import* yang disediakan untuk mempermudah proses pemasukan data dalam jumlah besar ke dalam sistem. Fitur ini memungkinkan pengguna mengunggah file berisi kumpulan informasi sehingga tidak perlu melakukan input satu per satu secara manual. Pendekatan ini secara signifikan meningkatkan efisiensi, terutama pada tahap awal implementasi atau ketika perusahaan melakukan pembaruan data dalam skala besar. Secara keseluruhan, gambar 3.8 hingga 3.13 menampilkan beberapa modul yang dites oleh peserta magang. Sebelum pelaksanaan pengetesan, peserta magang akan di briefing dengan alur dari masing-masing modul oleh supervisor sehingga peserta magang dapat memahami alur modul yang dites secara garis besar dan dapat menjalankan tes sesuai dengan scenario yang telah dibuat. Beberapa gambar dibawah akan menampilkan potongan scenario yang digunakan

Test Case ID	Description	Steps	Expected Results	Actual Results
TC-001	Test Case 1	Step 1	Expected 1	Actual 1
TC-002	Test Case 2	Step 1	Expected 1	Actual 1
TC-003	Test Case 3	Step 1	Expected 1	Actual 1
TC-004	Test Case 4	Step 1	Expected 1	Actual 1
TC-005	Test Case 5	Step 1	Expected 1	Actual 1
TC-006	Test Case 6	Step 1	Expected 1	Actual 1
TC-007	Test Case 7	Step 1	Expected 1	Actual 1
TC-008	Test Case 8	Step 1	Expected 1	Actual 1
TC-009	Test Case 9	Step 1	Expected 1	Actual 1
TC-010	Test Case 10	Step 1	Expected 1	Actual 1

Gambar 3.14 *Test Case Scenario* Fitur Group pada *Basic Pricing*

Gambar tersebut menggambarkan sejumlah test case scenario yang dimaksudkan untuk menguji fitur group, yaitu pengelolaan unit tempat di dalam sebuah perusahaan, seperti yang terlihat pada gambar 3.8. Semua skenario dicatat dalam spreadsheet yang mencakup bagian-bagian penting seperti deskripsi skenario, proses pengujian, kondisi awal yang harus dipenuhi, hasil yang diharapkan, dan output aktual yang muncul selama pengujian. Struktur pencatatan ini memastikan bahwa setiap bagian dari fitur dapat dievaluasi sepenuhnya, termasuk menambahkan lokasi baru, mengubah informasi, menghapus data, dan memeriksa apakah unit memiliki hubungan hierarki yang tepat.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

No	Test Case ID	Test Case Description	Test Data	Test Steps	Expected Results	Actual Results	Status
1	TC-001	Test Case for adding a new category	Category Name: New Category	1. Click on 'Add Category' button 2. Enter category name 3. Click on 'Save' button	Category is added successfully		Pass
2	TC-002	Test Case for updating a category	Category Name: Existing Category	1. Click on 'Edit Category' button 2. Enter new category name 3. Click on 'Save' button	Category is updated successfully		Pass
3	TC-003	Test Case for deleting a category	Category Name: Existing Category	1. Click on 'Delete Category' button 2. Confirm deletion 3. Click on 'Yes' button	Category is deleted successfully		Pass
4	TC-004	Test Case for checking category details	Category Name: Existing Category	1. Click on 'View Category' button 2. Check category details	Category details are displayed correctly		Pass

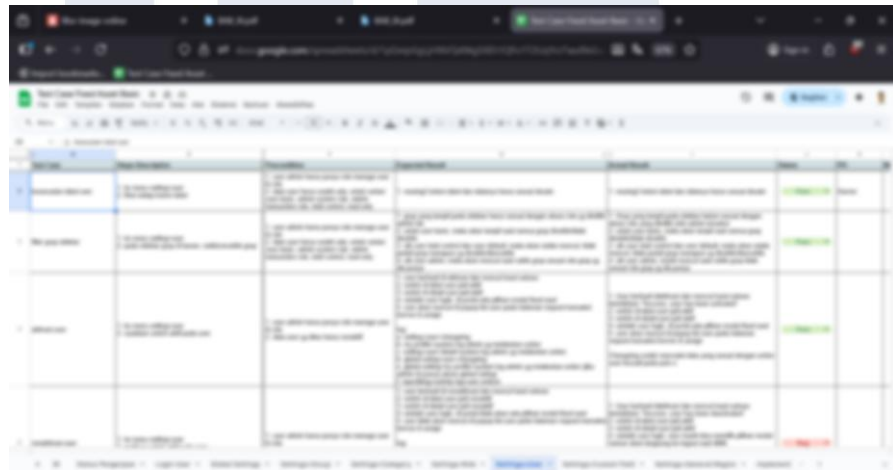
Gambar 3.15 *Test Case Scenario* Fitur *Category* pada *Basic Pricing*

Gambar 3.15 menunjukkan skenario kasus uji yang digunakan untuk menguji fitur *category*, yang merupakan pengaturan untuk mengklasifikasikan aset yang ditunjukkan pada Gambar 3.9. Seluruh skenario disajikan dalam tabel yang menunjukkan bagian-bagian penting seperti fungsi yang diuji, langkah-langkah untuk menjalankannya, kondisi awal, hasil yang diharapkan, dan keluaran aktual. Rangkaian kasus uji ini mencakup seluruh proses pengujian, mulai dari menambahkan kategori baru hingga memperbarui informasi, menghapus data, dan memeriksa input yang salah.

No	Test Case ID	Test Case Description	Test Data	Test Steps	Expected Results	Actual Results	Status
1	TC-001	Test Case for adding a new role	Role Name: New Role	1. Click on 'Add Role' button 2. Enter role name 3. Click on 'Save' button	Role is added successfully		Pass
2	TC-002	Test Case for updating a role	Role Name: Existing Role	1. Click on 'Edit Role' button 2. Enter new role name 3. Click on 'Save' button	Role is updated successfully		Pass
3	TC-003	Test Case for deleting a role	Role Name: Existing Role	1. Click on 'Delete Role' button 2. Confirm deletion 3. Click on 'Yes' button	Role is deleted successfully		Pass
4	TC-004	Test Case for checking role details	Role Name: Existing Role	1. Click on 'View Role' button 2. Check role details	Role details are displayed correctly		Pass

Gambar 3.16 *Test Case Scenario* Fitur *Role* pada *Basic Pricing*

Berdasarkan tampilan pada Gambar 3.16, dokumentasi skenario pengujian yang terlihat pada lembar kerja tersebut berisi rangkaian test case yang disusun untuk memverifikasi seluruh fungsi pada fitur *role* yang merupakan pengaturan peran pengguna. Skenario yang dicatat mencakup proses pembuatan peran baru, penyesuaian hak akses, pembaruan izin hingga penghapusan *role* yang tidak lagi digunakan. Setiap skenario dirancang untuk memastikan bahwa konfigurasi peran berjalan sesuai kebutuhan, termasuk memastikan bahwa pengguna hanya dapat mengakses bagian aplikasi yang relevan dengan tugasnya.



The image shows a screenshot of a test case scenario document. It is a table with multiple columns and rows. The columns are labeled: 'No', 'Uraian', 'Langkah', 'Data', 'Hasil', and 'Status'. The rows contain detailed test cases for the 'User' feature, including steps like 'Login', 'Logout', 'Add User', 'Edit User', and 'Delete User'. The 'Status' column shows the results of the tests, with some cells containing 'Pass' and others 'Fail'.

No	Uraian	Langkah	Data	Hasil	Status
1	Uraian	Langkah	Data	Hasil	Status
2	Uraian	Langkah	Data	Hasil	Status
3	Uraian	Langkah	Data	Hasil	Status
4	Uraian	Langkah	Data	Hasil	Status
5	Uraian	Langkah	Data	Hasil	Status
6	Uraian	Langkah	Data	Hasil	Status
7	Uraian	Langkah	Data	Hasil	Status
8	Uraian	Langkah	Data	Hasil	Status
9	Uraian	Langkah	Data	Hasil	Status
10	Uraian	Langkah	Data	Hasil	Status

Gambar 3.17 *Test Case Scenario* Fitur *User* pada *Basic Pricing*

Selain itu, Gambar 3.17 menunjukkan bahwa skenario pengujian yang dijelaskan difokuskan pada pengamatan detail operasi manajemen pengguna dalam fitur pengguna sistem, yang ditunjukkan pada Gambar 3.11. Skenario ini mencakup penambahan pengguna baru, pengisian informasi profil mereka, penetapan peran, dan pengecekan apakah pengguna baru dapat masuk dan menggunakan sistem berdasarkan hak akses yang diberikan kepada mereka.

Test Case ID	Test Case Description	Test Data	Test Steps	Expected Results	Actual Results	Status
TC-001	Validasi prosedur lengkap penambahan, konfigurasi, dan penggunaan kolom khusus dalam fitur custom field.	Input: Nama, Format Data, Status Aktif	1. Klik tombol 'Tambah Custom Field' 2. Isi form dengan data valid 3. Klik tombol 'Simpan'	Formulir pendaftaran aset menampilkan informasi tambahan yang benar.		Pending
TC-002	Validasi prosedur lengkap penambahan, konfigurasi, dan penggunaan kolom khusus dalam fitur custom field.	Input: Nama, Format Data, Status Aktif	1. Klik tombol 'Tambah Custom Field' 2. Isi form dengan data valid 3. Klik tombol 'Simpan'	Formulir pendaftaran aset menampilkan informasi tambahan yang benar.		Pending

Gambar 3.18 Test Case Scenario Fitur Custom Field pada Basic Pricing

Berdasarkan tampilan pada Gambar 3.18, skenario pengujian dirumuskan untuk memvalidasi prosedur lengkap penambahan, konfigurasi, dan penggunaan kolom khusus dalam fitur *custom field*, yang berfungsi sebagai informasi tambahan selama pendaftaran aset. Tampilan yang sesuai diilustrasikan pada Gambar 3.12. Pengujian ini melibatkan pembuatan bidang baru dengan berbagai tipe input, konfigurasi atribut seperti nama, format data, dan status aktif, serta verifikasi tampilan yang benar pada formulir pendaftaran aset.

Test Case ID	Test Case Description	Test Data	Test Steps	Expected Results	Actual Results	Status
TC-003	Validasi prosedur lengkap penambahan, konfigurasi, dan penggunaan kolom khusus dalam fitur import.	Input: Nama, Format Data, Status Aktif	1. Klik tombol 'Tambah Import' 2. Isi form dengan data valid 3. Klik tombol 'Simpan'	Formulir pendaftaran aset menampilkan informasi tambahan yang benar.		Pending
TC-004	Validasi prosedur lengkap penambahan, konfigurasi, dan penggunaan kolom khusus dalam fitur import.	Input: Nama, Format Data, Status Aktif	1. Klik tombol 'Tambah Import' 2. Isi form dengan data valid 3. Klik tombol 'Simpan'	Formulir pendaftaran aset menampilkan informasi tambahan yang benar.		Pending

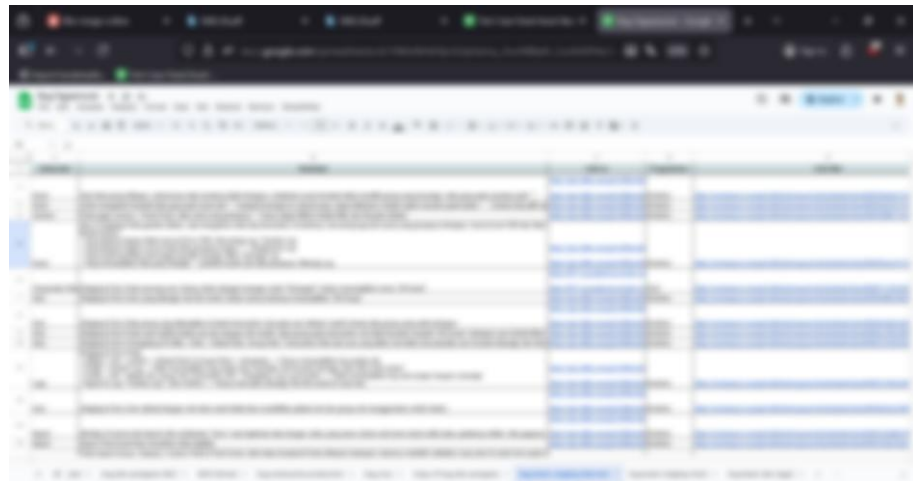
Gambar 3.19 Test Case Scenario Fitur Import pada Basic Pricing

Berdasarkan penggambaran pada Gambar 3.19, skenario pengujian untuk fitur *import* berpusat pada proses pengunggahan data

massal untuk memverifikasi bahwa sistem dapat menangani file yang terdiri dari kumpulan informasi secara akurat dan andal. Pengujian melibatkan verifikasi format file yang diizinkan, memastikan struktur kolom sesuai dengan spesifikasi sistem, dan menilai kemampuan aplikasi untuk menerima dan melihat pratinjau data sebelum diproses.

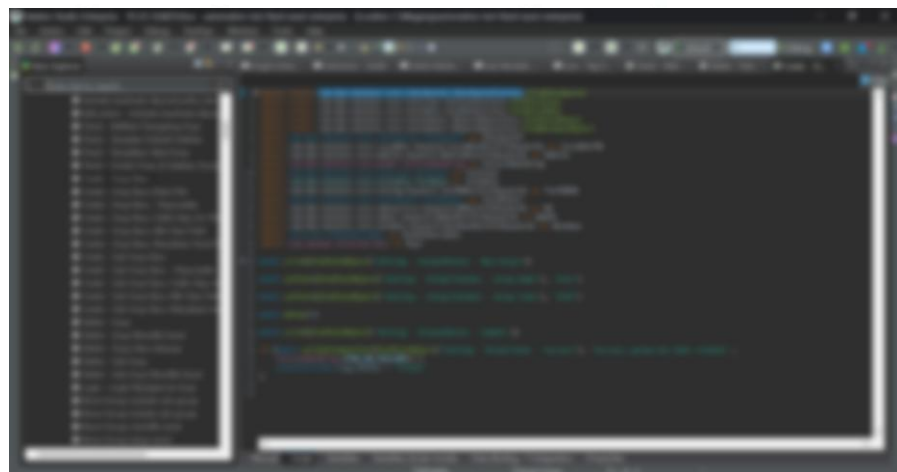
Merujuk pada Gambar 3.14 hingga 3.19, pengujian manual dilakukan pada setiap modul yang telah dijelaskan sebelumnya, dengan skenario yang dikembangkan untuk masing-masing modul yang mencakup semua kemungkinan eksekusi dalam modul tersebut, yang dikategorikan sebagai positif atau negatif. Pada setiap scenario, terdapat kolom status yang berguna untuk melakukan tracking terhadap kondisi dari scenario tersebut setelah dites, yang mencakup *bug* yang berarti masih ada kesalahan yang terjadi pada scenario tersebut dan *passed* yang berarti scenario tersebut berhasil dijalankan tanpa adanya masalah *bug*.

Pada modul-modul yang dites oleh peserta magang, ditemukan sejumlah 113 *bug* yang terbagi menjadi 3 kategori yaitu *critical bug*, *major bug* dan *minor bug*. Dalam temuannya, peserta magang menjabarkan sejumlah 7 bug yang termasuk dalam kategori bug kritis yang memiliki prioritas tertinggi karena berdampak pada beberapa proses transaksi dalam modul, 36 bug yang termasuk dalam bug mayor yang mengganggu fungsi utama tetapi biasanya disebabkan oleh kesalahpahaman dalam alur atau kebutuhan akan perubahan pada skema basis data, dan 70 bug minor yang tidak mengganggu proses transaksi modul, biasanya hanya kesalahan desain. Gambar 3.20 akan menunjukkan rangkuman bug yang telah dilaporkan oleh para peserta magang.



Gambar 3.20 Dokumentasi *Bug Reporting* untuk *Basic Pricing*

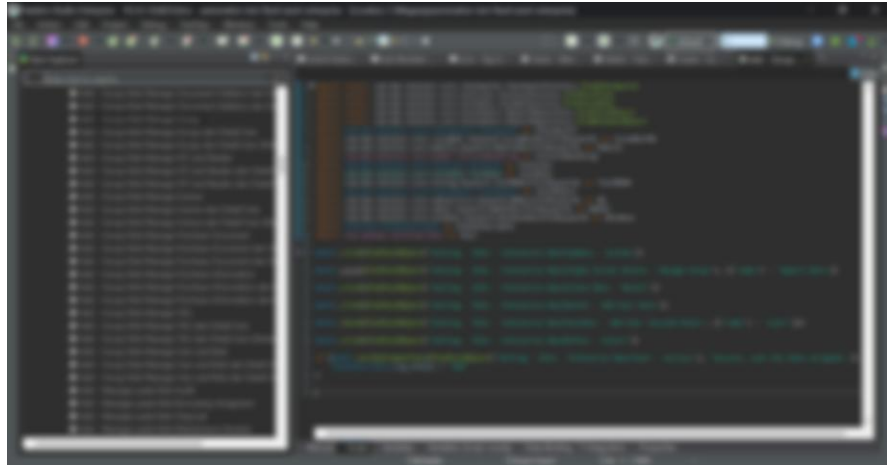
Setelah melalui proses *bug reporting* dan pengecekan ulang setelah *bug* telah selesai diperbaiki oleh tim *programmer*, proses akan dilanjutkan ke tahap *automation testing*. *Automation testing* akan menggunakan software *automation* populer yaitu Katalon Studio. Gambar dibawah akan menampilkan beberapa potongan gambar dari *automation script* yang dibuat oleh peserta magang dalam menjalankan *automation testing*.



Gambar 3.21 *Automation Script* *Fitur Group* pada *Basic Pricing*

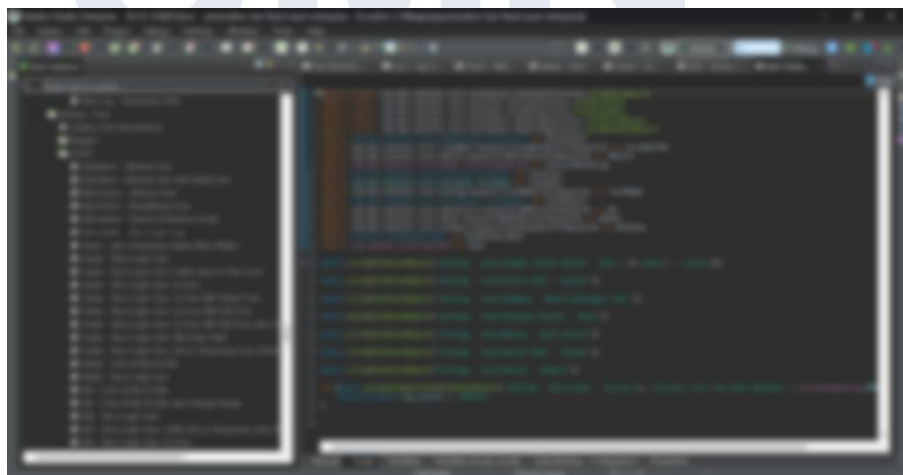
Gambar 3.21 diatas merupakan salah satu dari *automation script* yang dibuat oleh peserta magang untuk fitur *group* yang merupakan fitur untuk pengelolaan unit perusahaan. Pada tahap ini,

peserta magang membuat replikasi dari *steps* yang ada pada setiap *test case scenario* dan *automation* dilakukan untuk memastikan konsistensi dan beberapa *bug* ditemukan pada proses *automation*,



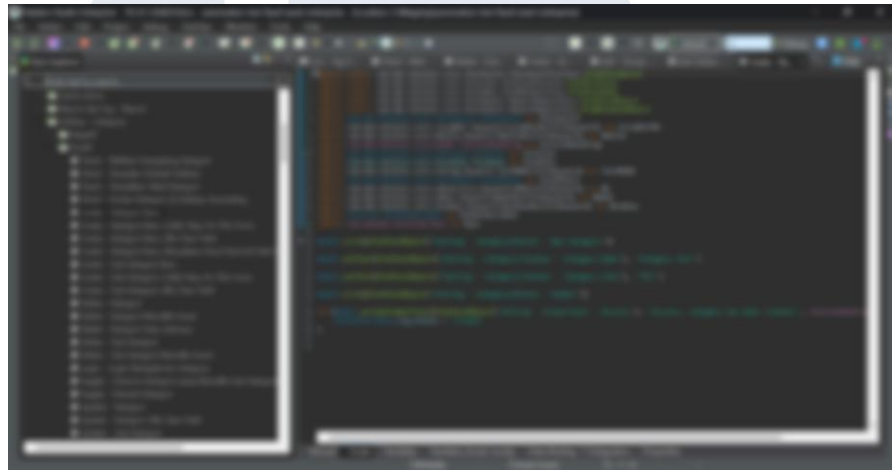
Gambar 3.22 *Automation Script* Fitur *Category* pada *Basic Pricing*

Gambar 3.22 di atas merupakan salah satu *automation script* yang dikembangkan oleh peserta magang untuk fitur pengaturan kategori aset. Pada tahap ini, peserta menyusun alur otomatis yang mereplikasi seluruh langkah pada *test case scenario*. *Automation* dijalankan untuk memastikan konsistensi respons sistem terhadap setiap skenario yang telah dirancang. Dari rangkaian pengujian tersebut, hal ini dilakukan untuk mengidentifikasi *bug* yang terjadi.



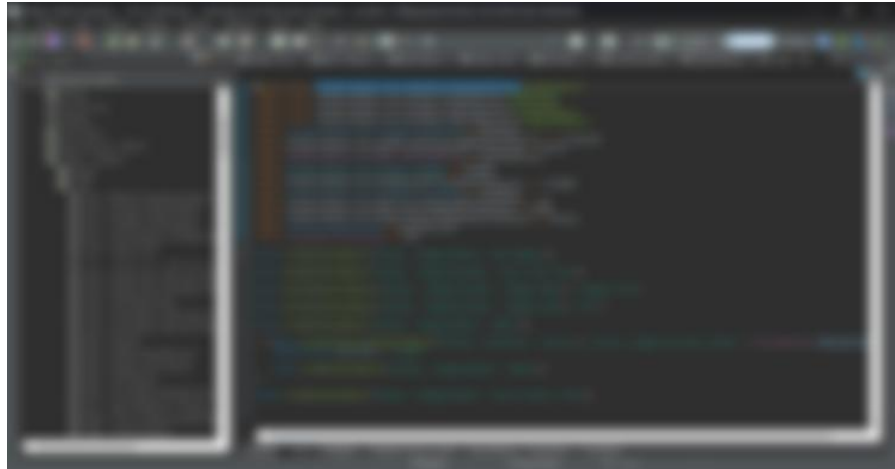
Gambar 3.23 *Automation Script* Fitur *Role* pada *Basic Pricing*

Gambar 3.23 di atas merupakan salah satu *automation script* yang dibuat oleh peserta magang untuk fitur *role*. Pada tahap ini, peserta menyusun rangkaian langkah otomatis yang mereplikasi seluruh skenario pengujian mulai dari proses penambahan peran baru hingga pengaturan untuk hak pengguna. *Automation* dijalankan untuk memastikan bahwa setiap alur berjalan secara konsisten tanpa bergantung pada interaksi manual, memastikan tidak ada *bug* yang muncul selama proses *automation*.



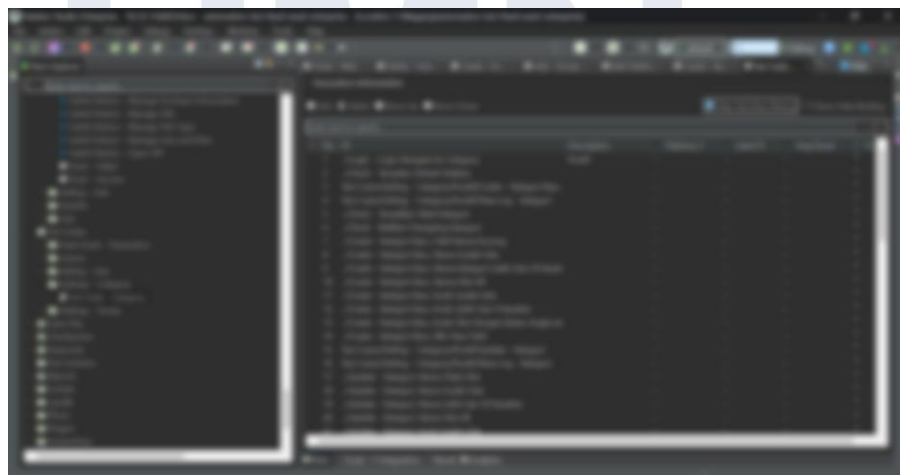
Gambar 3.24 *Automation Script* Fitur *User* pada *Basic Pricing*

Gambar 3.24 di atas merupakan salah satu *automation script* yang dikembangkan oleh peserta magang untuk fitur pengelolaan pengguna dalam sistem. Pada tahap ini, peserta menyusun rangkaian langkah otomatis yang mereplikasi skenario pengujian terkait proses administrasi pada fitur *user*. Tahap *automation* dijalankan untuk memastikan stabilitas fungsi dan konsistensi hasil tanpa perlu interaksi manual. Dalam prosesnya, ditemukan berbagai bug yang berkaitan dengan validasi input dan ketidaksesuaian tampilan setelah penyimpanan data.



Gambar 3.25 *Automation Script* Fitur *Custom Field* pada *Basic Pricing*

Berdasarkan tampilan pada gambar 3.25, *automation script* yang dikembangkan peserta magang berfungsi untuk mereplikasi skenario pengujian fitur *custom field*. *Script* ini menjalankan langkah-langkah yang ada dalam skenario pada gambar 3.18. *Automation* dijalankan untuk memastikan konsistensi hasil dan dari keseluruhan eksekusi ditemukan bug terkait validasi format data. Setelah proses pembuatan script selesai, sebuah test suite untuk masing-masing modul dimana salah satunya akan ditampilkan pada gambar 3.26.



Gambar 3.26 Contoh *Test Suite* untuk *Basic Pricing*

Gambar 3.26 merupakan *test suites* yang dibuat oleh peserta magang setelah menyelesaikan pengerjaan seluruh *automation script*

dari fitur yang ditugaskan. Test suites sendiri dibuat untuk menyimpan seluruh *test case scenario script* yang telah dibuat dan dikelompokkan per fiturnya. Pengelompokan ini bertujuan untuk memudahkan eksekusi pengujian secara terstruktur dan memastikan bahwa tiap skenario dapat dijalankan dalam satu rangkaian yang konsisten. Melalui test suite tersebut, seluruh skenario dapat dieksekusi secara iteratif dalam sekali proses tanpa harus dijalankan satu per satu, sehingga efisiensi waktu meningkat dan konsistensi hasil pengujian lebih terjaga.

Secara keseluruhan, para peserta magang mengembangkan skrip untuk setiap skenario yang sebelumnya digunakan dalam pengujian manual. Pengembangan skrip otomatisasi dimulai dengan proses pendidikan yang dilakukan oleh para peserta pelatihan untuk membiasakan diri dengan bahasa pemrograman Groovy dan fitur-fitur utama yang ditawarkan oleh Katalon Studio. Pada tahap ini, peserta memperoleh pengetahuan tentang pengelolaan Object Repository, sebuah repositori yang menyimpan semua objek UI yang terlibat dalam proses otomatisasi, termasuk tombol, kolom input, dan elemen lainnya. Selain itu, peserta menggunakan fitur Spy Web untuk menangkap elemen secara langsung dengan membuka aplikasi web, memastikan setiap komponen diidentifikasi secara akurat sebelum digunakan dalam skenario pengujian. Elemen yang ditangkap kemudian diberi XPath sebagai *locator* utama untuk menjamin bahwa setiap objek dapat diakses dan diinteraksikan dengan andal oleh skrip otomatisasi, dengan memanfaatkan kemampuan yang disediakan oleh perangkat lunak Katalon Studio. Pada tahap selanjutnya, gambar 3.26 akan menampilkan *Test Suites* yang dibuat oleh peserta magang yang berisi seluruh test case yang digabungkan menjadi satu agar dapat dijalankan secara langsung secara iteratif. Setelah seluruh proses berhasil berjalan dengan baik tanpa terhenti karena adanya bug,

peserta magang akan melanjutkan tanggungjawabnya ke bagian yang lainnya.

3.3.1.5 Menjalankan *Testing* untuk *Enterprise Pricing*

Setelah menyelesaikan fase pengujian untuk paket Basic Pricing, peserta magang selanjutnya diinstruksikan untuk melanjutkan ke pengujian tingkat layanan yang lebih canggih, khususnya pada paket Enterprise Pricing. Paket ini merupakan versi yang lebih baik dari situs web manajemen aset, yang dikembangkan untuk mengakomodasi kebutuhan perusahaan skala menengah hingga besar, sehingga menawarkan fungsionalitas yang jauh lebih luas dibandingkan dengan harga dasar. Enterprise Pricing mencakup semua fitur dasar yang ditawarkan dalam Basic Pricing dan juga meningkatkan kemampuan sistem melalui fungsionalitas canggih yang dirancang untuk memenuhi kebutuhan operasional organisasi dengan cakupan dan kompleksitas yang lebih besar. Pada tahap ini, sistem beralih dari manajemen aset dasar ke penawaran dukungan untuk proses bisnis yang lebih canggih, termasuk manajemen data yang ekstensif, manajemen aset dengan pemantauan dan fungsionalitas administratif tambahan, serta mekanisme pengendalian internal yang lebih baik. Peningkatan tersebut bertujuan memastikan bahwa perusahaan dapat menjalankan aktivitas manajemen aset secara efisien dalam skala yang lebih besar sehingga dapat menjadi sebuah solusi bagi perusahaan yang membutuhkan berbagai kebutuhan operasional yang lebih dinamis.

Dalam proses pengujian, peserta magang perlu memahami alur bisnis yang lebih mendalam karena setiap modul pada tingkat *Enterprise* saling terhubung dan memiliki dependensi yang lebih tinggi dibandingkan modul-modul yang ada pada *Basic Pricing*. Dalam melaksanakan proses pengetesan pada *Enterprise Pricing*, proses ini menuntut ketelitian dalam mengidentifikasi skenario positif dan negatif sehingga memastikan validasi input berjalan secara

konsisten sesuai dengan *test case scenario* yang telah dibangun, serta mengevaluasi apakah setiap proses dapat menangani volume data dan variasi kondisi yang lebih kompleks dari yang dilakukan pada *Basic Pricing*. Sebagai permulaan, peserta magang diberikan koordinasi oleh *supervisor* untuk langsung ikut serta dalam proses pengetesan *Enterprise Pricing* pada *website asset management*. Berikut adalah gambar dari tampilan utama *website asset management Enterprise Pricing* yang akan menjadi subjek pengetesan peserta magang.



Gambar 3.27 Tampilan Utama *Website Asset Management Enterprise Pricing*

Setelah itu, peserta magang diberikan beberapa penjelasan terkait alur bisnis utama dari *website* tersebut terutama pada modul-modul yang belum pernah peserta magang ketahui. Pada tahap ini,

supervisor memastikan peserta memiliki pemahaman mengenai hubungan antarfitur/*cross-module*, berbagai prosedur dalam penggunaan sistem serta berbagai informasi terkait pelaksanaan rangkaian proses pengetesan seperti informasi terkait penanggungjawab dari masing-masing modul yang akan dikerjakan oleh peserta magang serta pemberian akses terhadap *UI/UX design* yang dibuat dalam *platform* Figma. Informasi yang diberikan ini penting karena modul-modul pada Enterprise Pricing memiliki keterkaitan yang lebih kompleks sehingga kesalahan dalam memahami satu proses dapat memengaruhi akurasi hasil pengujian pada modul lainnya. Setelah itu, peserta magang diperkenalkan dengan beberapa *test case scenario* yang akan digunakan oleh peserta magang sebagai dasar pengetesan.

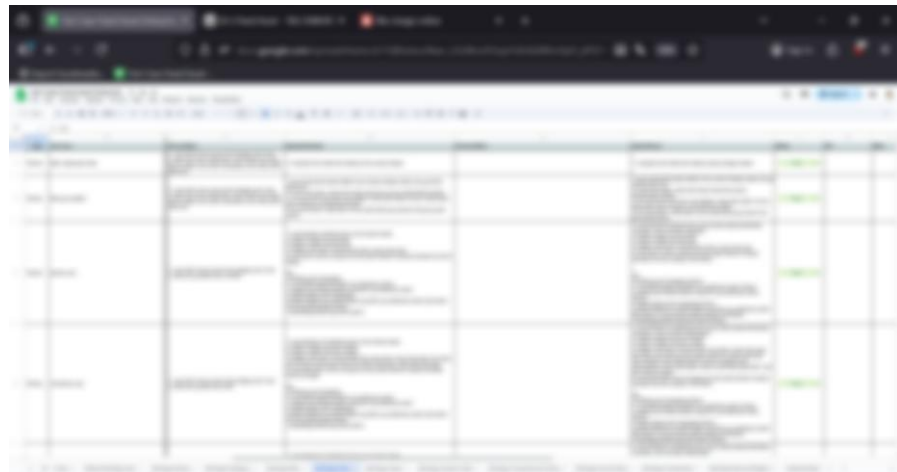
Setelah peserta magang memahami gambaran alur bisnis secara menyeluruh, *supervisor* kemudian memperkenalkan serangkaian *test case scenario* yang harus digunakan sebagai dasar pengujian. *Test case* tersebut mencakup berbagai kondisi fungsional, baik skenario positif untuk memastikan fitur berjalan sesuai desain, maupun skenario negatif untuk menguji ketahanan sistem terhadap kesalahan input atau kondisi yang tidak ideal. Peserta magang juga diberikan panduan mengenai prioritas skenario, dependensi antarfitur, serta batasan-batasan yang perlu diperhatikan selama proses pengujian melalui beberapa informasi pada *test case* seperti yang akan ditampilkan pada gambar-gambar berikut.

Test Case ID	Test Case Description	Test Data	Expected Results	Actual Results	Status
TC-001	Verify user can login with valid credentials	username: admin, password: 123456	User is redirected to the dashboard	User is redirected to the dashboard	Pass
TC-002	Verify user can login with invalid credentials	username: admin, password: 12345	User is redirected to the login page with an error message	User is redirected to the login page with an error message	Pass
TC-003	Verify user can create a new role	role name: admin, role description: admin	User is redirected to the role management page with a success message	User is redirected to the role management page with a success message	Pass
TC-004	Verify user can delete a role	role id: 1	User is redirected to the role management page with a success message	User is redirected to the role management page with a success message	Pass
TC-005	Verify user can assign a role to a user	user id: 1, role id: 1	User is redirected to the user management page with a success message	User is redirected to the user management page with a success message	Pass
TC-006	Verify user can remove a role from a user	user id: 1, role id: 1	User is redirected to the user management page with a success message	User is redirected to the user management page with a success message	Pass

Gambar 3.28 *Test Case Scenario* Fitur *Role* pada *Enterprise Pricing*

Gambar 3.28 adalah *test case scenario* yang dibuat untuk fitur *role* pada *Enterprise Pricing*. Ini memuat rangkaian test case yang disusun untuk memverifikasi keseluruhan fungsi pada fitur *role* dalam *Enterprise Pricing* dimana jumlah dan kompleksitas *role* yang tersedia jauh lebih luas dibandingkan *Basic Pricing*. Skenario yang disusun mencakup pembuatan berbagai jenis *role* yang lebih variatif untuk memastikan pemberian tanggungjawab yang sesuai terhadap modul-modul yang saling terhubung. Setiap skenario dirancang untuk memastikan bahwa pengaturan peran yang lebih berlapis dan beragam pada *Enterprise Pricing* berfungsi dengan benar termasuk memastikan bahwa setiap pengguna memperoleh akses terbatas sesuai tanggung jawabnya

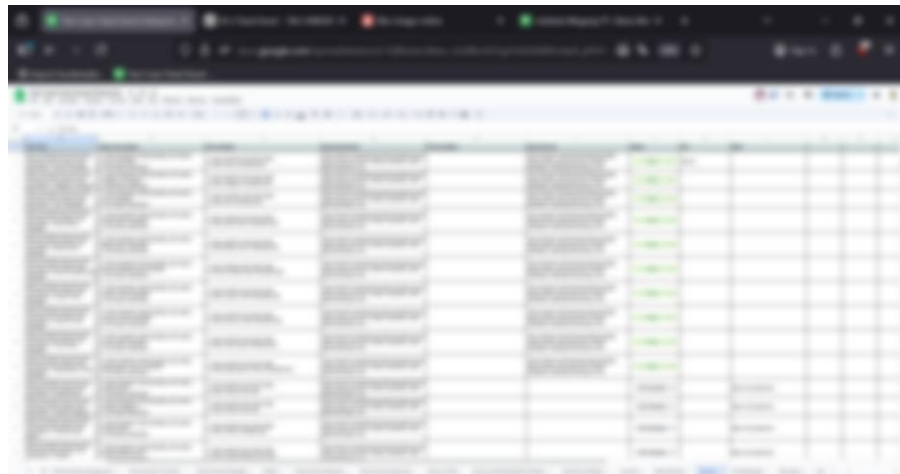
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.29 *Test Case Scenario* Fitur *User* pada *Enterprise Pricing*

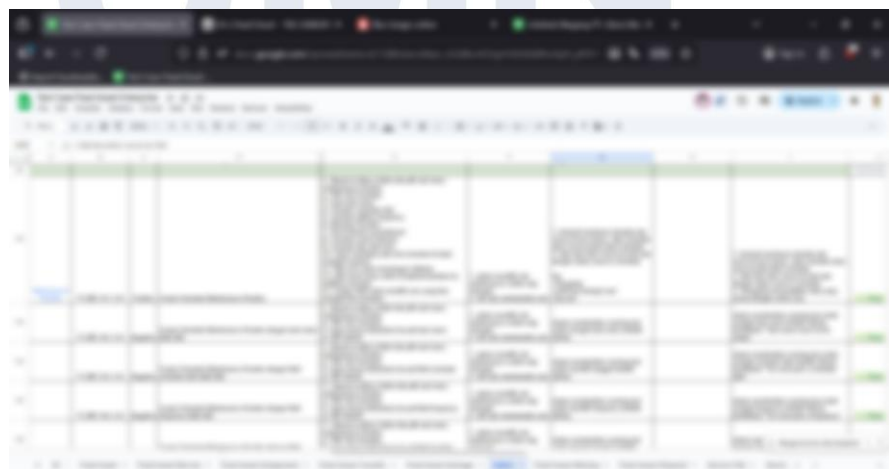
Selanjutnya, pada gambar 3.29, *test case scenario* yang terdokumentasi memuat rangkaian test case yang difokuskan pada evaluasi menyeluruh terhadap fungsi-fungsi pengelolaan pengguna pada fitur *user* yang menjadi pusat administrasi seluruh akun dalam sistem. Skenario tersebut mencakup proses pendaftaran pengguna baru dengan berbagai variasi data, pengisian dan pembaruan informasi profil, pengaturan peran dan hak akses yang sesuai dengan kebutuhan hingga verifikasi bahwa setiap pengguna yang ditambahkan dapat melakukan login serta mengakses modul secara konsisten sesuai izin dan *role* yang telah ditetapkan. Seluruh skenario dirancang untuk memastikan bahwa fitur mampu menangani perubahan data dalam berbagai kondisi.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



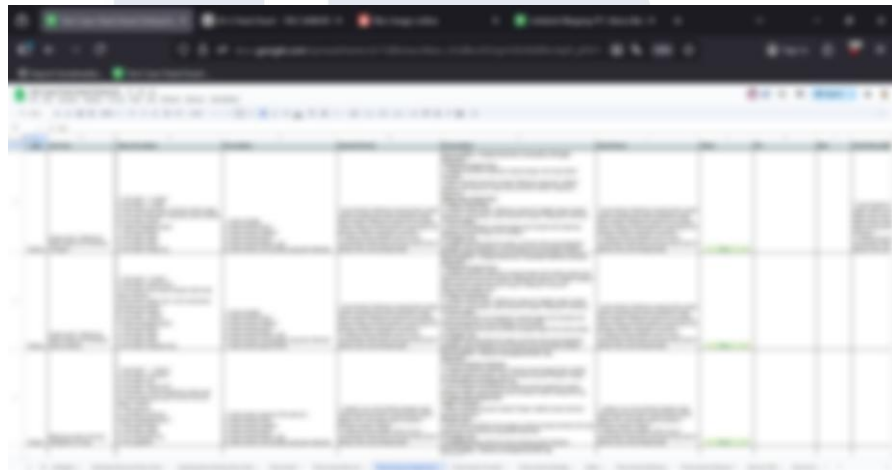
Gambar 3.30 *Test Case Scenario* Fitur *Import* pada *Enterprise Pricing*

Gambar 3.30 mengilustrasikan skenario kasus uji yang secara khusus dikembangkan untuk fungsi *import* dalam sistem yang didedikasikan untuk entri data massal. Skenario kasus uji ini dikembangkan untuk memverifikasi kemampuan sistem dalam memproses file yang berisi data yang selanjutnya akan diintegrasikan ke dalam sistem sesuai dengan proses bisnis yang telah ditetapkan. Pengujian mencakup verifikasi format file yang diizinkan, kepatuhan struktur kolom terhadap standar yang ditetapkan sistem, dan kemampuan aplikasi untuk mengakses dan melihat pratinjau data sebelum prosedur *import*.



Gambar 3.31 *Test Case Scenario* Fitur *Addon* pada *Enterprise Pricing*

Selanjutnya, *test case scenario* pada gambar 3.31 disusun untuk menguji beberapa fitur berbentuk *addon* yang dapat ditambahkan ke dalam akun perusahaan sesuai kebutuhan operasional. Terdapat lima *addon* yang tersedia yaitu penjadwalan *audit*, penjadwalan *maintenance*, pembuatan tiket perbaikan, *service center*, serta pelacakan aset yang hilang. Setiap *addon* memiliki alur kerja dan aturan pemrosesan data yang berbeda sehingga diperlukan skenario pengujian yang detail untuk memastikan bahwa proses aktivasi *addon* agar eksekusi tiap fungsi dapat berjalan dengan benar.

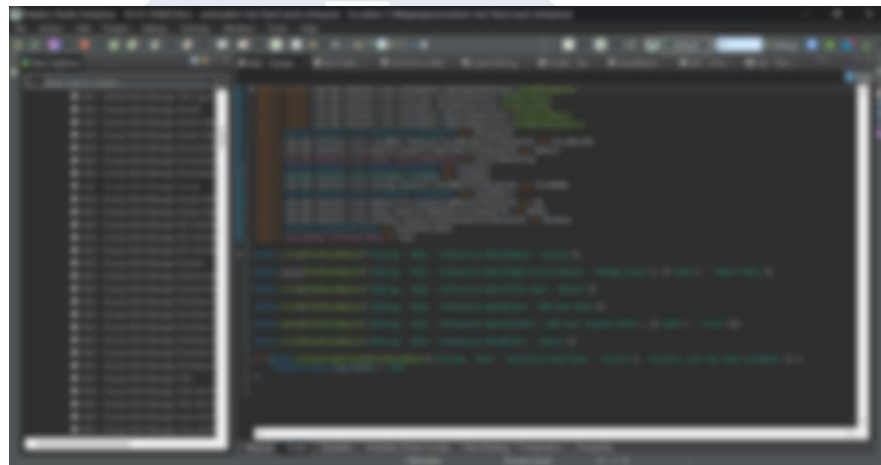


Gambar 3.32 *Test Case Scenario* Fitur *Transaction* pada *Enterprise Pricing*

Gambar 3.32 adalah test case scenario untuk fitur transaction pada *Enterprise Pricing*. Fitur transaksi merupakan elemen penting dari situs web manajemen aset, yang memfasilitasi keseluruhan proses operasional yang terkait dengan manajemen aset. Fitur ini memungkinkan pengguna untuk melakukan berbagai transaksi aset, termasuk peminjaman aset, penugasan kepada pengguna yang ditunjuk, transfer ke unit kerja yang berbeda, dokumentasi keadaan aset seperti kerusakan atau kehilangan, dan pencatatan proses pembuangan aset. Setiap skenario kasus uji menjamin bahwa proses transaksi berfungsi sesuai dengan alur bisnis yang telah ditentukan dan

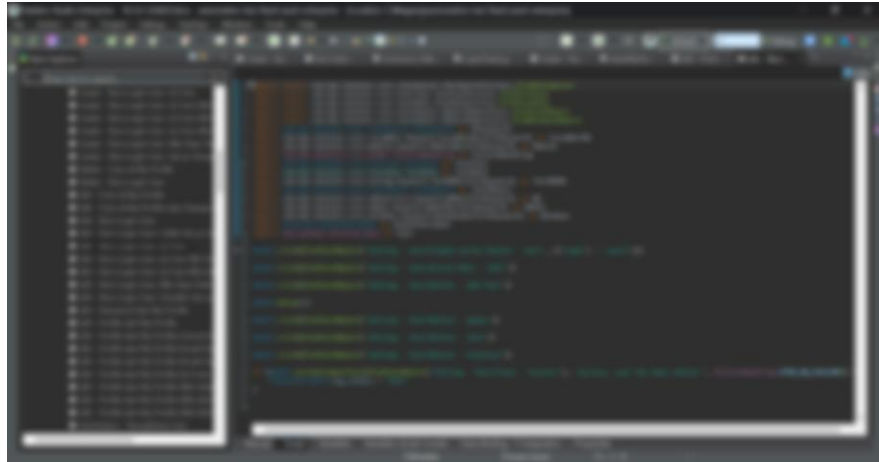
dapat diselesaikan dalam berbagai kondisi, baik secara positif maupun negatif.

Setelah proses pengetesan manual telah diselesaikan, peserta magang melanjutkan proses pengetesan dengan menjalankan *automation testing* terhadap modul-modul yang telah dites secara manual sebelumnya seperti yang akan dideskripsikan berikut.



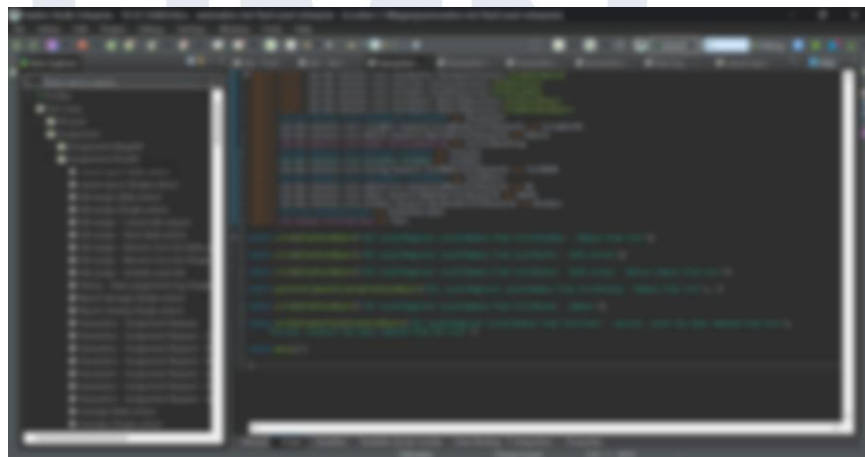
Gambar 3.33 *Automation Script* Fitur Role pada *Enterprise Pricing*

Gambar 3.33 menunjukkan salah satu *automation script* yang dikembangkan oleh peserta magang untuk menguji fungsi pada fitur *role*. Sama seperti yang dilakukan pada *Basic Pricing*, peserta magang menyesuaikan pembuatan *script* mengikuti *steps* yang ada pada *test case scenario*, dimana *test suite* dari *automation script* untuk modul tersebut tersebut dijalankan guna memastikan setiap proses berfungsi secara stabil dan sesuai dengan alur. Dari perancangan tersebut, sebagian besar bug yang ditemukan adalah bug yang dipengaruhi karena perbedaan konsep pengembangan dibanding dengan alur bisnisnya.



Gambar 3.34 *Automation Script* Fitur *User* pada *Enterprise Pricing*

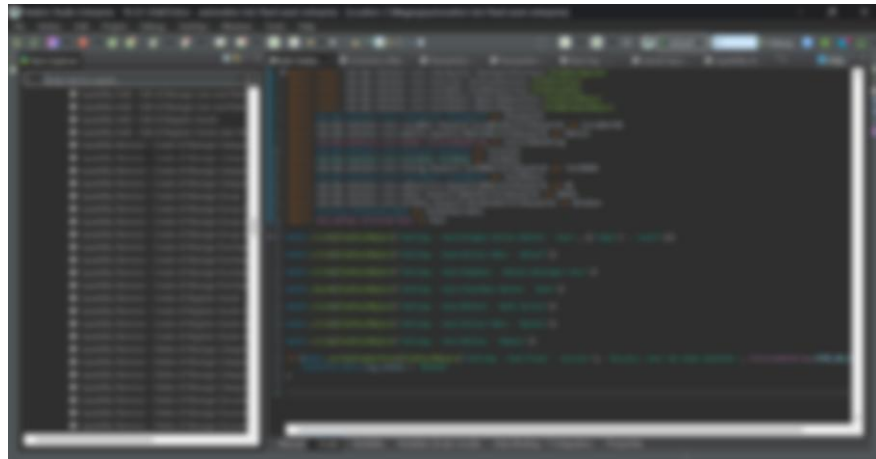
Gambar 3.34 menampilkan salah satu *automation script* yang dikembangkan oleh peserta magang untuk menguji fitur pengelolaan pengguna dalam sistem. Pada tahap ini, peserta merancang rangkaian langkah otomatis yang mereplikasi seluruh skenario pengujian terkait proses administrasi pada fitur *user*. Automation tersebut kemudian dijalankan dan teridentifikasi beberapa bug yang berkaitan dengan validasi input serta ketidaksesuaian fitur dengan alur bisnis yang telah dibuat.



Gambar 3.35 *Automation Script* Fitur *Addon* pada *Enterprise Pricing*

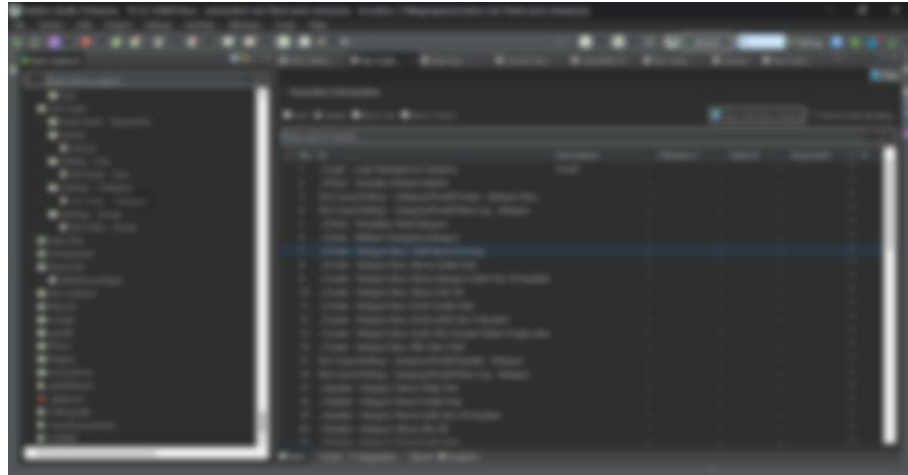
Gambar 3.35 menampilkan *automation script* yang dikembangkan oleh peserta magang untuk menguji fitur *addon* yang

dapat ditambahkan ke perusahaan. Pada tahap ini, *automation* tersebut dijalankan dan teridentifikasi *bug* yang sebagian besarnya diakibatkan oleh ketidaksesuaian fitur dengan alur bisnis yang telah dibuat.



Gambar 3.36 Automation Script Fitur Transaction pada Enterprise Pricing

Gambar 3.36 menunjukkan salah satu *automation script* yang dikembangkan oleh peserta magang untuk menguji fitur *transaction* pada *website asset management*. Sama seperti yang dilakukan pada *Basic Pricing*, peserta magang menyesuaikan pembuatan *script* mengikuti *steps* yang ada pada *test case scenario*. *Automation* tersebut dijalankan guna memastikan setiap proses berfungsi secara stabil dan sesuai dengan alur, dimana ditemukan beberapa *bug* yang muncul selama eksekusi *automation* terkait ketidaksesuaian data, visual dan validasi.



Gambar 3.37 Contoh *Test Suites* untuk *Enterprise Pricing*

Gambar 3.37 merupakan *screenshot* dari salah satu *test suites* yang dibuat oleh peserta magang setelah menyelesaikan pengerjaan seluruh *automation script* dari fitur yang ditugaskan pada *Enterprise Pricing*. *Test suites* ini dibuat untuk menjalankan sekumpulan test case scenario yang telah dibuat dan disusun secara iteratif. Melalui *test suite* tersebut, seluruh skenario dapat dieksekusi secara berurutan dalam sekali proses tanpa harus dijalankan satu per satu, sehingga dapat mengidentifikasi bug tanpa perlu menjalankan *test case* satu per satu.

3.3.2 Kendala yang Ditemukan

Selama pelaksanaan kegiatan magang di PT. Qtera Mandiri, terdapat beberapa kendala yang dialami oleh peserta magang sehingga mempengaruhi kelangsungan proses kerja, meliputi:

1. Alur website yang cukup kompleks. *Website Asset Management* yang dikembangkan memiliki struktur modul, alur navigasi serta variasi fitur yang tergolong kompleks. Kompleksitas ini membuat peserta membutuhkan waktu lebih banyak untuk memahami hubungan antar modul, logika kerja fitur dan proses bisnis yang diimplementasikan pada *website*. Hal ini sempat menghambat proses pelaksanaan pengujian.

2. Koneksi internet yang kurang stabil. Proses pengerjaan testing, terutama automation testing, membutuhkan koneksi internet yang stabil agar eksekusi script dapat berjalan dengan lancar. Kondisi jaringan kantor terkadang tidak stabil sehingga menyebabkan gangguan serta kegagalan pada saat menjalankan testing karena proses *loading* yang terhenti sehingga pekerjaan menjadi kurang efisien.
3. Kurangnya komunikasi antar divisi. Beberapa tugas memerlukan koordinasi lintas divisi, seperti divisi UI/UX, *Backend Developer* dan *Frontend Developer*. Namun, pada beberapa kesempatan, komunikasi antar divisi belum berjalan optimal sehingga peserta mengalami kesulitan dalam memperoleh informasi teknis dan terkadang tidak mendapatkan informasi detail terkait perubahan yang terjadi, khususnya terkait penjelasan fitur dan perubahan requirement yang terjadi selama pengembangan.

3.3.3 Solusi atas Kendala yang Ditemukan

Untuk mengatasi kendala yang ditemukan selama pelaksanaan kegiatan magang di PT. Qtera Mandiri, beberapa solusi yang diterapkan oleh peserta magang agar pelaksanaan kegiatan magang tetap berjalan secara optimal, yang mencakup:

1. Berkonsultasi langsung kepada supervisor atau divisi UI/UX terkait alur website. Solusi ini dilakukan untuk membantu peserta magang memahami alur kerja website yang tergolong kompleks. Dengan melakukan konsultasi langsung kepada supervisor maupun tim UI/UX, peserta magang dapat memperoleh penjelasan yang lebih rinci mengenai struktur modul, alur navigasi serta logika desain yang digunakan pada *website Asset Management*. Melalui sesi diskusi bersama tim terkait, peserta magang dapat memahami hubungan antar fitur dan proses bisnis yang diterapkan dalam *website* sehingga meminimalkan

terjadinya miskonsepsi saat melakukan pengujian *test case* sehingga pekerjaan dapat dilakukan secara lebih efektif dan sesuai dengan kebutuhan project.

2. Menggunakan koneksi internet pribadi. Agar proses testing, terutama automation testing, tetap berjalan optimal, peserta dapat menggunakan koneksi internet pribadi sebagai alternatif ketika jaringan kantor tidak stabil. Penggunaan *hotspot* atau koneksi cadangan membantu menjaga kelancaran eksekusi script automation yang sensitif terhadap gangguan koneksi.
3. Bersikap lebih proaktif dalam berkomunikasi dengan anggota divisi lainnya. Untuk mengatasi hambatan komunikasi, peserta perlu bersikap lebih proaktif dalam melakukan koordinasi dengan anggota divisi lain yang terkait proses pengembangan seperti *UI/UX*, *Backend Developer* dan *Frontend Developer*. Ketika mengalami permasalahan, peserta magang akan menghampiri anggota yang menjadi penanggungjawab untuk berkonsultasi secara langsung atau dapat dilakukan melalui chat pribadi di aplikasi Whatsapp. Hal ini dilakukan agar peserta magang dapat meminta penjelasan teknis terkait fitur serta update perubahan *requirement*.