

BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Selama periode magang, dalam divisi *Quality Assurance* (QA) di bawah naungan Bapak Bondan yang memimpin departemen Teknologi, berperan sebagai anggota tim magang. Departemen Teknologi merupakan bagian integral dari struktur organisasi PT Indo Koala Remittance, yang fokus pada pengembangan dan pemeliharaan aplikasi *mobile* Jack & Transfez serta aspek teknis lainnya yang mendukung operasional perusahaan. Dalam struktur organisasi ini, tindakan sebagai *Quality Assurance intern*, yang secara langsung dibimbing oleh Kak Rio sebagai *supervisor* dan kak Luthfiandra selaku mentor. Peran mentor sangat penting dalam mengarahkan dan melatih dalam menjalankan tugas-tugas QA dengan efektif dan efisien.

Sebagai anggota tim QA, tentunya akan terlibat dalam berbagai proyek yang diarahkan oleh mentor. Tugas-tugas tersebut mencakup pengujian perangkat lunak, identifikasi masalah atau cacat dalam aplikasi, serta memberikan masukan konstruktif untuk meningkatkan kualitas produk. Dalam menjalankan tugas-tugas ini, nantinya akan berkoordinasi langsung dengan mentor dan anggota tim lainnya dalam tim proyek. Koordinasi tim dilakukan secara berkala melalui beberapa saluran komunikasi. Pertama, terdapat pembaruan harian (*daily update*) yang dilakukan secara online via zoom. Selanjutnya, ada juga pertemuan mingguan (*weekly meeting*) di mana anggota tim QA berkumpul via *online* untuk membahas proyek-proyek yang sedang berjalan, menyelesaikan masalah yang mungkin muncul, dan merencanakan langkah-langkah selanjutnya.

Selain itu, sebagai bagian dari divisi *Quality Assurance*, yang juga terhubung dengan divisi teknologi lainnya melalui aplikasi Slack. Slack ini bertujuan untuk memfasilitasi komunikasi antara tim QA dan manajemen yang bertanggung jawab atas operasional aplikasi Jack & Transfez serta proyek-proyek terkait. Dengan cara ini, sistem koordinasi yang terstruktur dan beragam memungkinkan tim QA untuk

bekerja secara efektif dalam menjalankan tugas-tugas mereka, memastikan kualitas produk yang tinggi, dan menyumbangkan nilai tambah bagi kesuksesan perusahaan secara keseluruhan. Di divisi Teknologi PT Indo Koala Remittance, pengguna bekerja dengan dua aplikasi utama: Jack, yang digunakan untuk bisnis ke bisnis (B2B), dan Transfez, yang digunakan untuk bisnis ke konsumen (B2C). Keduanya memiliki fokus dan kebutuhan teknis yang berbeda. Tim yang memiliki proyek bersangkutan dengan aplikasi Transfez lebih sering menggunakan Gather, sebuah *platform* komunikasi yang membantu tim memantau beban aplikasi dan berkoordinasi secara efisien. Gather memungkinkan anggota tim untuk berkomunikasi langsung, berbagi informasi, dan melacak perkembangan proyek meskipun bekerja dari tempat yang berbeda. Koordinasi kegiatan yang dilakukan oleh tim QA dijabarkan dalam tabel 3.1.

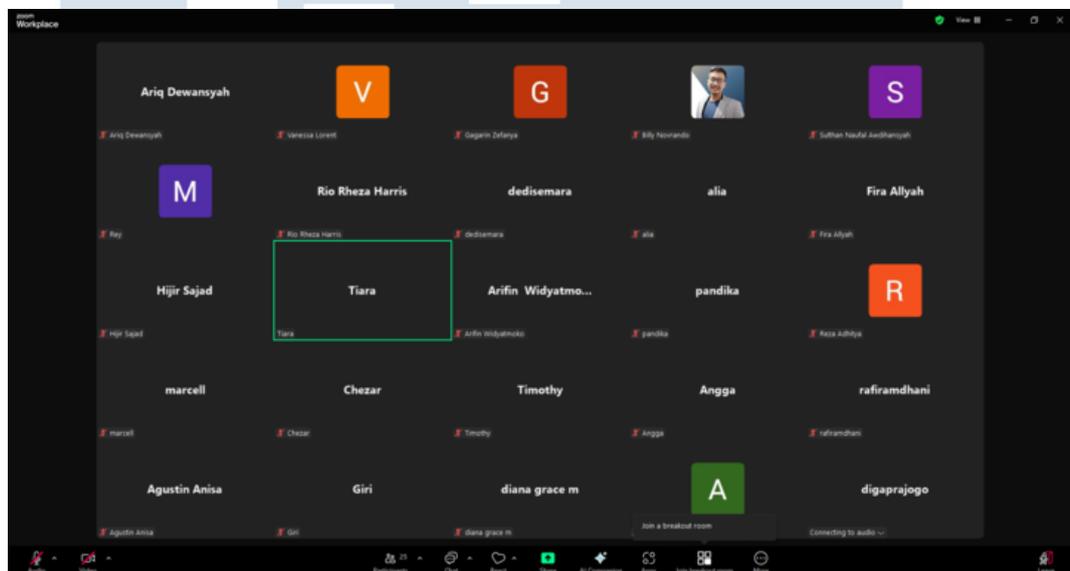
Tabel 3.1 Koordinasi Kegiatan Tim QA

Kegiatan	Frekuensi Kegiatan	Tujuan
<i>Daily Update</i>	Setiap Hari	<ul style="list-style-type: none"> - Menyampaikan laporan mengenai aktivitas yang telah dikerjakan pada hari sebelumnya. - Menyampaikan rencana kerja atau tugas yang akan dilakukan pada hari itu. - Membahas kendala atau masalah yang muncul selama proses pelaksanaan proyek.
<i>Weekly Meeting</i> (tim QA)	Setiap hari Jumat	<ul style="list-style-type: none"> - Memberikan update mengenai perkembangan proyek yang sedang dikerjakan oleh masing-masing anggota divisi <i>Quality Assurance</i>. - Mengadakan diskusi mengenai rencana kerja mingguan, yang mencakup tugas-tugas yang akan dikerjakan oleh anggota tim QA sepanjang minggu tersebut.

Kegiatan	Frekuensi Kegiatan	Tujuan
<i>Kick off</i>	Setiap kali proyek baru akan dimulai.	<ul style="list-style-type: none"> - <i>Kick-off</i> dilakukan setiap kali memulai proyek baru dengan tujuan untuk memahami keseluruhan proyek, termasuk tujuan, ruang lingkup, dan deadline yang ditetapkan. Ini juga membantu menyamakan pemahaman antara semua pihak yang terlibat agar setiap anggota tim tahu peran dan tanggung jawabnya.
Gather	Setiap Hari	<ul style="list-style-type: none"> - Mempermudah interaksi langsung antara anggota tim dalam proyek-proyek teknis, meskipun bekerja dari lokasi yang berbeda. - Membantu tim dalam melacak perkembangan proyek dan menyelesaikan masalah untuk memastikan kelancaran proses kerja.
<i>Deployment</i>	Setiap Hari Rabu	<ul style="list-style-type: none"> - Mengirimkan aplikasi atau fitur baru ke lingkungan produksi agar dapat diakses dan digunakan oleh pengguna akhir. - Memastikan bahwa aplikasi atau sistem berjalan dengan lancar di lingkungan yang sesungguhnya dan dapat berfungsi sesuai dengan yang diharapkan tanpa masalah teknis.

3.1.1 Daily Update

Daily update adalah kegiatan rutin yang dilakukan oleh seluruh tim divisi teknologi untuk melaporkan perkembangan pekerjaan. Dalam sesi ini, setiap anggota tim memberikan pembaruan terkait pekerjaan yang telah diselesaikan, tugas yang sedang dikerjakan, serta rencana berikutnya. Setelah itu, pembaruan ini akan disampaikan kembali di tribe masing-masing untuk memastikan seluruh tim memiliki pemahaman yang selaras dan terkoordinasi dengan baik.

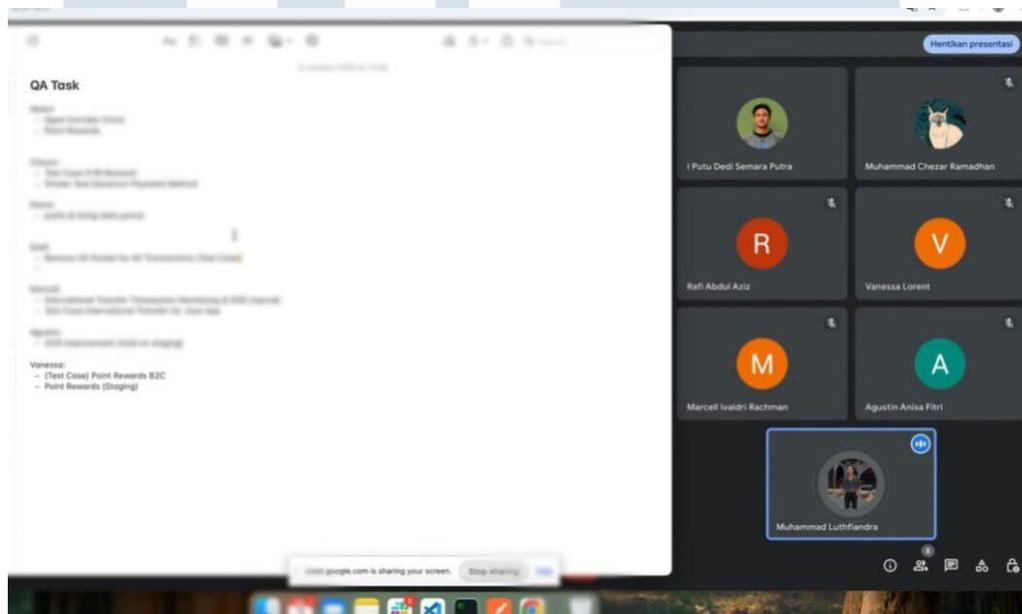


Gambar 3.1 *Daily Update*

Gambar 3.1 diatas menampilkan kegiatan *daily update* oleh divisi teknologi. Selama proses pekerjaan magang, *daily update* ini biasanya dilakukan secara *online* melalui Zoom, sehingga memudahkan komunikasi lintas tim meskipun bekerja secara *remote*. Kegiatan ini bertujuan untuk meningkatkan transparansi, memantau progres proyek, serta mendeteksi hambatan secara dini sehingga solusi dapat segera diterapkan..

3.1.2 Weekly Meeting

Weekly meeting adalah pertemuan rutin yang dilakukan khusus oleh tim QA untuk membahas progres pekerjaan selama seminggu terakhir, mengevaluasi hasil pengujian, dan merencanakan tugas untuk minggu berikutnya. Dalam sesi ini, laporan mengenai kendala yang dihadapi, solusi yang telah diterapkan, serta strategi untuk meningkatkan kualitas pengujian dibahas secara terperinci.

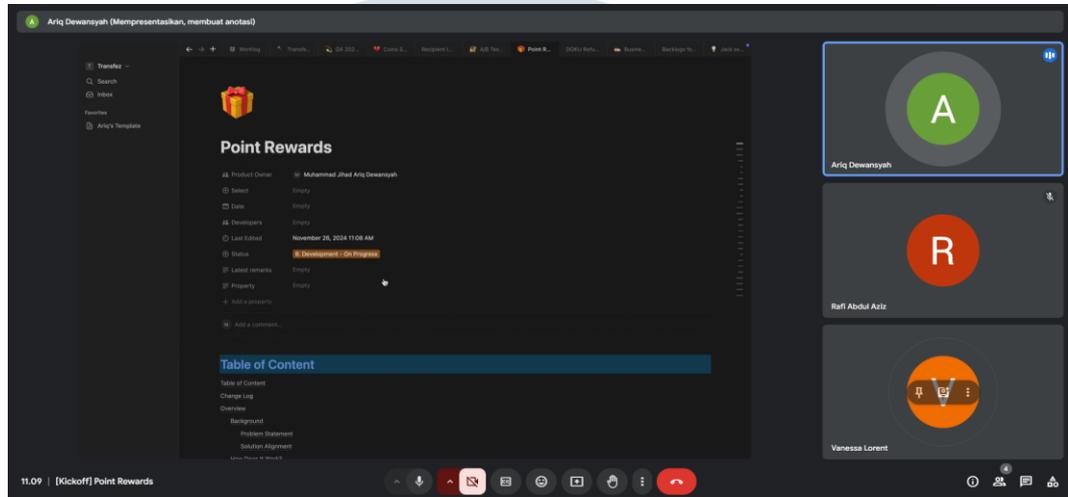


Gambar 3.2 Weekly Meeting Tim QA

Pada gambar 3.2 diatas menampilkan Tim QA yang sedang melakukan *weekly meeting*. *Weekly meeting* biasanya dilakukan melalui Zoom untuk memastikan koordinasi yang baik di antara anggota tim, memperkuat kolaborasi, dan menjaga fokus pada pencapaian tujuan tim QA.

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

3.1.3 Kick-off



Gambar 3.3 Kick Off

Kick-off adalah kegiatan yang dilakukan setiap kali ingin memulai proyek baru. Pada gambar 3.3 memberikan gambaran mengenai kegiatan *Kick-off* yang dilakukan secara *online*. Tujuannya adalah untuk memastikan semua pihak yang terlibat memiliki pemahaman yang sama mengenai proyek tersebut. Selama *kick-off*, tujuan, ruang lingkup, jadwal, dan *deadline* proyek dijelaskan secara rinci, serta peran dan tanggung jawab setiap anggota tim ditentukan. Kegiatan ini juga bertujuan untuk:

- Menyelaraskan ekspektasi dan memastikan komunikasi yang jelas antara semua anggota tim.
- Menyusun rencana kerja dan langkah-langkah yang perlu diambil.
- Mengidentifikasi potensi tantangan atau hambatan yang mungkin muncul selama proyek berjalan.

Dengan melakukan *kick-off*, semua pihak yang terlibat bisa memulai proyek dengan pemahaman yang sama, sehingga memperlancar proses dan mencapai tujuan proyek secara efisien.

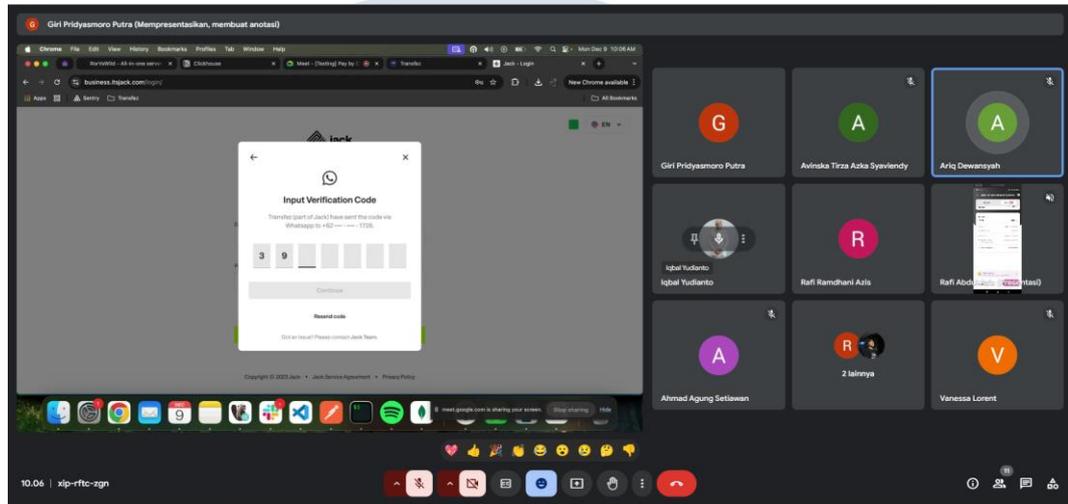
3.1.4 Gather



Gambar 3.4 Gather

Gather adalah *tools* yang digunakan untuk aktivitas rutin yang dilaksanakan setiap hari oleh tim B2C (Transfez) sesuai dengan jam kantor untuk memfasilitasi komunikasi, koordinasi, dan pertukaran informasi terkait progres pekerjaan. Pada Gambar 3.4 menampilkan aktivitas tim dalam Gather. Kegiatan ini menjadi wadah bagi tim untuk membahas kendala yang dihadapi, menyampaikan pembaruan pekerjaan, dan memastikan keselarasan dalam mencapai tujuan yang telah ditetapkan. Selain itu, Gather juga digunakan untuk merancang strategi penyelesaian tugas, memberikan umpan balik, serta memperkuat kolaborasi antar anggota tim. Dengan adanya Gather yang terjadwal sesuai jam kantor, tim dapat memantau progres harian secara efektif, menjaga kualitas hasil kerja, dan memastikan bahwa semua anggota tetap terhubung serta selaras dalam menjalankan tanggung jawab mereka.

3.1.5 Deployment



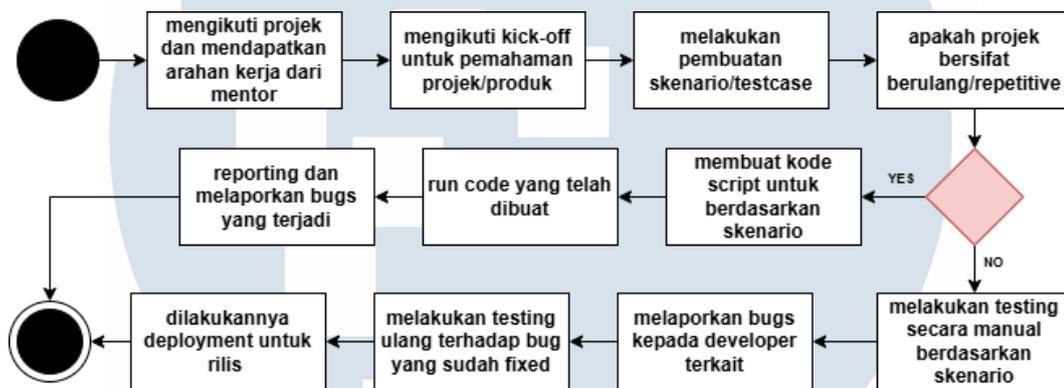
Gambar 3.5 Kegiatan *Deployment*

Deployment adalah proses rutin yang dilakukan setiap hari Rabu untuk mengirimkan aplikasi atau fitur baru ke lingkungan produksi agar dapat diakses dan digunakan oleh pengguna akhir. Pada saat *deploy*, semua anggota tim yang terkait berkumpul di Zoom seperti pada Gambar 3.5 untuk memastikan bahwa proses pengiriman aplikasi atau fitur berjalan dengan lancar. Aktivitas ini bertujuan memastikan bahwa aplikasi atau sistem berfungsi sesuai dengan yang diharapkan di lingkungan yang sesungguhnya tanpa masalah teknis. Selama sesi tersebut, tim berkoordinasi langsung, memantau setiap langkah yang dilakukan, dan segera mengatasi masalah yang mungkin muncul. Dengan demikian, tim dapat memastikan bahwa aplikasi atau sistem yang diterapkan di lingkungan produksi berjalan dengan baik dan stabil.

3.2 Tugas dan Uraian Kerja Magang

Program magang di PT. Indo Koala Remittance berlangsung selama minimal tiga bulan, dimulai pada bulan September 2024. Setiap peserta magang akan

ditempatkan pada salah satu aplikasi, yaitu Jack atau Transfez, dan proyek akan diberikan sesuai dengan penugasan masing-masing. Dalam bidang Quality Assurance (QA), tugas yang akan dilakukan mencakup pengujian perangkat lunak secara manual maupun menggunakan alat otomatis. Peserta magang nantinya akan terlibat dalam berbagai tahap pengujian produk untuk memastikan bahwa kualitasnya memenuhi standar yang telah ditetapkan.



Gambar 3.6 *Flowchart* kerja magang

Selama menjalani program magang, proses kerja dapat digambarkan dengan diagram alir yang terdapat dalam Gambar 3.6. Semua langkah yang diambil dimulai dengan instruksi dan pembagian tugas dari mentor, kemudian akan bergabung pada proyek yang telah disesuaikan. Proses dimulai dengan mengikuti *kick-off* untuk mendapatkan pemahaman yang komprehensif mengenai proyek. *Kick-off* diadakan sebelum proyek dimulai, bertujuan untuk menyelaraskan tujuan dan ekspektasi semua pihak yang terlibat. Setiap *kick-off* mencakup pembagian sistem desain yang meliputi ruang lingkup proyek, alur fitur, serta konsep desain fitur yang akan diterapkan. Selain itu, di dalam *kick-off* juga dilakukan penjelasan mengenai peran dan tanggung jawab setiap anggota tim, *timeline* proyek, serta potensi tantangan yang mungkin dihadapi. Hal ini memastikan bahwa semua anggota tim memiliki pemahaman yang sama dan siap untuk melaksanakan tugas mereka dengan efisien. Pembuatan skenario dilakukan berdasarkan PRD dan desain yang telah disediakan, yang mencakup skenario tampilan antarmuka serta elemen-elemen dalam seperti sisi *database*. Skenario tersebut disusun menggunakan *Google Sheets* dan meliputi

komponen seperti *Requirement*, Alur Cerita, Rangkuman progres pekerjaan, *Scenario Test*, dan *QA Buglist*. Skenario ini berfungsi sebagai panduan dalam pengujian dan pembuatan kode untuk pengujian otomatis. Pengujian otomatis diterapkan jika proyek memiliki sifat repetitif.

Untuk pengujian portal Jack, *automation* menggunakan Cypress dengan bahasa JavaScript, sementara untuk aplikasi *mobile*, pengujian dilakukan menggunakan WebDriverIO yang nantinya akan terhubung ke Appium. Kode pengujian dibuat berdasarkan skenario dan *test case* yang telah disusun. Setelah kode selesai dibuat, dilakukan uji coba untuk memastikan tidak ada masalah teknis yang menghalangi proses pengujian. Pada pengujian manual, *bug* yang ditemukan akan dilaporkan secara manual, dan setelah perbaikan dilakukan, pengujian ulang dilakukan untuk memastikan fitur berfungsi dengan baik. Di sisi lain, pada pengujian otomatis, apabila ada *bug*, Cypress akan mengirimkan notifikasi secara otomatis. Jika proyek tidak memerlukan pengujian otomatis, pengujian akan dilakukan secara manual.

3.2.1 Tugas Kerja Magang

Sebagai QA di PT Indo Koala Remittance, pengguna dapat ditempatkan pada proyek B2B atau B2C, dengan tanggung jawab utama melakukan pengujian pada aplikasi Jack atau Transfez, khususnya pada fitur-fitur baru sebelum dirilis ke publik. Pengujian dilakukan secara manual untuk fitur baru yang tidak repetitif, sementara fitur yang memerlukan pengujian berkala seperti *welcome page*, *login*, *dashboard*, dan lainnya, akan diuji menggunakan metode *automation testing*.

Setiap proyek yang dikerjakan dimulai dengan *kick-off* yang menjelaskan gambaran umum proyek dan cakupan tugas dari setiap divisi yang terlibat. Selanjutnya, dilakukan pembuatan skenario pengujian berdasarkan desain sistem dan UI/UX yang telah dibuat di Figma serta PRD

yang ada di Notion. Setelah aplikasi diperbarui oleh tim pengembang, divisi QA akan memulai proses pengujian terhadap aplikasi tersebut.

Dalam pengujian manual, setiap kali ditemukan *bug* atau masalah pada aplikasi, akan disusun laporan yang dapat diakses oleh seluruh anggota tim proyek. Laporan ini dibuat oleh tim *Quality Assurance* (QA), dan divisi terkait akan melakukan langkah-langkah yang diperlukan. Setelah perbaikan selesai dan aplikasi siap diuji ulang, tim QA akan melakukan pengujian untuk memastikan masalah tersebut telah teratasi. Pada pengujian otomatis, kita hanya perlu menjalankan kode yang telah dibuat, dan terdapat bot yang secara otomatis memberi tahu jika pengujian gagal. Notifikasi terkait kegagalan tersebut akan dikirimkan melalui Slack, memungkinkan tim untuk segera mengetahui dan menangani masalah yang muncul. Hal ini mempermudah koordinasi antar tim dan mempercepat proses perbaikan, sehingga pengujian dapat terus berlanjut dengan lebih efisien.

Pada pengujian otomatis untuk aplikasi *mobile*, *framework* yang digunakan adalah WebDriverIO dengan JavaScript, yang nantinya akan terhubung ke Appium untuk menjalankan proses otomatisasi pengujian pada aplikasi *mobile*. WebDriverIO berfungsi untuk menulis skrip pengujian, sedangkan Appium akan mengeksekusi pengujian pada perangkat *mobile*. Untuk pengujian otomatis pada *website* Jack, digunakan Cypress, sebuah *framework* yang memungkinkan pengujian *front-end* dilakukan secara efisien. Kode pengujian disimpan dan dikelola melalui Bitbucket, yang berfungsi sebagai repositori untuk menyimpan skrip pengujian dan memudahkan kolaborasi tim dalam pengelolaan kode.

3.2.2 Uraian Kerja Magang

Selama periode magang di PT. Indo Koala Remittance, yang berlangsung dari 16 September 2024 hingga Desember 2024, terdapat keterlibatan dalam berbagai proyek yang sedang berlangsung di perusahaan. Pekerjaan dilakukan sesuai dengan proyek-proyek yang ada, memberikan

kesempatan untuk memperoleh pengalaman praktis yang berharga dalam dunia kerja.

Informasi lengkap mengenai tugas magang dapat ditemukan dalam Tabel 3.2. Tabel ini menyajikan rincian lengkap tentang aktivitas yang dilakukan selama periode magang, termasuk proyek-proyek yang dikerjakan, tanggung jawab yang diemban, serta pencapaian yang telah diraih. Melalui pengalaman magang ini, diharapkan dapat memperoleh pemahaman yang lebih dalam tentang proses kerja di PT. Indo Koala Remittance dan mengembangkan keterampilan serta pengetahuan yang sesuai dengan industri tersebut. Hal ini akan menjadi dasar yang kuat untuk pengembangan karir di masa depan.

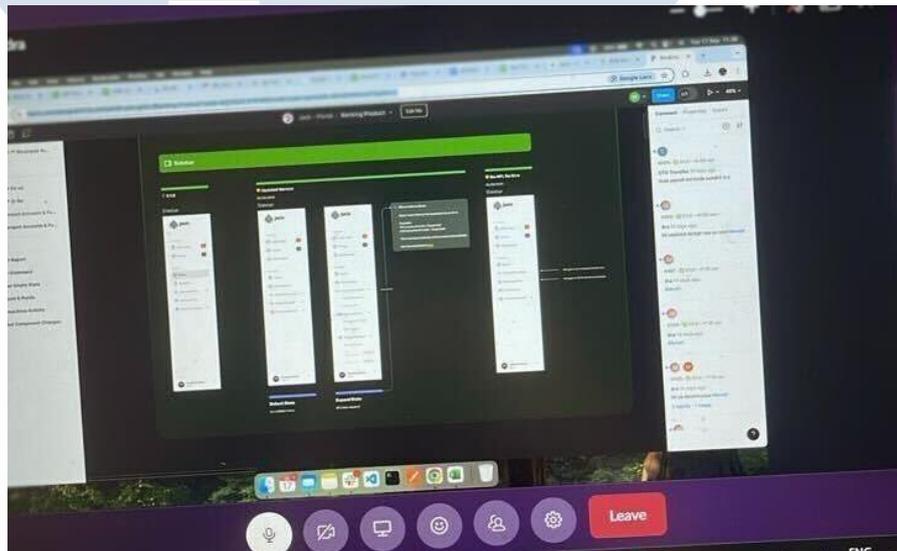
Tabel 3.2 Uraian Kerja Magang

Tanggal	Uraian Kerja Magang
16 September 2024	Pengenalan Perusahaan, Mempelajari <i>user guide</i> , instalasi, dan mempelajari <i>query-query</i>
17 - 30 September 2024	Melakukan <i>full cycle rollout</i> aplikasi beserta fitur pada aplikasi Jack di Portal, <i>Mobile</i> dan Transfez di <i>Mobile</i>
1 Oktober - 11 November 2024	<i>Project Localization Mobile App Jack</i>
12 November - 27 Desember 2024	<i>Project Points Reward</i> pada Aplikasi Transfez

Tabel 3.2 menampilkan informasi mengenai tanggal-tanggal spesifik dan rincian pekerjaan magang yang dilakukan selama periode magang di PT. Indo Koala Remittance. Setiap baris dalam tabel menggambarkan tanggal tertentu beserta uraian pekerjaan yang dikerjakan pada tanggal tersebut.

3.2.2.1 Pengenalan Perusahaan dan Tools yang digunakan

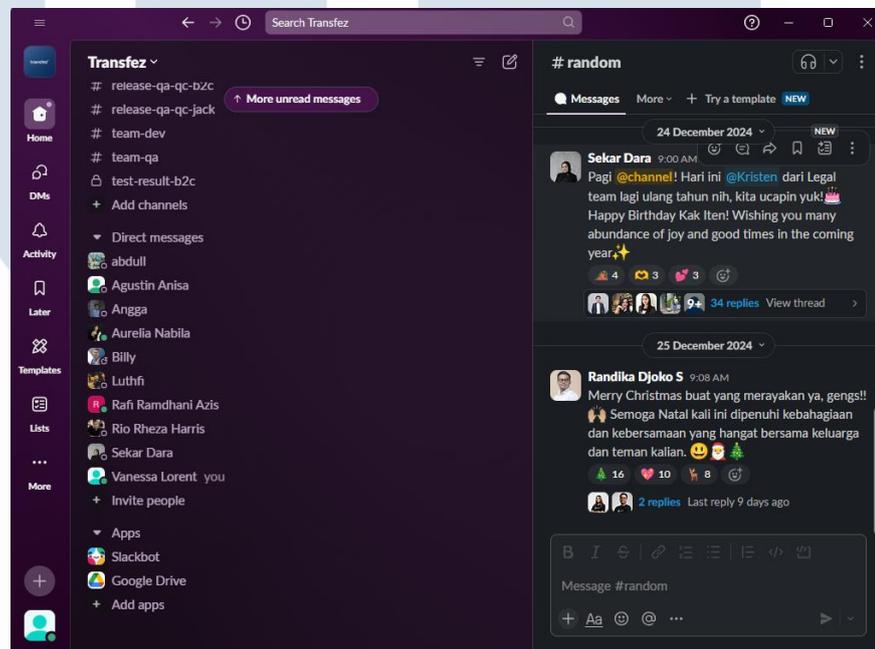
Pada awal magang, kegiatan dimulai dengan sesi *onboarding* bersama HR yang bertujuan untuk mengenalkan perusahaan, struktur organisasi, dan divisi-divisi yang ada di dalamnya. Selama *onboarding* ini, dijelaskan pula mengenai visi, misi, nilai-nilai perusahaan, serta tujuan dan budaya kerja yang diterapkan. Setelah sesi tersebut, *onboarding* dilanjutkan dengan sesi bersama mentor QA dan supervisi tim teknis. Gambar 3.7 menampilkan kegiatan onboarding dengan mentor QA dan supervisi dimana dalam sesi ini, diperkenalkan lebih lanjut mengenai tim *Quality Assurance*, tugas dan tanggung jawab yang dijalankan, serta proses pengujian aplikasi yang diterapkan di perusahaan.



Gambar 3.7 Onboarding

Selanjutnya, diperkenalkan juga secara rinci mengenai proses dan *flow* kerja tim QA. Proses pengujian dimulai dengan pemahaman terhadap kebutuhan aplikasi, kemudian dilanjutkan dengan perencanaan pengujian, pelaksanaan pengujian, hingga pelaporan hasil pengujian. Setiap langkah memiliki prosedur dan standar tertentu untuk memastikan aplikasi berjalan sesuai dengan

harapan. Selain itu, diperkenalkan berbagai *tools* yang digunakan dalam pengujian, seperti VScode, *framework* Cypress untuk otomatisasi pengujian menggunakan bahasa pemrograman JavaScript, serta *tools manual testing* untuk memverifikasi fungsionalitas aplikasi. Pengenalan ini bertujuan untuk mempersiapkan peserta magang agar dapat beradaptasi dengan alur kerja yang ada dan menjalankan tugas-tugas pengujian dengan efektif.

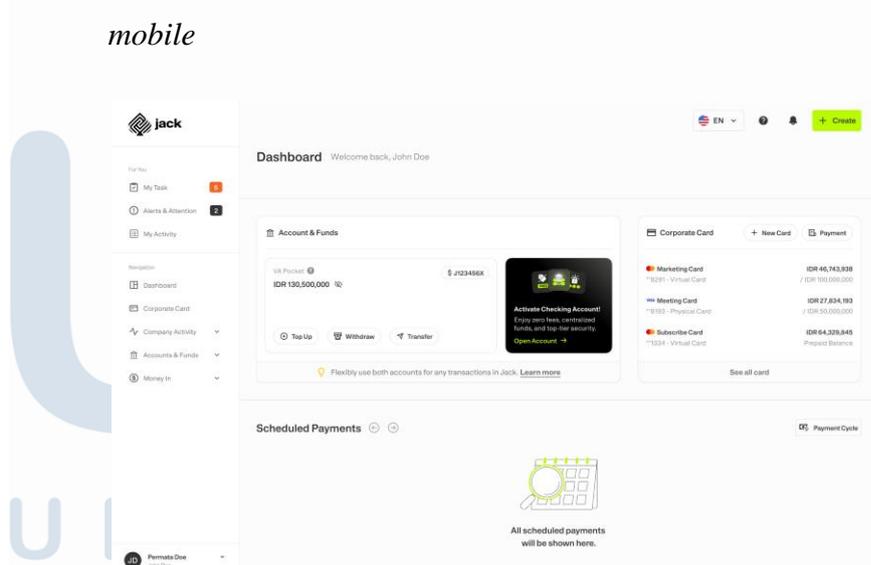


Gambar 3.8 Tampilan Slack

Untuk komunikasi internal antar tim, perusahaan menggunakan aplikasi Slack sebagai *platform* utama yang dapat dilihat tampilannya pada Gambar 3.8. Slack memungkinkan tim untuk berkomunikasi secara *real-time*, berbagi informasi, mendiskusikan progres pekerjaan, dan mengatur berbagai saluran diskusi yang relevan untuk setiap topik atau proyek yang sedang dikerjakan. Melalui Slack, anggota tim dapat dengan mudah bertukar pesan, melakukan panggilan suara atau video, serta berbagi *file* atau dokumen penting secara langsung di dalam saluran tertentu,

yang memudahkan kolaborasi tanpa hambatan. Aplikasi ini juga memungkinkan pengelompokan percakapan dalam saluran (*channel*) khusus berdasarkan tim, proyek, atau topik tertentu, sehingga komunikasi tetap terorganisir dan mudah diakses. Slack juga dilengkapi dengan berbagai fitur seperti integrasi dengan *tools* lain, notifikasi, dan pencarian yang cepat, yang membantu tim dalam melacak percakapan atau dokumen yang diperlukan, serta memastikan bahwa tidak ada informasi penting yang terlewatkan. Penggunaan Slack ini mendukung koordinasi yang lebih efisien antar anggota tim dan mempermudah proses berbagi pembaruan atau dokumen terkait pekerjaan yang sedang berlangsung, baik di tim QA maupun dengan tim lain dalam perusahaan.

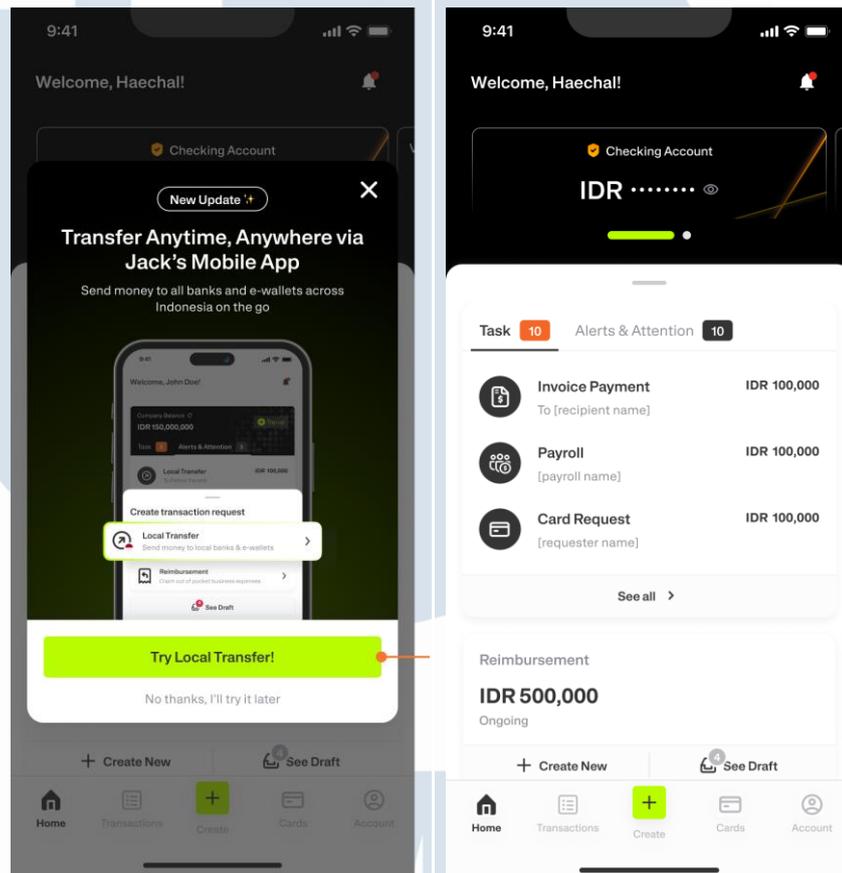
Setelah proses *onboarding* dan pelatihan selesai, langkah selanjutnya adalah melakukan eksplorasi atau pengenalan kepada aplikasi Jack dan Transfez. Untuk aplikasi Jack, terdapat dua *platform* yang perlu diperhatikan, yaitu *portal web* dan aplikasi *mobile*



Gambar 3.9 Tampilan Website Jack

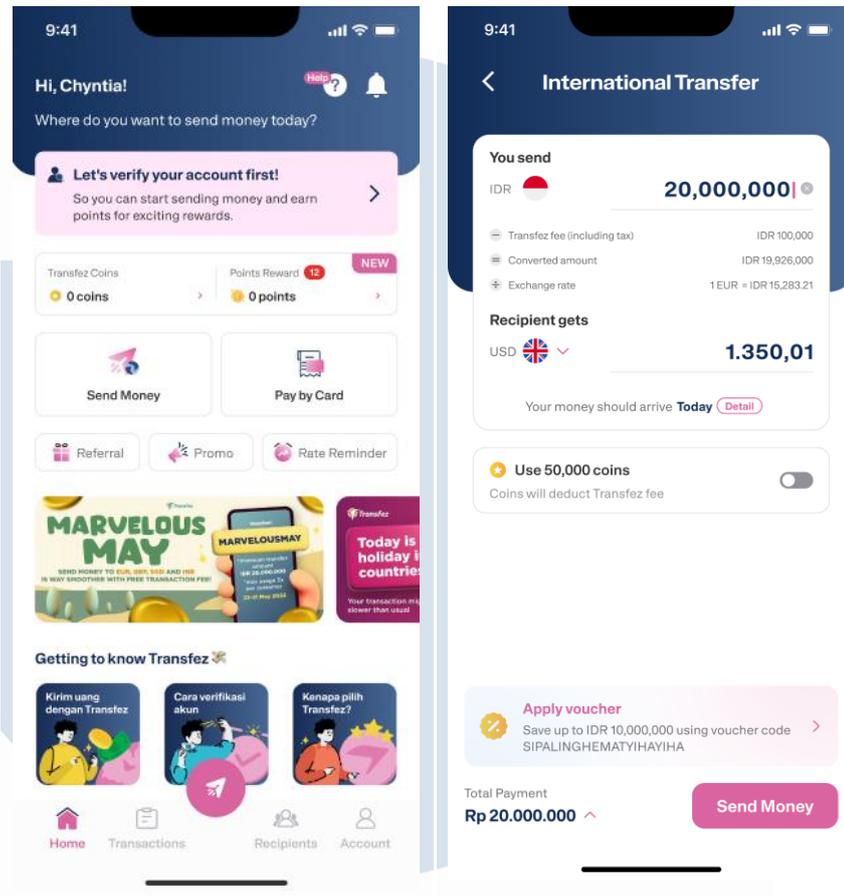
Gambar 3.9 menampilkan gambaran *website* Jack. *Portal web* Jack dirancang untuk memberikan akses yang lebih lengkap dan

komprehensif bagi pengguna yang menginginkan pengalaman melalui perangkat *desktop* atau laptop. Pengguna dapat mengakses berbagai fitur, seperti manajemen transaksi, laporan, dan pengaturan akun.



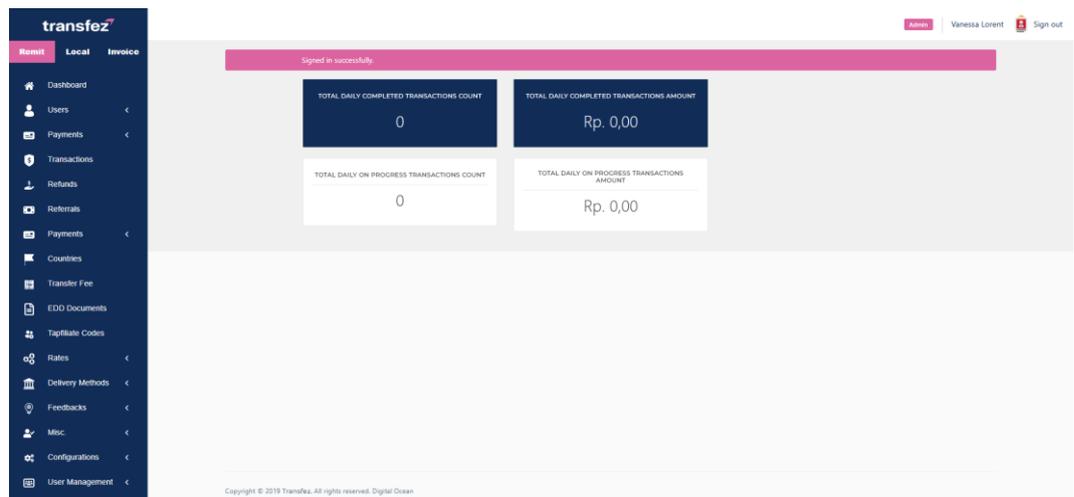
Gambar 3.10 Tampilan Aplikasi *Mobile Jack*

Sementara itu, aplikasi *mobile Jack* yang ditampilkan pada Gambar 3.10 ditujukan untuk memberikan kemudahan bagi pengguna yang lebih sering beraktivitas di perangkat seluler, sehingga semua fitur utama yang ada pada portal juga tersedia dalam bentuk yang lebih ramah untuk penggunaan *mobile*. QA perlu memastikan bahwa kedua *platform* ini, baik portal maupun aplikasi *mobile*, bekerja dengan mulus, memiliki performa yang baik, dan bebas dari *bug* di semua skenario penggunaan.



Gambar 3.11 Tampilan Aplikasi Mobile Transfez

Gambar 3.11 menampilkan aplikasi mobile transfez. Dimana pada aplikasi Transfez, QA akan fokus pada aplikasi *mobile*, yang dirancang untuk memudahkan pengguna dalam melakukan berbagai transaksi keuangan langsung melalui perangkat mereka. Aplikasi *mobile* Transfez ini memungkinkan pengguna untuk melakukan berbagai tindakan seperti pengiriman uang, pelacakan transaksi, dan manajemen akun secara *real-time*. Selain itu, Transfez juga memiliki portal *back office* yang ditampilkan pada Gambar 3.12 dimana digunakan oleh tim QA untuk melakukan pengujian pada tahap staging.

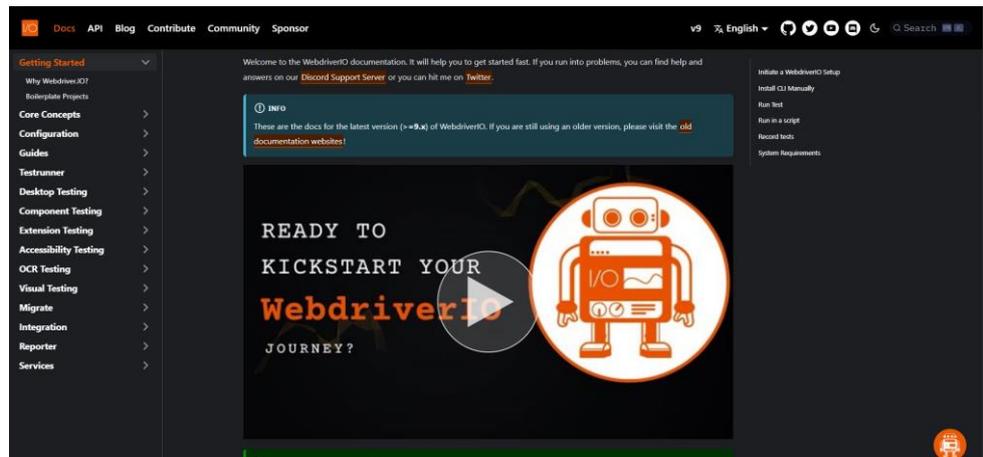


Gambar 3.12 Portal *Back Office* Transfez

Portal *back office* ini memungkinkan QA untuk mengakses dan memantau proses *backend* yang tidak dapat diakses langsung oleh pengguna aplikasi *mobile*, memberikan visibilitas yang lebih dalam terhadap sistem dan fungsi yang sedang diuji. QA akan memeriksa alur kerja, integrasi sistem, dan kinerja aplikasi untuk memastikan bahwa setiap transaksi dan interaksi yang dilakukan pengguna melalui aplikasi *mobile* Transfez berjalan dengan lancar dan aman.

Pada Divisi QA sendiri menggunakan berbagai *tools* untuk menjalankan tugasnya, di antaranya WebDriverIO, Cypress, Appium, Android Studio, BrowserStack, Google Docs, Google Sheets, dan Bitbucket. Berikut ini adalah penjelasan lebih mendalam terkait aplikasi atau *tools* yang sering digunakan dalam proses kerja magang:

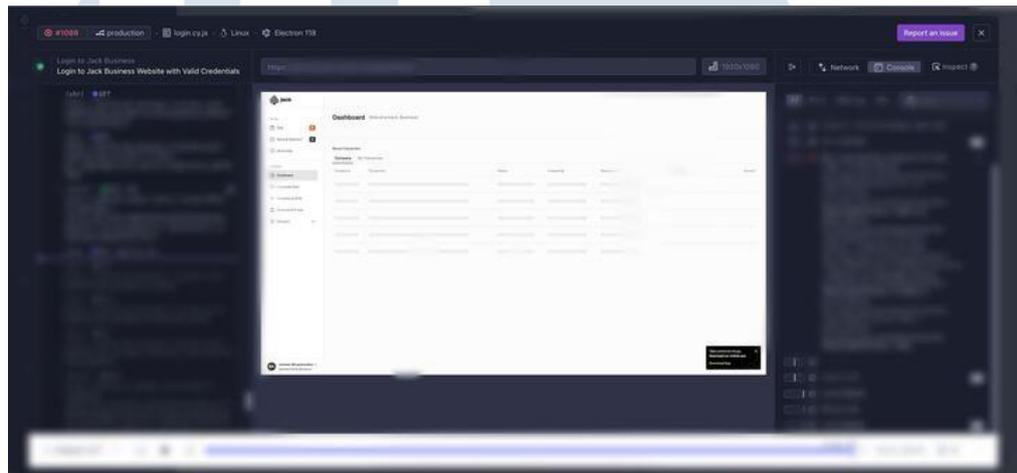
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.13 WebdriverIO

Gambar 3.13 menampilkan *website* dari WebDriverIO. WebDriverIO adalah *framework* otomatisasi pengujian berbasis JavaScript yang digunakan untuk melakukan pengujian terhadap aplikasi *web* dan *mobile*. Framework ini sangat fleksibel, mendukung berbagai jenis pengujian seperti pengujian fungsional, *end-to-end*, dan integrasi. Dengan API yang mudah dipahami dan digunakan, WebDriverIO memungkinkan tim QA untuk berinteraksi dengan elemen-elemen aplikasi, melakukan navigasi, serta memverifikasi hasil yang diinginkan. Di perusahaan ini, WebDriverIO digunakan bersama dengan *framework* Mocha untuk menulis dan menjalankan tes. *Framework* ini digunakan terutama untuk pengujian aplikasi *mobile*, memungkinkan tim QA untuk mengotomasi pengujian pada perangkat Android dan iOS dengan efisien. WebDriverIO diintegrasikan dengan Appium, sebuah *tool open-source* untuk otomatisasi pengujian aplikasi *mobile*, yang memungkinkan pengujian aplikasi mobile Transfez dapat dilakukan dengan lebih efektif. Dengan pengaturan ini, tim QA dapat melakukan pengujian *end-to-end* pada aplikasi *mobile* Transfez, memastikan bahwa aplikasi berjalan dengan lancar di berbagai perangkat dan sistem operasi, serta meningkatkan kualitas dan keandalan aplikasi yang diuji. Integrasi antara WebDriverIO dan

Mocha juga memastikan bahwa pengujian yang dilakukan memiliki struktur yang jelas dan dapat diandalkan, serta memungkinkan pengujian yang lebih mudah dipelihara dan diperluas seiring dengan perkembangan aplikasi.



Gambar 3.14 Tampilan Cypress

Gambar 3.14 menampilkan tampilan Cypress. Cypress adalah *framework* pengujian aplikasi web yang memudahkan tim QA dalam menulis dan menjalankan pengujian *end-to-end*. Salah satu fitur utama Cypress adalah kemampuannya untuk melakukan pengujian secara *real-time*, yang memungkinkan tim untuk memonitor dan *mendebug* tes langsung saat pengujian dilakukan. Cypress juga memiliki kemampuan untuk menjalankan pengujian secara otomatis pada berbagai kondisi yang berbeda, seperti koneksi internet yang lambat atau perangkat yang berbeda. Selain itu, Cypress dilengkapi dengan alat bantu yang memungkinkan pengujian lebih efektif, seperti *snapshot* dan video dari setiap tes yang dijalankan, yang berguna untuk analisis hasil pengujian.

Framework ini nantinya akan digunakan untuk melakukan pengujian pada *web* portal Jack, yang merupakan salah satu platform utama yang digunakan oleh pengguna untuk mengakses berbagai

layanan. Pengujian dengan Cypress pada *web* portal Jack akan mencakup berbagai skenario *end-to-end* untuk memastikan bahwa setiap elemen pada portal, seperti formulir transaksi, manajemen akun, dan fitur laporan, berfungsi dengan baik. Kemampuan Cypress untuk memonitor pengujian secara *real-time* juga akan sangat berguna dalam memverifikasi performa portal, memastikan bahwa pengguna mendapatkan pengalaman yang lancar dan bebas dari gangguan teknis. Dengan fitur-fitur seperti snapshot dan video tes, tim QA dapat lebih mudah menganalisis dan memperbaiki *bug* yang ditemukan, sehingga meningkatkan kualitas dan keandalan *web* portal Jack secara keseluruhan.



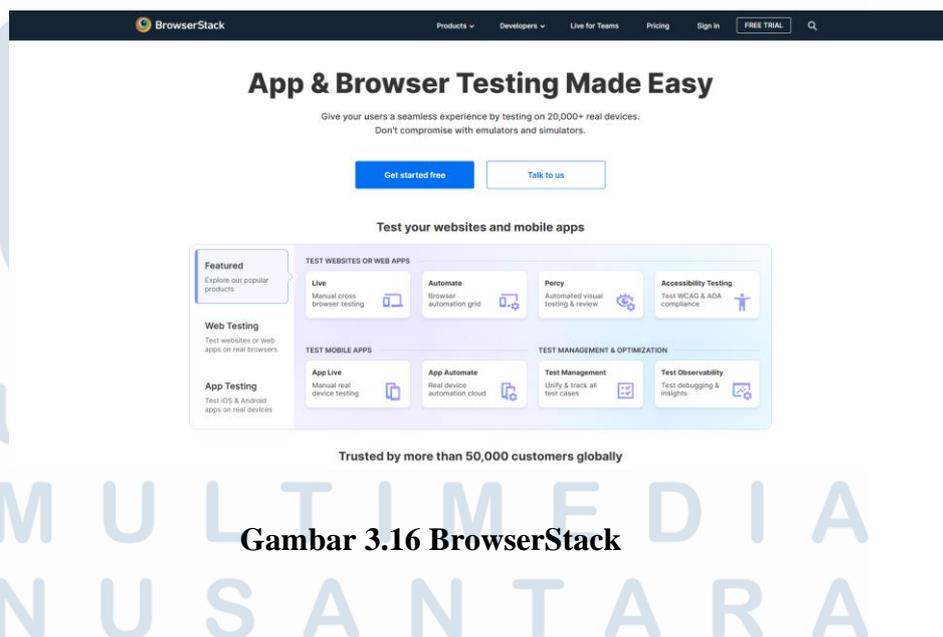
Gambar 3.15 Appium

Appium adalah *framework* otomatisasi pengujian yang dirancang untuk menguji aplikasi *mobile* pada berbagai *platform*, termasuk Android dan iOS. Tampilan Appium dapat dilihat pada Gambar 3.15. Appium dapat digunakan untuk menjalankan otomatisasi pengujian pada perangkat fisik (*real devices*) maupun *emulator* atau *simulator*. Appium memungkinkan integrasi dengan

WebDriverIO, sehingga kode yang ditulis sebelumnya menggunakan WebDriver dapat dijalankan pada perangkat *mobile*.

Untuk pengujian aplikasi *mobile*, Appium akan mengambil alih eksekusi otomatisasi di *real device*, baik itu perangkat Android atau iOS, dan menjalankan skrip pengujian yang ditulis menggunakan WebDriverIO. Appium menggunakan Android Studio atau Xcode untuk berinteraksi dengan perangkat tersebut. Dengan cara ini, tim QA dapat melakukan pengujian pada perangkat fisik dengan lebih akurat, meniru interaksi pengguna nyata pada aplikasi *mobile*.

Dengan memanfaatkan WebDriver yang sudah ditulis sebelumnya, Appium akan memanfaatkan kode yang ada untuk berinteraksi dengan elemen-elemen aplikasi di *real device* atau *emulator*. Hal ini memungkinkan otomatisasi pengujian yang lebih efisien dan memastikan bahwa aplikasi *mobile* berfungsi dengan baik pada perangkat yang sebenarnya, serta memberikan kepercayaan bahwa aplikasi dapat bekerja sesuai dengan ekspektasi pada berbagai kondisi penggunaan.



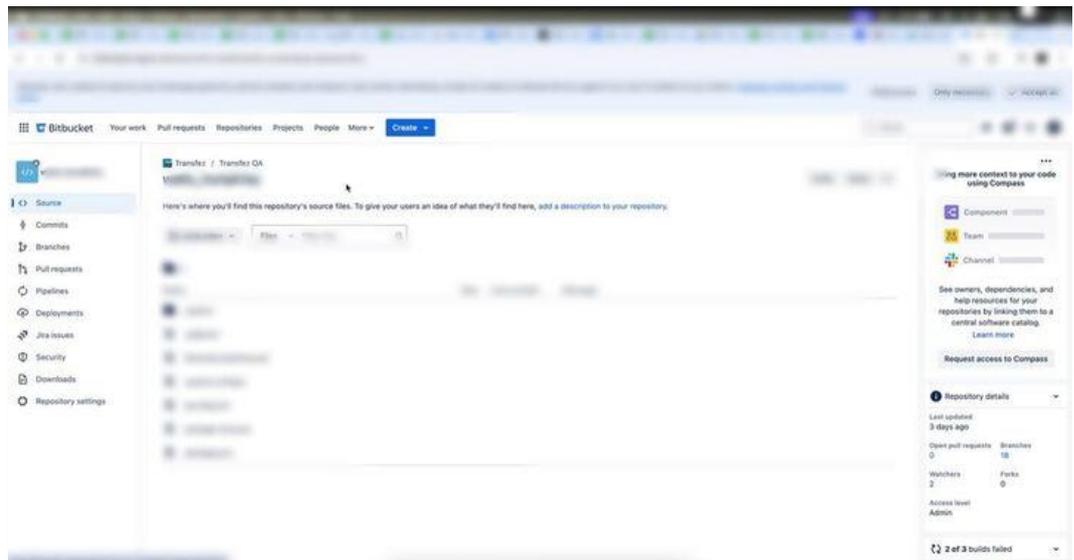
Gambar 3.16 BrowserStack

Gambar 3.16 merupakan tampilan BrowserStack. BrowserStack adalah *platform cloud-based* yang memungkinkan tim QA untuk menguji aplikasi *web* dan *mobile* pada berbagai perangkat dan browser yang berbeda tanpa perlu memiliki perangkat fisik secara langsung. Dengan BrowserStack, pengujian dapat dilakukan pada berbagai *real devices* yang tersedia di *cloud*, baik itu perangkat Android, iOS, maupun *desktop* dengan berbagai versi browser.

Untuk pengujian aplikasi *mobile*, BrowserStack menyediakan perangkat *mobile* fisik yang dapat digunakan untuk menjalankan pengujian otomatis atau manual. Tim QA dapat mengakses perangkat Android dan iOS yang berbeda, serta melakukan pengujian pada berbagai kondisi, seperti versi sistem operasi yang berbeda atau resolusi layar yang berbeda, tanpa harus memiliki perangkat tersebut secara fisik.

Selain itu, BrowserStack juga memungkinkan pengujian aplikasi *web* pada berbagai browser dan perangkat *desktop*. Platform ini menyediakan berbagai macam browser dengan versi yang berbeda, memungkinkan pengujian untuk memastikan kompatibilitas lintas browser dan *platform*. Dengan menggunakan BrowserStack, tim QA dapat melakukan pengujian secara lebih efisien dan lebih cepat, karena tidak perlu menyiapkan perangkat atau *browser* secara fisik, dan dapat melaksanakan pengujian pada berbagai kombinasi perangkat dan *browser* secara simultan di *cloud*.

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



Gambar 3.17 BitBucket

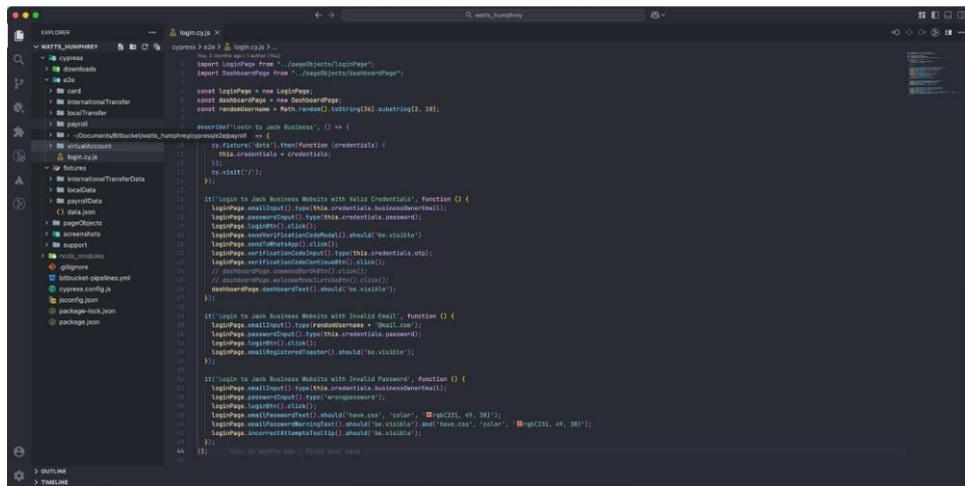
Gambar 3.17 menampilkan tampilan Bitbucket. Bitbucket adalah *platform* untuk manajemen kode sumber yang memungkinkan tim pengembang untuk mengelola repositori *Git* mereka secara efisien. Dengan Bitbucket, tim QA dapat mengakses repositori yang berisi kode otomatisasi pengujian dan melakukan pengujian pada aplikasi yang sedang dikembangkan. Repositori ini dapat mencakup skrip pengujian, konfigurasi pengujian, serta dokumentasi terkait.

Selain itu, Bitbucket menyediakan berbagai fitur yang mendukung kolaborasi dan pengelolaan proyek, termasuk kemampuan untuk mengatur dan menjadwalkan pengujian secara otomatis. Salah satu fitur penting yang dapat dimanfaatkan adalah integrasi dengan *Continuous Integration (CI) tools* seperti *Bitbucket Pipelines*. Dengan *Bitbucket Pipelines*, tim QA dapat menjadwalkan pengujian untuk dijalankan secara otomatis setiap kali ada perubahan pada repositori, seperti *commit* baru atau *pull request*. Hal ini memungkinkan pengujian dilakukan secara terjadwal dan konsisten, serta mengurangi risiko terjadinya regresi dalam aplikasi.

Pengujian yang dijadwalkan ini dapat mencakup berbagai jenis tes, mulai dari tes fungsional, integrasi, hingga pengujian *end-to-end*, dan dapat dijalankan di berbagai lingkungan, seperti perangkat nyata (*real devices*), *emulator*, atau *browser cloud* seperti BrowserStack. Dengan Bitbucket, tim QA dapat memastikan bahwa aplikasi diuji secara berkelanjutan setiap kali ada pembaruan, sehingga kualitas aplikasi tetap terjaga dan bug dapat ditemukan lebih awal dalam siklus pengembangan.

3.2.2.2 Fullcycle Rollout aplikasi Jack dan Transfez

Setelah proses *onboarding* dan pengenalan alat serta lingkungan pengujian selesai, langkah selanjutnya adalah melakukan *fullcycle rollout* pada aplikasi Jack dan Transfez. Pada aplikasi Transfez, pengujian dimulai dengan menggunakan WebDriverIO sebagai *framework* utama untuk menulis dan menjalankan skrip pengujian otomatisasi menggunakan JavaScript. Skrip pengujian ini kemudian dikirim ke Appium, yang berfungsi untuk menjalankan otomatisasi pengujian pada perangkat *mobile*. Appium, meskipun berbasis Java, dapat terintegrasi dengan WebDriverIO, sehingga tim QA dapat menjalankan pengujian otomatisasi pada perangkat Android maupun iOS. Pengujian di perangkat Android dilakukan melalui Android Studio, sedangkan untuk perangkat iOS, pengujian dijalankan melalui Xcode. Proses ini memastikan aplikasi Transfez dapat diuji di perangkat fisik, dengan berbagai konfigurasi perangkat dan sistem operasi, memberikan gambaran yang lebih realistis tentang performa aplikasi di dunia nyata.



Gambar 3.18 Tampilan Codingan Automation

Pada *website* Jack, pengujian dilakukan menggunakan Cypress untuk memastikan bahwa *web* portal berjalan dengan baik di berbagai *browser* dan perangkat. Untuk pengujian lintas *browser*, saat ini digunakan BrowserStack, *platform cloud* yang menyediakan perangkat nyata dan *emulator* untuk menguji aplikasi pada berbagai konfigurasi perangkat dan *browser*. Pengujian otomatisasi di Cypress dilakukan langsung di *browser* tanpa menggunakan *middleware*, sehingga proses pengujian berjalan lebih efisien. Gambar 3.18 menunjukkan implementasi pengujian otomatis menggunakan Cypress untuk menguji fitur *login* pada sebuah situs *web* Jack. Di bagian kiri, terdapat struktur direktori yang berisi *file* dan *folder* terkait, seperti `fixtures` untuk menyimpan data uji, `pageObjects` untuk mendefinisikan elemen halaman, serta *file* konfigurasi Cypress.

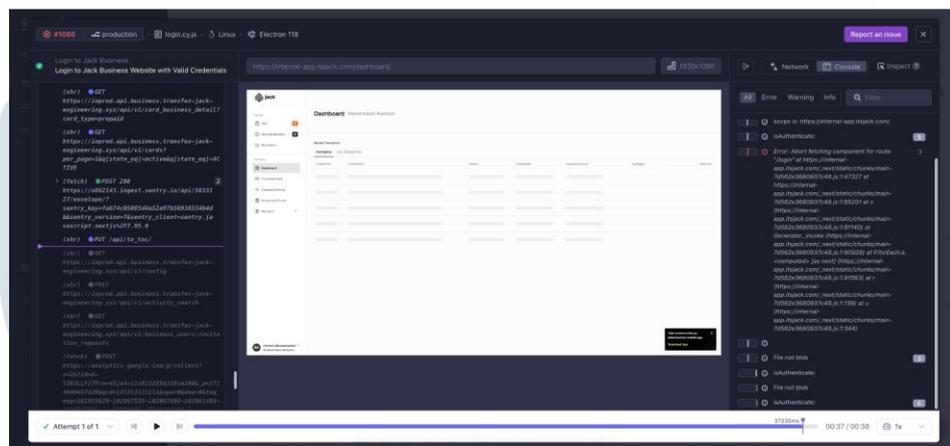
Kode pengujian di dalam *file* `login.cy.js` mencakup beberapa skenario pengujian, antara lain:

1. Login dengan kredensial valid: Memastikan pengguna dapat masuk dengan email dan password

yang benar, kemudian memverifikasi elemen dashboard muncul setelah login.

2. Login dengan email tidak valid: Menggunakan email acak yang salah, dan memeriksa apakah muncul pesan peringatan yang sesuai.
3. Login dengan password salah: Memasukkan password yang salah untuk email yang valid, dan memverifikasi bahwa sistem menampilkan peringatan terkait.

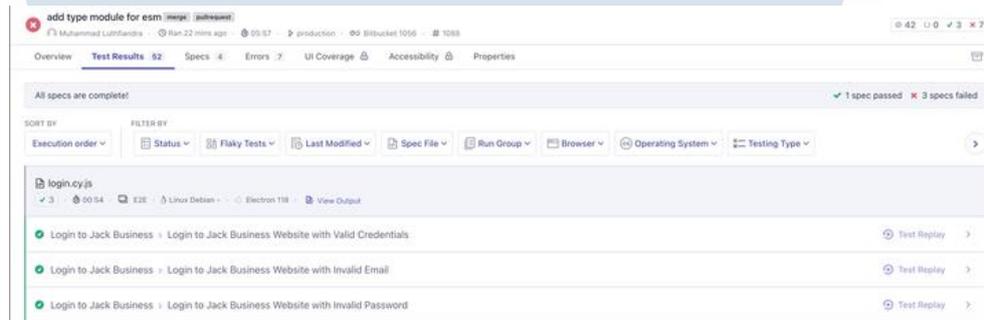
Setiap pengujian menggunakan fungsi-fungsi Cypress seperti `.type()` untuk memasukkan teks ke dalam *input field*, `.click()` untuk menekan tombol, dan `.should()` untuk memverifikasi bahwa elemen tertentu terlihat di halaman atau memiliki atribut tertentu. Dengan pendekatan ini, pengujian dilakukan secara terstruktur dan otomatis untuk memvalidasi berbagai skenario *login*.



Gambar 3.19 Tampilan Cypress ketika di Run

Gambar 3.19 menunjukkan contoh hasil pengujian otomatis menggunakan Cypress pada fitur *login* situs *web* Jack dengan kredensial yang valid. Di bagian kiri, terlihat *log* aktivitas jaringan (*network*) seperti permintaan **GET** dan **POST** ke berbagai *endpoint*

API yang digunakan dalam proses *login*, seperti autentikasi, konfigurasi, dan pengambilan data untuk *dashboard*. Di tengah, tampilan *dashboard* menunjukkan halaman berhasil dimuat setelah *login* berhasil, menandakan pengujian valid. Bagian kanan menampilkan konsol dengan beberapa *log* kesalahan, salah satunya adalah *error* terkait "*Abort fetching component for route '/login'.*" Namun, *error* ini tampaknya tidak memengaruhi hasil akhir pengujian. Waktu total pengujian tercatat sekitar 37 detik, yang diindikasikan di bagian bawah. Hal ini menunjukkan proses *login* telah diuji dan berjalan sesuai harapan.

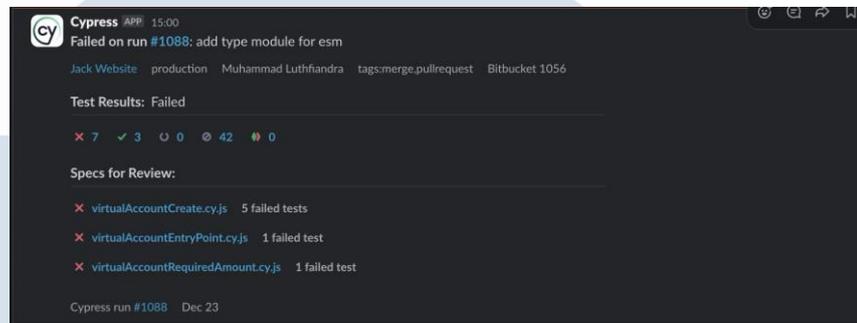


Gambar 3.20 Hasil Pengujian Pada Cypress Dashboard

Gambar 3.20 menunjukkan hasil pengujian otomatis di Cypress *Dashboard* untuk file *login.cy.js*. Terdapat tiga skenario pengujian yang dilakukan, yaitu *login* dengan kredensial valid, *login* dengan email tidak *valid*, dan *login* dengan *password* yang salah. Dari ketiga skenario tersebut, hanya pengujian *login* dengan kredensial valid yang berhasil, sedangkan dua lainnya gagal.

Skrip pengujian disimpan di Bitbucket, yang memungkinkan tim QA untuk mengelola repositori kode pengujian dan menjadwalkan eksekusi otomatis menggunakan Bitbucket *Pipelines*. Setelah pengujian dijalankan, hasilnya akan dipantau melalui Cypress *Cloud*, dan status hasil pengujian akan dikirim ke Slack *Bot* yang dapat dilihat pada Gambar 3.21, memberikan

pembaruan langsung mengenai apakah pengujian berhasil atau gagal.



Gambar 3.21 Hasil Notifikasi Cypress di Slack

Proses *fullcycle rollout* ini menguji aplikasi dari awal hingga akhir pada kedua platform, baik di perangkat *mobile* untuk Transfez maupun di *web* portal untuk Jack. Dengan menggunakan alat seperti Appium, Cypress, Bitbucket, dan BrowserStack, tim QA dapat memastikan bahwa aplikasi berjalan dengan baik di berbagai perangkat dan lingkungan, serta mengidentifikasi dan memperbaiki *bug* sebelum aplikasi dirilis kepada pengguna akhir. Melalui pendekatan *fullcycle* ini, aplikasi Jack dan Transfez dapat diuji secara menyeluruh dan dipastikan memiliki kualitas yang optimal sebelum diluncurkan.

3.2.2.3 Project Localization Mobile App Jack

Proyek Lokalisasi Aplikasi Jack adalah upaya untuk menyesuaikan aplikasi agar dapat digunakan dengan baik oleh pengguna dari berbagai lokasi dan latar belakang budaya. Fokus utamanya adalah menambahkan opsi pilihan Bahasa Indonesia pada aplikasi untuk meningkatkan pengalaman pengguna di Indonesia. Proyek ini bertujuan untuk memastikan bahwa aplikasi dapat diakses dengan mudah tanpa hambatan bahasa atau perbedaan budaya.

TC ID	Test Case	PreCondition	Test Type	Test Steps	Expected Result	Staging	Test Status	Production	Notes
Project Name: Localization Language (ID) Mobile App					Total Test Case	411			
Test Case Designed by: Vanessa					Positive	398			
Test Case Reviewed by: Luthfi					Negative	13			
TC_LC2_APP_001	[Account] Klik menu "Akun" pada tool bar	- User is on homepage	Positive		- Direct to "Akun" page - The copy and design of the "Account" page match the specifications in Figma	PASSED	PASSED		
TC_LC2_APP_002	[Account] Change profile page	- User is on the Account page	Positive	1. Click on "Ubah Profil"	- The user is directed to the "Ubah Profil" page - Test on green button "Simples"	PASSED	PASSED		
TC_LC2_APP_003	[Account] Verify Name Changes	- The user is on the "Ubah Profil" page	Positive	1. Update the name field with a new name.	- The "Simples" button text changes to "Kembalikan"	SKIPPED	SKIPPED		
TC_LC2_APP_004	[Account] Verify Name Change Confirmation and PIN Page Navigation		Positive		- The user is directed to the "PIN Jack" page - A PIN input box is displayed - The copy and design should match the specifications in Figma - A loading indicator is displayed	PASSED	PASSED		
TC_LC2_APP_005	[Account] Verify loading screen, profile update, and success toast after PIN confirmation		Positive	1. Enter PIN Correctly	- The user is redirected back to the "Account" page with the updated name - A green toast notification appears, showing the message "Informasi profil berhasil diupdate"	PASSED	PASSED		
TC_LC2_APP_006	[Account] Klik opsi "Kembalikan Akun" pada Account page	- The user is on the "Account" page	Positive	1. Click on "Kembalikan Akun"	- The user is directed to the "Kembalikan Akun" page - Two options are displayed: "Ubah Password" and "Ubah PIN"	PASSED	PASSED		
TC_LC2_APP_007	[Account] Change password of page		Positive	1. Click on "Ubah Password"	- The user is directed to the "Ubah Password" page - A field is displayed for entering the current password	PASSED	PASSED		
TC_LC2_APP_008	[Account] Change Password- Wrong Current Password Validation		Negative	1. Enter an incorrect current password in the password field 2. Click the "Simples" button.	- A red line is displayed around the password input box indicating an error - An error message is shown stating that the password is invalid	PASSED	PASSED		
TC_LC2_APP_009	[Account] Change Password- Valid Current Password		Positive	1. Enter the correct current password in the password field. 2. Click the "Langsung" button.	- The user is directed to the "Masukkan Password Baru" (Enter New Password) page.	PASSED	PASSED		
TC_LC2_APP_010	[Account] Change Password- New Password Entry and Verification Code Navigation		Positive	1. Enter a new password in the "Password Baru" field. 2. Enter the same new password in the "Ukonfirmasi Password" (Confirm Password) field. 3. Click the "Ubah Password" button.	- The user is directed to the "Kode Verifikasi" (Verification Code) page upon clicking the button - The copy and design should match the specifications in Figma.	PASSED	PASSED		

Gambar 3.22 Testcase Localization Mobile App Jack

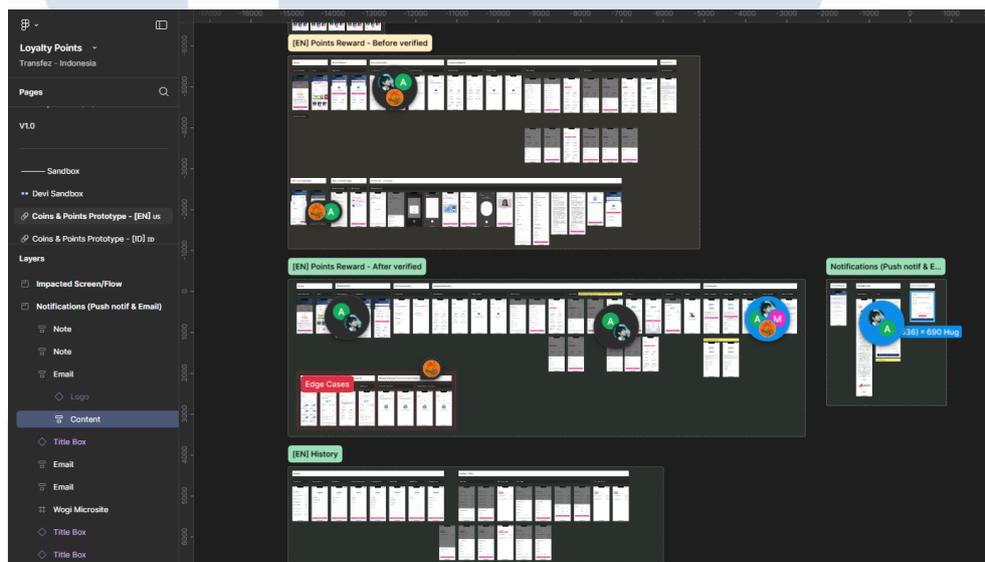
Pada Gambar 3.22, terdapat skenario yang digunakan sebagai dasar pengujian. Setelah skenario selesai disusun, langkah selanjutnya adalah memulai proses pengujian dengan menggunakan perangkat Android maupun iOS. Skenario yang disusun berisi serangkaian langkah atau tindakan yang perlu dijalankan oleh pengguna untuk memverifikasi fungsi atau fitur tertentu dari aplikasi atau sistem. Ini mencakup berbagai situasi penggunaan dalam aplikasi, termasuk pengujian fitur, navigasi antarmuka pengguna, dan respons sistem terhadap input pengguna.

Setelah skenario disiapkan, pengujian praktis dilakukan menggunakan perangkat Android dan iOS dengan menjalankan langkah-langkah yang telah ditetapkan. Selama pengujian, aplikasi akan diamati untuk memastikan bahwa fitur atau fungsinya berfungsi sebagaimana mestinya. Pengujian pada berbagai perangkat dan platform seperti Android dan iOS sangat penting untuk memastikan aplikasi dapat berjalan dengan baik di semua lingkungan dan perangkat pengguna. Hal ini memastikan pengalaman pengguna yang konsisten dan berkualitas, serta

menjamin aplikasi siap digunakan sebelum dirilis ke pengguna akhir.

3.2.2.4 *Project Points Reward* pada Aplikasi Transfez

Fitur *Points Reward* adalah sistem penghargaan yang memungkinkan pengguna mengumpulkan poin berdasarkan aktivitas tertentu, seperti transaksi atau interaksi dengan aplikasi, yang kemudian dapat ditukarkan dengan hadiah menarik. Sistem ini bertujuan untuk meningkatkan keterlibatan pengguna dengan memberikan insentif moneter melalui poin yang bisa ditukarkan dengan berbagai hadiah, seperti *cashback* atau *voucher*.



Gambar 3.23 Figma *Design* Fitur *Points Reward*

Gambar 3.23 di atas adalah tampilan Figma yang berisi rancangan fitur *Points Reward*. Figma ini digunakan sebagai acuan utama dalam proses pembuatan *test case* dan pengujian aplikasi. Desain yang ditampilkan membantu tim QA dan pengembang untuk memastikan bahwa seluruh alur, elemen UI, dan fungsionalitas dari fitur tersebut telah sesuai dengan spesifikasi yang direncanakan. Dengan mengacu pada desain ini, pengujian dilakukan untuk

memverifikasi bahwa implementasi di aplikasi nyata berjalan sesuai dengan yang dirancang, sehingga pengalaman pengguna tetap optimal.

TC ID	Test Case	Pre-Condition	Test Type	Test Steps	Expected Result	Tagging	Test Status	Production	Notes
					Total Test Case	89			
					Passed	63			
					Not passed	26			
TC_POINT_001	Show Popup for New Reward/Points Feature on First App Access	- User is logged into the app. - User's account is not yet verified. - App version supports the "Points Reward" feature.	Positive	1. User open the mobile app 2. User are navigated to home screen page 3. Popup bottomsheet reward info are shown	- Popup bottomsheet reward info should be shown. - Button "Go to Points Reward" can be clicked.	FAILED			
TC_POINT_002	Display Notification Prompt on Home Page for Unverified Users	- User is logged into the app. - User's account is not yet verified. - App version supports the "Points Reward" feature.	Positive	1. User open the mobile app 2. User are navigated to home screen page 3. Popup bottomsheet reward info are shown 4. User click button "Go to Points Reward" and popup bottomsheet are closed 5. Full homepage are shown and can be access	- A message or prompt asking the user to verify their account should be display.	FAILED			
TC_POINT_003	Display Points Reward Page for Unverified Users	- User is logged into the app. - User's account is not yet verified. - App version supports the "Points Reward" feature.	Positive	1. User open the mobile app 2. User are navigated to home screen page 3. Popup bottomsheet reward info are shown 4. User click button "Go to Points Reward" and popup bottomsheet are closed 5. Show label "New" in button Points Reward 6. Show label "New" in button Points Reward 7. User click points reward button and will be directed to Points Reward page	- User will be navigated to Points reward page. - A button message for user to verify their account should be display. - Label points should be shown. - Empty "You have no points activity" and description "Let see points by verifying your account first" should be display. - Button "Learn more points" can be click reward list with points, points reward, and clickable items should be display.	FAILED			
TC_POINT_004	Display Mission List Page when User Clicking on Earn More Points button	- User is logged into the app. - User's account is not yet verified. - App version supports the "Points Reward" feature.	Positive	1. User open the mobile app 2. User are navigated to home screen page 3. Popup bottomsheet reward info are shown 4. User click button "Go to Points Reward" and popup bottomsheet are closed 5. Full homepage are shown and can be accessed 6. Show label "New" in button Points Reward 7. User click points reward button and will be directed to Points Reward page 8. User click button earn more points	- The user should be directed to the mission page. - Information "The more you send, the more you earn" should be shown. - The earn 2 tab menu "Active" and "Expired" has mission with points reward, and every date should be shown.	FAILED			
TC_POINT_005	Display Empty states when there are no active mission list	- User is logged into the app. - User's account is not yet verified. - App version supports the "Points Reward" feature. - User is on page reward points. - There is no active mission list	Positive	1. User open the mobile app 2. User are navigated to home screen page 3. Popup bottomsheet reward info are shown 4. User click button "Go to Points Reward" and popup bottomsheet are closed 5. Show label "New" in button Points Reward 6. User click points reward button and will be directed to Points Reward page 7. User click button earn more points 8. User click button earn more points 9. User is on mission list page 10. Show 2 tab "Active" and "Expired" 11. User is on mission list page and there is no active mission.	- Empty state "Looks like you've completed all the missions" should be display and "There is there, more exciting missions are coming soon!" should be display.				

Gambar 3.24 Testcase Project Points Reward

Gambar 3.24 menampilkan *test case* yang digunakan dalam melakukan *testing* proyek *points reward*. *Test case* digunakan untuk mendokumentasikan skenario pengujian yang dirancang untuk memastikan bahwa fitur aplikasi berfungsi sesuai dengan spesifikasi yang ditentukan. Dalam pengujian, *test case* menjadi panduan utama yang menjelaskan langkah-langkah yang perlu dilakukan, kondisi awal yang harus dipenuhi, serta hasil yang diharapkan. Dengan adanya *test case*, tim QA dapat memvalidasi bahwa fitur berjalan sesuai dengan desain dan spesifikasi, mendeteksi jika ada *bug* atau masalah dalam implementasi, serta memastikan bahwa pengalaman pengguna tetap optimal. Selain itu, *test case* juga berfungsi sebagai dokumentasi resmi yang mencatat semua skenario pengujian untuk referensi atau kebutuhan audit di masa depan.

No	Issue	Priority	Issue Status	Assigned To	Steps to Reproduce (if Reward)	Video / Screenshots	Notes
1.	Banner verify pada page Points Reward tidak show (jurnal ifed user)	High	Retired Passed	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button and will be directed to Points Reward page	https://drive.google.com/file/d/1v3y4ur3q8M9g5WV9m9dC6k9v9m3u3o3v1e/view	
2.	Logo points pada page points reward belum sesuai	Medium	Retired Passed	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button and will be directed to Points Reward page	https://drive.google.com/file/d/19dTY_dR8B2rG2mpe8m0Dy1Qe4WwV9v3u3ed3v1e/view	
3.	Border corner points activity pada page Points Reward putus-putus	Medium	Retired Passed	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button and will be directed to Points Reward page	https://drive.google.com/file/d/1B69y4e78T1A1A7C6k9v9T1423v9m3u3o3v1e/view	
4.	Banner Verify di homepage terleak di bawah section points, seharusnya berada diatas section points	High	To Do	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page	https://drive.google.com/file/d/1u3v77k3B2c2u9483m0Dy1Qe4WwV9v3u3ed3v1e/view	
5.	Logo, Teks, dan warna pada banner verify di homepage belum sesuai figure	Medium	To Do	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page	https://drive.google.com/file/d/1u3v77k3B2c2u9483m0Dy1Qe4WwV9v3u3ed3v1e/view	
6.	Section poin berada di header	High	To Do	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page	https://drive.google.com/file/d/1u3v77k3B2c2u9483m0Dy1Qe4WwV9v3u3ed3v1e/view	
7.	Section rewards di halaman rewards point tidak show pada user yang belum verifikasi	High	To Do	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button and will be directed to Points Reward page	https://drive.google.com/file/d/1B69y4e78T1A1A7C6k9v9T1423v9m3u3o3v1e/view	
8.	Tidak Ada Keterangan "Setelah Nominal di Halaman Rewards	Medium	Retired Passed	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button 4. User on points reward page 5. User click see all rewards 6. User will direct to reward list page	https://drive.google.com/file/d/1D0Xv818v8v1u3v77k3B2c2u9483m0Dy1Qe4WwV9v3u3ed3v1e/view	
9.	Search by keyword pada halaman reward list namun tidak terjadi filtering	High	Retired Passed	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button 4. User on points reward page 5. User click see all rewards 6. User will direct to reward list page	https://drive.google.com/file/d/1K3R28B8v832_v9w8k83Cv9m3u3o3v1e/view	
10.	Angka pada Tab Explore Tidak Menampilkan Jumlah Reward yang Tersedia	High	Retired Passed	Rafi Ramdhani	1. User open Transfer app 2. User are navigated to home screen page 3. User click points reward button 4. User on points reward page 5. User click see all rewards 6. User will direct to reward list page	https://drive.google.com/file/d/1G7Dv9v81N88u3v9m3u3o3v1e/view	

Gambar 3.25 Bug List

Saat pengujian berlangsung, jika ditemukan *bug* atau *error* yang tidak seharusnya terjadi, masalah tersebut akan dicatat secara rinci dalam sebuah *bug list* dan dilaporkan kepada divisi terkait. Seperti pada Gambar 3.25 dimana *Bug list* adalah daftar yang berisi dokumentasi semua masalah atau *bug* yang ditemukan selama proses pengujian aplikasi. Setiap *bug* yang dicatat dilengkapi dengan deskripsi rinci untuk membantu tim pengembang memahami masalah dan menyelesaikannya. Selain itu, setiap *bug* juga diberi status prioritas untuk menentukan tingkat urgensi perbaikannya. Setelah *bug* atau *error* tersebut diperbaiki dan dinyatakan siap untuk diuji ulang, pengujian ulang (*re-test*) akan dilakukan guna memastikan masalah telah terselesaikan sepenuhnya.

3.3 Kendala yang Ditemukan

Selama menjalani program kerja magang di PT. Indo Koala Remittance, pastinya terdapat beberapa kendala, di antaranya:

- a. **Keterbatasan Komunikasi dan Koordinasi Terhambat Bekerja secara full WFH** mengurangi interaksi langsung dengan rekan tim,

yang membuat komunikasi dan koordinasi lebih terbatas. Meskipun sudah menggunakan berbagai platform komunikasi, kesulitan dalam memahami konteks atau arahan sering terjadi, serta koordinasi antar anggota tim menjadi lebih lambat karena perbedaan zona waktu dan pengaturan jadwal yang tidak selalu dapat disesuaikan.

b. **Penggunaan Tools Baru**

Selama magang, dihadapkan pada penggunaan beberapa *tools* baru yang sebelumnya belum pernah digunakan. Meskipun *tools* tersebut cukup efektif, mempelajari fungsionalitasnya membutuhkan waktu lebih lama, yang sempat menghambat kelancaran beberapa tugas. Adaptasi terhadap *tools* ini memerlukan pemahaman mendalam untuk dapat digunakan secara optimal.

3.4 Solusi atas Kendala yang Ditemukan

Untuk mengatasi kendala yang ditemukan selama program magang di PT. Indo Koala Remittance, beberapa langkah solusi telah diimplementasikan, di antaranya:

- a. Untuk mengurangi keterbatasan komunikasi, dengan mulai lebih aktif dalam memanfaatkan fitur-fitur dari *platform* komunikasi yang tersedia, seperti *video call dan chat*, agar dapat berinteraksi lebih langsung dengan rekan tim. Selain itu, mengusulkan pengaturan jadwal *meeting* secara lebih terstruktur dan memastikan bahwa setiap anggota tim memahami peran dan tanggung jawabnya. Hal ini membantu memperlancar alur kerja dan meminimalkan kendala terkait zona waktu.
- b. Untuk mengatasi kesulitan dalam penggunaan *tools* baru, diambil inisiatif untuk mempelajari materi dan tutorial yang disediakan oleh tim atau vendor *tools* tersebut. Selain itu, dengan berdiskusi dengan rekan yang lebih berpengalaman untuk mempercepat proses adaptasi. Dengan pendekatan

ini, berhasil mengoptimalkan penggunaan *tools* dan meningkatkan efisiensi dalam menjalankan tugas-tugas yang diberikan.

