

SKRIPSI

**AUTOMATION TESTING TOOL DALAM PENGUJIAN
APLIKASI *THE POINT OF SALE***



N a m a : Yoga Kosasih
NIM : 16523081

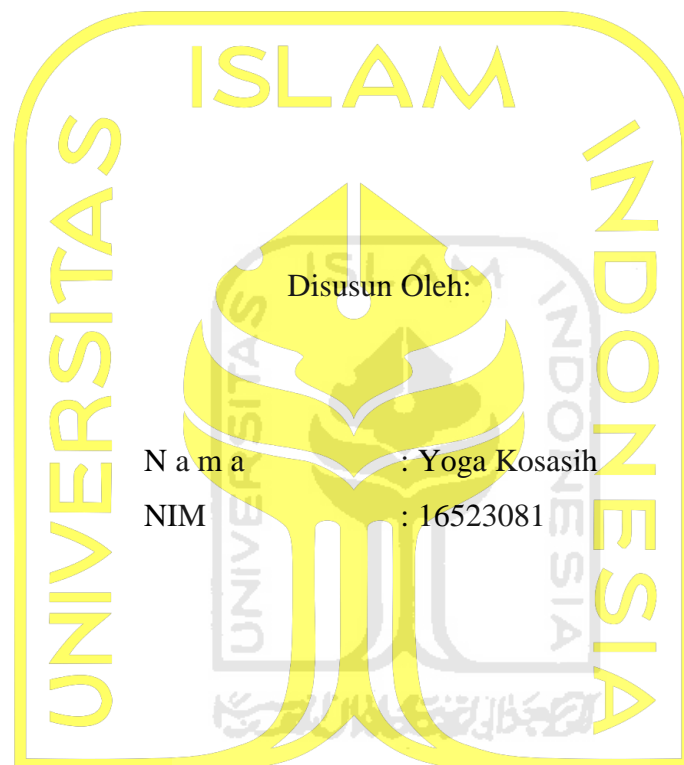
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2020

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**AUTOMATION TESTING TOOL DALAM PENGUJIAN
APLIKASI *THE POINT OF SALE***

TUGAS AKHIR JALUR MAGANG



Disusun Oleh:

N a m a : Yoga Kosasih

NIM : 16523081

المعهد الإسلامي
Yogyakarta, 14 Januari 2021
الجامعة الإسلامية
باندونج

Pembimbing,

Andhik Budi Cahyono S.T., M.T.

HALAMAN PENGESAHAN DOSEN PENGUJI

**AUTOMATION TESTING TOOL DALAM PENGUJIAN
APLIKASI *THE POINT OF SALE***

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 14 Januari 2021

Tim Penguji

Andhik Budi Cahyono S.T., M.T.



Anggota 1

Hendrik, S.T., M.Eng.



Anggota 2

Elyza Gustri Wahyuni, S.T., M.Cs.

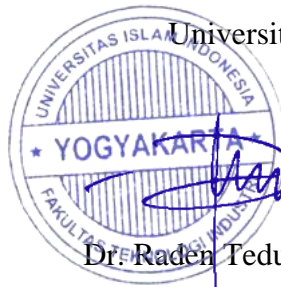


Mengetahui,

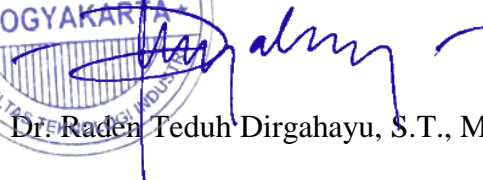
Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



Dr. Raden Teduh Dirgahayu, S.T., M.Sc.



HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Yoga Kosasih

NIM : 16523081

Tugas akhir dengan judul:

**AUTOMATION TESTING TOOL DALAM PENGUJIAN
APLIKASI *THE POINT OF SALE***

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 14 Januari 2021



HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahim...

Segala puji bagi Allah subhanahu wa Ta'ala rabb semesta alam, berkat rahmat dan kasih sayang-Nya sehingga saya dapat menyelesaikan Tugas Akhir ini.

Melalui kesempatan ini, saya ingin mengucapkan terima kasih yang sebesar-besarnya kepada pihak-pihak yang terkait, dan sebagai wujud rasa terimakasih saya ingin mempersembahkan tugas akhir ini kepada Kedua orang tua saya, Ibunda saya Minarsih dan Ayah saya Sahudi yang selalu memberikan dukungan penuh kepada saya, bukan hanya dukungan dalam materi tetapi juga memberikan nasehat dan do'a yang selalu menyertai langkah demi langkah saya, Keluarga besar saya, keluarga besar Ahmadi sulis dan keluarga besar Mataram, Sahabat sahabat dan teman-teman dekat saya dan masih banyak lagi yang tidak bisa saya sebutkan satu persatu, terima kasih banyak.



HALAMAN MOTO

“Berpikir itu gratis” – osas

“Paniklah dengan tenang!” – osas

“Kau pikir dengan menangis kau akan menemukan jalan keluar?” – Luffy



KATA PENGANTAR

Assalamu'alaikum WarahmatullahiWabarakatuh

Alhamdulillah, saya panjatkan kehadirat Allah SWT yang telah memberikan rahmat, hidayat serta karunia-Nya, sehingga laporan Tugas Akhir dapat saya selesaikan. Tugas Akhir ini adalah sebagai salah satu syarat yang harus dipenuhi untuk memperoleh gelar sarjana di Jurusan Informatika FTI UII. Adapun Tugas Akhir ini mengenai “AUTOMATION TESTING TOOL DALAM PENGUJIAN APLIKASI *THE POINT OF SALE* (STUDI KASUS TPOS PT. JAVASIGNA INERMEDIA)”,

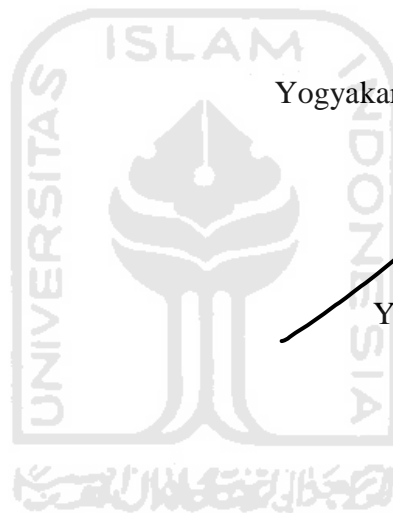
Pelaksanaan tugas akhir ini merupakan salah satu mata kuliah wajib dari jurusan informatika FTI UII dan juga merupakan sarana bagi saya untuk menambah wawasan serta pengalaman dan menerapkan di kehidupan selama berada di bangku perkuliahan. Oleh karena itu, kesempatan kali ini saya menyampaikan rasa terima kasih kepada:

1. Bapak Hendrik, S.T., M.Eng. selaku Ketua Jurusan Informatika Universitas Islam Indonesia dan sebagai dosen penguji penulis sidang pendadaran yang banyak memberikan saran dalam penyusunan laporan ini.
2. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc selaku Ketua Program Studi Informatika Program Sarjana.
3. Bapak Andhik Budi Cahyono, S.T, M.T. selaku dosen pembimbing yang banyak membantu dan meluangkan waktu untuk membimbing, memeriksa, dan memberikan saran dalam penyusunan laporan ini.
4. Bapak Sahudi dan Ibu Minarsih selaku kedua oran tua dan adik saya Rangga Dwi Saputra yang selama ini selalu mendoakan dan memberi dukungan moral materi.
5. Bapak Andwi Valentine sebagai *Chief Executive Officer* di PT. Javasigna Intermedia yang telah mengizinkan penulis melakukan kegiatan magang di perusahaan beliau.
6. Mas Rangga sebagai pembimbing lapangan, yang telah bersedia untuk meluangkan waktu untuk membimbing, memeriksa, serta memberikan petunjuk-petunjuk selama pelaksanaan magang.
7. Semua dosen Program Studi Informatika UII yang telah memberikan ilmu bermanfaat dan arahan positif selama menjalani studi ini. Semua rekan Program Studi Informatika UII angkatan 2016, yang saling membantu dan menyemangati satu sama lain.

8. Dua Sahabat penulis Raden Mifthakhurozak Budi Nugraha dan Rio Pradana Aji.
Terima Kasih
9. Seluruh keluarga, rekan, dan semua pihak yang tidak bisa saya sebutkan satu persatu. Terima kasih telah banyak sekali membantu penulis dalam pelaksanaan tugas akhir. Semoga sehat selalu cita- cita dan mimpi yang diinginkan semuanya tercapai.

Menyadari bahwa di laporan ini masih belum sempurna, karena keterbatasan kemampuan dan kurangnya pengalaman di lapangan. Oleh karena itu sangat mengharapkan kritik dan saran yang membangun demi sempurnanya Laporan Tugas Akhir ini. Akhir kata sangat berharap agar laporan ini dapat bermanfaat bagi semua pihak.

Wassalamu'alaikum WarahmatullahiWabarakatuh



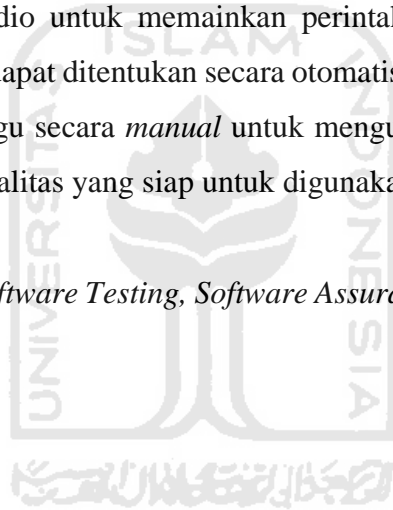
Yogyakarta, 14 Januari 2021

Yoga Kosasih

SARI

PT. Javasingna Intermedia mengembangkan sebuah aplikasi berbasis *website* untuk memudahkan mengelola penjualan produk di bermacam daerah di Indonesia. Namun terdapat banyak sekali kecacatan yang ditemukan pada aplikasi yang sedang dikembangkan, karena hanya mengandalkan pengujian *manual* saja, padahal ada *tool* yang bisa dimanfaatkan salah satunya yaitu Katalon studio. Untuk itu, pada penulisan laporan kali ini akan memanfaatkan *automation testing tool* yaitu Katalon studio. Tidak seperti pengujian *manual* Katalon studio memiliki fitur *record/merekam* yang membantu penggunaannya dalam pembuatan *test case* dan sangat sensitif terhadap *error*. Tujuan dari penulisan laporan ini untuk memastikan *software* yang dikembangkan bekerja sesuai dengan fungsi yang diharapkan dan untuk menganalisis efektivitas sebuah aplikasi. Penulisan laporan ini menjelaskan mekanisme *automatic waiting* atau perintah tunggu Katalon studio untuk memainkan perintah tes. Dan hasil percobaan menunjukkan bahwa waktu tunggu dapat ditentukan secara otomatis dan dinamis sehingga tidak perlu menambahkan perintah tunggu secara *manual* untuk mengurangi kesalahan buatan dan menghasilkan *software* yang berkualitas yang siap untuk digunakan.

Kata kunci: Katalon Studio, Software Testing, Software Assurance, Blackbox Testing, Web Testing, Point Of sale



GLOSARIUM

<i>Point of sale</i>	merupakan aktivitas yang berorientasi pada penjualan serta sistem yang membantu proses transaksi penjualan dan stok barang.
<i>Bug</i>	suatu cacat desain pada perangkat keras atau perangkat lunak yang mengakibatkan terjadinya galat pada peralatan atau program sehingga tidak berfungsi sebagaimana mestinya.
<i>Backlog</i>	antrian fleksibel pekerjaan-pekerjaan, yang diminta pihak bisnis untuk dikerjakan pihak eksekutor di masa depan.
<i>E-commerce</i>	penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti internet, televisi, dan jaringan komputer lainnya.
<i>Git</i>	perangkat lunak pengendali versi atau proyek manajemen kode perangkat lunak.
<i>Pull</i>	menarik seluruh data perubahan yang telah dilakukan apabila tertinggal beberapa titik dibawah.
<i>Push</i>	untuk menyimpan perubahan yang telah dilakukan ke dalam sebuah <i>repository</i> yang berada di <i>git server</i> .
<i>Commit</i>	menyimpan perubahan di lokal baru melakukan <i>push</i> ke server.
<i>Compile</i>	proses untuk mengubah berkas kode program dengan berkas lain yang terkait menjadi berkas yang siap untuk dieksekusi oleh sistem operasi secara langsung.
<i>Debug</i>	langkah untuk menelusuri kesalahan kode program.
<i>Merge</i>	penggabungan <i>code</i> antara <i>branch</i> 1 dengan <i>branch</i> yang lainnya
<i>Branch</i>	biasa disebut percabangan tempat melakukan perubahan.

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI.....	ix
GLOSARIUM	x
DAFTAR ISI	xi
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Ruang Lingkup Magang.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Sistematika Penulisan.....	2
BAB II DASAR TEORI.....	4
2.1 Quality Assurance / Software Tester.....	4
2.2 Sistem POS (Point Of Sale).....	7
2.3 Katalon Studio	7
BAB III PELAKSANAAN MAGANG	9
3.1 Manajemen Proyek	9
3.1.1 Pemantauan Proyek	9
3.1.2 Pendefinisian Proyek	11
3.1.3 Perencanaan Proyek.....	15
3.1.4 Pelaksanaan Proyek	17
3.1.5 Penutupan Proyek.....	32
3.2 Pengujian Aplikasi TPOS.....	33
3.2.1 Test Case	34
3.2.2 Stop Record, Save & Playback.....	35
3.3 Hasil Pengujian.....	36
BAB IV REFLEKSI PELAKSANAAN MAGANG	43
4.1 Implementasi Menggunakan Katalon Studio	43
4.2 Perbandingan Solusi dengan Alternatif Lain.....	44
4.3 Hasil dari Magang pada Kualitas Diri.....	45
BAB V KESIMPULAN DAN SARAN	46
5.1 Kesimpulan.....	46
5.2 Saran.....	46
DAFTAR PUSTAKA	47

DAFTAR TABEL

Tabel 3.1 Tabel Analisis Kebutuhan Data.....	12
Tabel 3.2 Arsitektur Aplikasi TPOS	13
Tabel 3.3. Kebutuhan Sistem.....	14
Tabel 3.4 Spesifikasi Browser	14
Tabel 3.5 Spesifikasi Desktop	15
Tabel 3.6 Kebutuhan Fungsionalitas Aplikasi TPOS	15
Tabel 3.7 Keseluruhan Aktivitas.....	17
Tabel 3.8 Skenario Pengujian TPOS.....	33
Tabel 3.9 Hasil Skenario Pengujian TPOS.....	37



DAFTAR GAMBAR

Gambar 3.1 Trello	10
Gambar 3.2 GitLab	10
Gambar 3.3 Sourcetree	11
Gambar 3.4 Slack.....	11
Gambar 3.5 Halaman Login TPOS	20
Gambar 3.6 Halaman Registrasi Email	20
Gambar 3.7 Halaman Registrasi Telpon	21
Gambar 3.8 Halaman Daftar Toko.....	21
Gambar 3.9 Dashboard TPOS	22
Gambar 3.10 Halaman pelanggan.....	23
Gambar 3.11 Halaman Tambah Pelanggan	23
Gambar 3.12 Halaman tambah Produk	24
Gambar 3.13 Halaman Daftar Produk.....	25
Gambar 3.14 Halaman Detail Produk.....	25
Gambar 3.15 Halaman Edit produk	26
Gambar 3.16 Halaman Pindah Toko Lain.....	26
Gambar 3.17 Halaman Kelola Kategori.....	27
Gambar 3.18 Halaman tambah pesanan.....	27
Gambar 3.19 Halaman Daftar Pesanan	28
Gambar 3.20 Halaman Invoice Pesanan.....	28
Gambar 3.21 Halaman Bayar Pesanan.....	29
Gambar 3.22 Halaman Atur Pesanan	29
Gambar 3.23 Halaman Info Grafis.....	30
Gambar 3.24 Halaman Pegawai Toko.....	31
Gambar 3.25 Halaman Tambah Pegawai	31
Gambar 3.26 Halaman pengaturan.....	32
Gambar 3.27 Halaman Akses Level	32
Gambar 3.28 Halaman Tambah Produk.....	35
Gambar 3.29 Halaman Tambah Pesanan	35
Gambar 3.30 Proses Perekaman Event Pada Katalon	36
Gambar 3.31 Test Case untuk Menguji TPOS	39
Gambar 3.32 Report Automation Test Case.....	40

Gambar 3.33 Log View 41
Gambar 3.34 Report Manual test Case..... 41
Gambar 4.1 Konfigurasi menggunakan selenium webdriver 44



BAB I

PENDAHULUAN

1.1 Latar Belakang

PT. Javasingna Intermedia merupakan sebuah industri kreatif yang bergerak di bidang teknologi informasi, berdiri pada akhir tahun 2014 di Yogyakarta. Javasingna tidak hanya memiliki produk aplikasi yang sudah jadi atau bisa digunakan secara langsung oleh pengguna, tapi juga bergerak dalam sektor digital agensi. Javasingna menyediakan pelayanan perancangan aplikasi menggunakan teknologi terkini yang dibutuhkan untuk memecahkan masalah yang dihadapi oleh berbagai macam institusi. Mulai dari Aplikasi *website*, *desktop* dan *mobile*, tidak hanya itu Javasingna juga memberikan beberapa pelayanan tambahan seperti *design creative* dan *internet/social media marketing*.

The Point Of Sale atau biasa disebut TPOS dalam karya ilmiah merupakan aplikasi *website* untuk mengelola penjualan produk pada tiap-tiap titik toko di bermacam daerah di Indonesia. Secara teori *The Point Of Sale* (TPOS) merupakan tempat di mana pelanggan melakukan pembayaran untuk barang atau jasa, dan di mana pajak penjualan dapat dibayarkan. TPOS masih dalam tahap pengembangan dan TPOS diharapkan dapat membantu transaksi penjualan dan perhitungan stok barang sehingga para admin toko dapat melakukan penjualan dan perhitungan stok barang melalui TPOS dari gawai mereka (Sukandar, 2019). Manfaat lain yang diharapkan adalah kemudahan admin untuk mengelola toko diberbagai titik daerah di Indonesia melalui satu aplikasi mulai dari informasi penjualan hingga stok produk.

TPOS memiliki 3 aktor dan 2 situs. Aktor TPOS terdiri dari super admin, *user*, dan pegawai. Super admin dapat mengelola toko, pengguna, hak akses, dan admin di situs khusus untuk super admin. *User* dan pegawai memiliki tugas yang sama yaitu mengelola pelanggan, produk, pesanan, pegawai toko dan pengaturan toko di situs khusus untuk pegawai dan *user*.

Tujuan penulisan laporan ini adalah untuk mengukur kualitas perangkat lunak berdasarkan pengujian otomatis, Masalah yang diangkat dalam penyusunan laporan ini dibatasi pada pengukuran kualitas perangkat lunak berbasis web mengacu pada beberapa batasan aspek pengujian yaitu pengukuran perangkat lunak berdasarkan aspek *functional test*, pengukuran perangkat lunak berdasarkan aspek *Fron-tend Performance test*, pengukuran perangkat lunak berdasarkan aspek *Back-end Performance test*, dan pengukuran perangkat lunak berdasarkan aspek *User Interface test*.

12 Ruang Lingkup Magang

Pelaksanaan magang berlangsung kurang lebih selama enam bulan di PT.Javasigna Intermedia mulai dari akhir bulan Oktober 2019 hingga akhir april 2020. Ruang lingkup magang memiliki beberapa batasan dan hal-hal yang dilakukan selama magang batasan tersebut dibuat untuk menghindari terjadi pengerjaan ulang dan membuang-buang waktu. Adapun aktivitas yang dilakukan selama magang adalah sebagai berikut:

1. Proyek TPOS. Terlibat dalam proses pengembangan perangkat lunak ini sejak awal mulai magang hingga akhir masa magang.
2. Pengembangan proyek TPOS menggunakan Katalon Studio.
3. Pengerjaan tugas sesuai dengan yang diberikan oleh *project manager*.
4. Melakukan pengujian untuk menunjukkan cacat aplikasi TPOS.
5. Melakukan pengujian bergantung pada konsteks pengujian.
6. Pengelompokan cacat/*bug* TPOS.
7. Melakukan pengujian secara berulang ulang.

13 Tujuan

Tujuan berfokus pada pengujian otomatis untuk mengembangkan aplikasi *The Point Of Sale* (TPOS) untuk Mencari kecacatan dari sebuah aplikasi agar supaya nanti kalau sudah rilis cacatnya sudah sangat minim, tidak menghilangkan cacat/*bug* 100% mulai dari tahap perencanaan hingga tahap pengiriman produk sehingga menghindari terjadi pengerjaan ulang dan membuang-buang waktu.

14 Manfaat

Pengembangan aplikasi *The Point Of Sale* (TPOS) menggunakan *tool* Katalon Studio yang dikerjakan selama berjalannya magang mempunyai beberapa manfaat. Evaluasi dilakukan terus menerus dan secara berkala agar supaya tidak menemukan kecacatan yang lebih banyak sehingga beberapa manfaat dari implementasi Katalon Studio pada aplikasi TPOS adalah sebagai berikut:

1. Pengujian dapat dilakukan secara berulang yang dapat di-*update* untuk mengurangi pekerjaan ulang yang merugikan.
2. Meningkatkan efisiensi operasional.
3. Menghindari pemborosan dan menghasilkan aplikasi yang berkualitas.

15 Sistematika Penulisan

Berisi susunan bab dan subbab pada keseluruhan laporan

A. BAB 1 PENDAHULUAN

Pada bab ini berisi tentang latar belakang ditulisnya laporan ruang lingkup magang yang merupakan pekerjaan magang yang diterima kemudian dijadikan bahan TA, tujuan dari pekerjaan tersebut, manfaat yang didapat dari pekerjaan tersebut dan sistematika penulisan sebagai struktur inti laporan

B. BAB 2 DASAR TEORI

Pada bab ini berisi tentang sumber teori teori yang digunakan dalam pekerjaan yang dilakukan pada saat magang dan menuliskan laporan TA ini.

C. BAB 3 PELAKSANAAN MAGANG

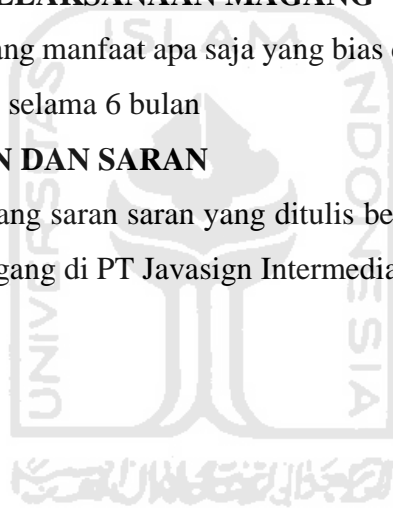
Pada bab ini berisi tentang penjelasan terkait pelaksanaan magang yang telah dilakukan dan pekerjaan apa yang telah dilakukan selama magang di PT Javasingna Intermedia

D. BAB 4 REFLEKSI PELAKSANAAN MAGANG

Pada bab ini berisi tentang manfaat apa saja yang bias diambil dari hasil magang di PT Javasingna Intermedia selama 6 bulan

E. BAB 5 KESIMPULAN DAN SARAN

Pada bab ini berisi tentang saran saran yang ditulis berdasarkan pengalaman yang telah dilalui selama magang di PT Javasingna Intermedia



BAB II DASAR TEORI

21 Quality Assurance / Software Tester

Quality Assurance atau biasa disebut QA adalah seseorang yang bertanggung jawab terhadap perencanaan jaminan kualitas, kesalahan, penyimpanan rekaman, analisis dan pelaporan. Jaminan kualitas perangkat lunak adalah aktivitas pelindung yang diaplikasikan pada seluruh proses perangkat lunak. Proses ini dilaksanakan oleh seorang *QA Tester*. *QA Tester* memiliki tugas utama yaitu melakukan pengujian terhadap perangkat atau emulator, membuat alur pengujian, serta membuat laporan hasil pengujian yang terkadang bertugas untuk membuat program pengujian otomatis, membuat laporan pengujian, memberikan masukan atas aplikasi yang diuji, serta berkomunikasi dengan pihak-pihak yang berkepentingan, seperti pengembang UI/UX, *Back-End* atau Project Manager (PM) (Zakiyah, 2018) .

Penulis dalam pengembangan aplikasi TPOS menggunakan *7 principles of testing* atau 7 hal prinsip dasar tentang testing yang tujuannya untuk mengoptimalkan pada saat melakukan testing apa saja 7 prinsip itu? (Mirza, 2020) Berikut adalah 7 hal prinsip dasar tentang testing :

1. ***Testing shows presence of defects*** atau pengujian menunjukkan cacat dalam melakukan pengujian perangkat lunak proses yang dilakukan sebenarnya adalah untuk mencari cacat pada sebuah perangkat lunak yang dikembangkan, yang dimana tester harus membuat sebuah perangkat lunak itu menjadi gagal. Itu diperlukan supaya proses pengujian perangkat lunak bisa menemukan kegagalan program sebanyak banyaknya. Meskipun dalam beberapa proses pengujian sangat mustahil untuk memastikan bahwa sebuah perangkat lunak 100% bebas dari cacat/*bug*, bahkan jika tidak ada cacat yang ditemukan itu bukan bukti kebenaran
2. ***Early testing*** atau pengujian dilakukan lebih awal pengujian yang dimulai sedini mungkin dalam pengembangan perangkat lunak, sehingga cacat bisa ditemukan oleh tester diawal testing yang nantinya tester bisa merencanakan testing yang akan dibuat itu seperti apa dan ketika menemukan sebuah cacat/*bug* nantinya tester bisa memberi saran kepada developer
3. ***Exhaustive testing is not possible*** atau mustahil melakukan pengujian secara menyeluruh melakukan pengujian perangkat lunak secara keseluruhan itu sangat tidak mungkin, tidak perlu melakukan testing secara menyeluruh melainkan bisa

juga tester set ekspektasi dan berasumsi membuat skenario batas atas dan batas bawah sebuah perangkat lunak dan tidak perlu mengetes semuanya melainkan menggunakan asumsi bahwa tes sudah dilakukan maka itu akan mencakup keseluruhan

4. **Testing is context dependent** atau pengujian bergantung pada konsteks pengujian bergantung pada konteks yang pada dasarnya berarti bahwa cara menguji sebuah situs akan berbeda dari cara menguji aplikasi komersial setiap *test case* yang dibuat tergantung konteksnya seperti contoh semisal *test case* A tentang *login* sedangkan *test case* B tentang *logout* itu hal yang berbeda dan ekspektasinya juga berbeda baru bisa melakukan testing. Sehingga pada saat proses pengujian pada perangkat lunak harus benar-benar sesuai dengan konteks dari perangkat lunak itu sendiri karena setiap perangkat lunak itu memiliki cara pengujian yang berbeda masing-masing tidak bisa dipukul sama rata.
5. **Defect clustering** atau pengelompokan cacat/*bug* pengelompokan cacat sebuah perangkat lunak pada proses pengujian lebih diutamakan melakukan pengujian pada modul atau fungsional program yang kecil, biasanya semakin kecil modulnya maka cacat/*bug* yang akan ditemukan akan semakin banyak
6. **Pesticide paradox** atau paradox pestisida proses pengujian yang dilakukan secara berulang pada akhirnya tidak akan menemukan cacat pada perangkat lunak. tester tidak bisa begitu saja bergantung pada teknik pengujian yang ada maka harus terus menerus memperbaiki metode yang ada untuk membuat pengujian lebih efektif seperti melakukan kesalahan membiarkan *test case* tidak di-*update* dan menganggap remeh bahwa *test case* tersebut selalu hijau dan tidak perlu melakukan *update test case* itu hal yang salah dan tidak boleh dilakukan
7. **Absence of error fallacy** atau tidak ada *bug* merupakan kesalahan dalam pengujian tidak ditemukan adanya cacat/*bug* maka hal itu perlu dicurigai, bisa saja perangkat lunak yang 99% bebas dari cacat masih tidak dapat digunakan, hal ini bisa terjadi jika sistem diuji secara menyeluruh untuk persyaratan yang salah maka pengujian perangkat lunak tidak hanya menemukan cacat tetapi juga untuk memeriksa bahwa perangkat lunak memenuhi kebutuhan bisnis, sering menemukan bahwa tesnya berhasil tapi tidak sesuai dengan fungsi jadi ketika *error* itu tidak ada bukan berarti aplikasi itu dinyatakan bebas lolos dari QA.

Penulis melakukan proses pengujian menggunakan 2 metode yaitu pengujian otomatis dan pengujian *manual*, pada saat kapan tester menggunakan otomatis testing dan kapan menggunakan *manual testing*? (Mirza, Spotify, 2020) berikut adalah waktu yang tepat kapan tester harus menggunakan otomatis testing dan kapan tester menggunakan *manual testing*:

1. Automation Testing:

- Tes kritikal dari sebuah produk, tester harus membuat skenario testing untuk *core functionality*, *core functionality* biasanya banyak berfungsi pada fungsi fungsi lainnya maka dibutuhkan software testing secara *repetitive*.
- Tes yang harus di eksekusi pada setiap *build* atau rilis, jika aplikasi yang dimiliki memiliki banyak versi dan harus diuji setiap versinya.
- Tes yang harus dijalankan pada banyak kombinasi *platform* dan os version, seperti 1 skenario di eksekusi untuk *platform* yang berbeda beda seperti Android, IOS, WEB, API dan *browser* yang berbeda seperti *Chrome* atau *Mozilla*.
- Tes yang dieksekusi pada *work flow* yang sama tapi menggunakan data yang berbeda beda, seperti contoh semisal ada 1 *flow* yang selalu di ulang ulang tapi *inputan* datanya berbeda beda.
- Tes yang membutuhkan *penginputan* data secara masif dan berulang, contohnya seperti pada saat mengisi *form* terkadang banyak ada yang berulang juga.
- Tes yang digunakan pada *performance* testing, biasanya akan dilakukan pada *performance* tes pada sebuah aplikasi kalau itu pakainya *manual* bagaimana caranya kita mengeksekusinya.
- Tes yang dalam eksekusinya membutuhkan waktu yang lama, biasanya dilakukan *smoke test* atau tes yang berjalan terus terusan itu harus menggunakan *automted* tes.

2. Manual Testing:

1. Tes yang hanya dieksekusi 1 kali.
2. *User experience*, tes yang membutuhkan user untuk merasakan secara langsung atau mendapat *experience* secara langsung terhadap sebuah aplikasi atau *real user*.
3. Tes yang harus dilakukan secara *emergency*, kalau ada *bug fix* yang harus segera di tes dan kalau menggunakan *automation script* biasanya akan membutuhkan waktu sedikit lebih lama dibanding *manual* tes.

4. Tes yang dilakukan secara *random / exploratory testing*, ada kalanya sebagai software tester harus bermindset *testing shows presence of defects* jadi harus *explore* terus mencari *corner case* yang belum kita tes
5. Tes yang membutuhkan konfirmasi visual, contoh semisal yang ingin *pixel perfect* button ditengah tengah atau kurang *margin* ada ditengah kanan sedikit kiri sedikit terlalu kepinggir *1 pixel 2 pixel* atau gradien warnanya

Kesimpulan tergantung dari kebutuhan *testing* yang dilakukan jika membutuhkan pengujian yang repetitif atau harus menguji banyak *platform* data yang besar sebaiknya itu di buat *automation script* dan jika sifatnya cuma 1 kali *testing* atau dibutuhkan perasaan atau *experience* langsung biasanya sifatnya *emergency* sebaiknya *manual* juga bisa dipertimbangkan

22 Sistem POS (Point Of Sale)

Sistem POS (Point of Sale) ialah suatu sistem untuk menyinkronkan serta mengintegrasikan data pesanan, data reservasi, data kartu hadiah, data *e-commerce*, ataupun data poin loyalitas yang terletak di fitur POS dengan fitur *website* orang dagang dan menyinkronkan informasi yang terletak di basis informasi *website* ke perangkat POS terkait . Sistem POS digunakan untuk mengintegrasikan sistem merchant *e-commerce* dan sistem pemesanan reservasi, dan lainnya yang disediakan pada situs *website*. Sistem POS mencakup fitur POS, server *website* POS, susunan basis informasi POS, aplikasi situs *website* POS (United States Patent No. US 2012/0296679 A1, 2012).

Adapun karya ilmiah yang berjudul *Web Based Point of Sale System* atau disingkat WPOS terbentuk sistem berbasis *website* yang membolehkan manajemen laporan toko secara jarak jauh, serta membolehkan pelanggan melakukan penjadwalan ataupun penjadwalan ulang waktu pengiriman. Server menyediakan seluruh informasi dan data penting yang dibutuhkan melalui *website browser* pelanggan. Dalam sistem ini, server antar toko bisa berkomunikasi satu sama lain dengan *mainframe* kantor pusat. WPOS bisa diimplementasikan bagaikan rangkaian terintegrasi untuk kerja sama antar posisi toko (United States Patent No. US 2004/0181454 A1, 2004).

23 Katalon Studio

Untuk kebutuhan proses pengujian dilakukan dengan menggunakan *tool* yaitu Katalon Studio. *Tool* tersebut merupakan aplikasi open source untuk melakukan pengujian secara

otomatis yang telah dikembangkan oleh Katalon LLC yang dapat dijalankan di semua sistem operasi seperti Windows, MAC os dan linux. Katalon Studio menggunakan antarmuka IDE khusus untuk melakukan pengujian. Katalon Studio memiliki tiga fitur utama untuk menunjang beberapa pengujian di beberapa *platform* seperti *Web testing*, *API testing*, dan *Mobile testing* (Studio, 2020). Selain itu Katalon studio telah terintegrasi dengan beberapa teknologi dari luar seperti *github*. Katalon Studio sendiri telah menyediakan UI grafis yang beragam tampilan, menu, pohon tabel dan lain-lain untuk mengelola *test case* objek dan file data. Walaupun masih dalam tahap pengembangan Katalon Studio juga sudah mendukung beberapa *environment* seperti *browser* dan sistem operasi komposisinya sangat ideal untuk penguji yang hanya ingin melakukan *drag-drop* dan memiliki kemampuan pemrograman terbatas.

Makalah yang membahas tentang pengembangan sistem web dan contohnya adalah Pengembangan Sistem Informasi Point Of Sales Terintegrasi Dalam Lingkup Rumah Makan Beserta Cabangnya (Studi Kasus: RM. Pecel Pincuk Bu Tinuk) (Sani, Pradana, & Rusdianto, 2018). Makalah tersebut menjelaskan proses pengembangan sistem POS untuk digitalisasi proses konvensional yang sudah ada. Pada sistem yang dibangun telah melalui pengujian fungsional *Whitebox* dan *Blackbox testing* serta pengujian nonfungsional agar sistem dapat dikatakan sudah berjalan dengan baik tanpa harus terjadi kesalahan yang akan muncul di kemudian hari sehingga web telah dinyatakan telah memenuhi semua kebutuhan sistem.

Analisis yang dilakukan pada Katalon Studio digunakan untuk menganalisis *response time* pada beberapa halaman aplikasi TPOS. Katalon Studio bisa menangkap *object/element* (*button, field text, datapicker, link, checkbox, radio button, select, ect.*) pada setiap halaman di web *browser* ataupun aplikasi *mobile* yang sedang *record* / rekam untuk dapat dijalankan pada *testing*. Namun jika terdapat *error* di saat melakukan *play* sesuai *record* yang telah dibuat, maka bisa menambahkan penambahan *script* untuk melakukan pengecekan lebih lanjut. Katalon Studio menyembunyikan semua kompleksitas teknis di belakang layar dan menyediakan UI yang ramah dengan mode *manual* tetapi masih menyimpan sesuatu yang dibutuhkan untuk pengguna yang lebih secara teknis yang dapat menggali lebih dalam koding dengan menggunakan mode *scripting* yang sepenuhnya mendukung kenyamanan pengembangan seperti sintaksis, saran kode dan *debugging*. Walaupun masih dalam tahap pengembangan Katalon Studio juga sudah mendukung beberapa *environment* seperti *browser* dan sistem operasi(OS). Dalam pembentukan *test case*, Katalon Studio memiliki 3 cara yaitu merekam aktivitas dalam web, membuat *manual test case* dan membentuk *script*, yaitu *test case* dibuat secara *manual* oleh penguji (Matin Mujduddin, 2019)

BAB III

PELAKSANAAN MAGANG

3.1 Manajemen Proyek

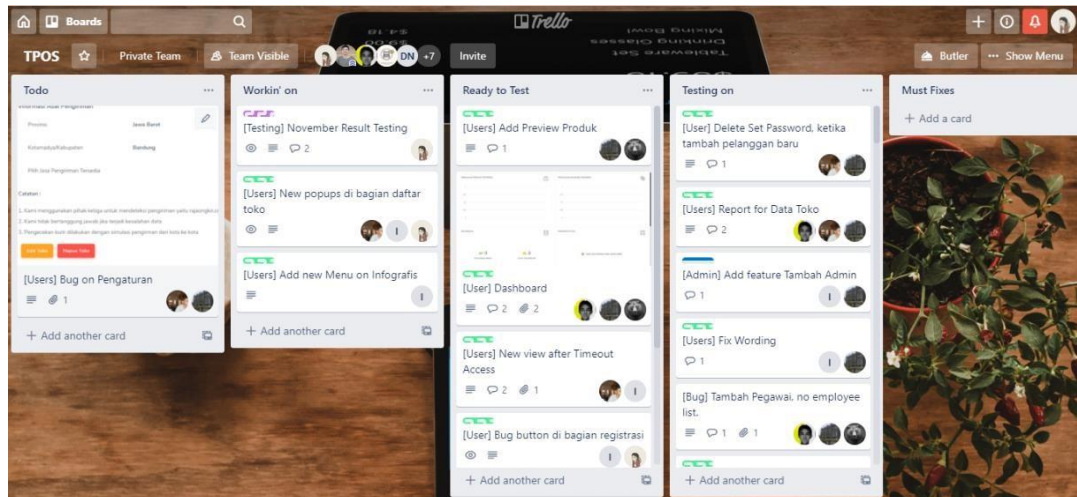
Dalam pengembangan aplikasi *The Point Of Sale* (TPOS) telah dilaksanakan melalui tahapan-tahapan manajemen proyek sebagai berikut:

3.1.1 Pemantauan Proyek

Pengembangan aplikasi *The Point Of Sale* (TPOS) dikembangkan oleh dua orang *developer*, satu orang *quality assurance* (penulis) dan satu *project manager*. Proses pemantauan dan pengontrolan proyek dilakukan melalui aplikasi pendukung pihak ketiga, aplikasi tersebut adalah:

a. Trello

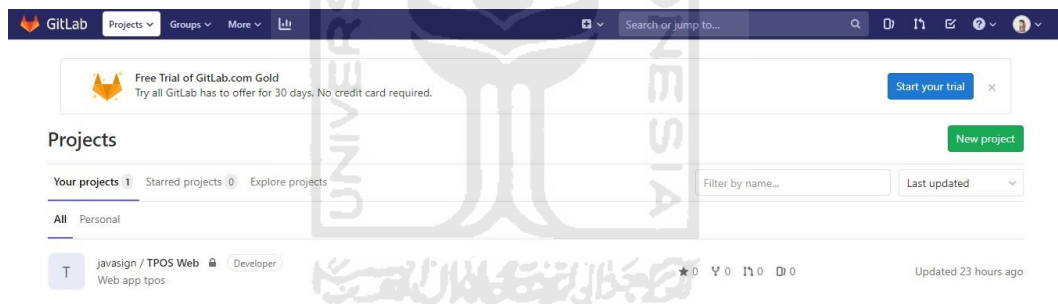
Trello merupakan sebuah aplikasi untuk kolaborasi berbasis web yang digunakan untuk mengelola sebuah proyek. Semua orang yang ada didalam proyek tersebut dapat mengetahui aktivitas apa saja yang sedang dikerjakan oleh masing-masing anggota. **Gambar 3.1** merupakan tampilan dari manajemen proyek TPOS pada aplikasi Trello. Aktivitas yang berada pada bagian *Backlog* merupakan antrian pekerjaan yang diminta oleh *client* berupa hal apa saja yang diharapkan dalam aplikasi TPOS. Antrian pekerjaan di *Backlog* akan diterjemahkan oleh *project manager* menjadi tugas atau aktivitas yang nantinya akan dikerjakan oleh *developer* di antrian *Todo*. Antrian pada *Workin' on* merupakan tugas yang sedang dikerjakan oleh pengembang, jika tugas sudah selesai, maka tugas akan dipindahkan ke antrian *Ready to Test* dan siap untuk diuji oleh *tester*. *Testing on* adalah kumpulan aktivitas yang sudah selesai dikerjakan dan siap untuk di tes oleh *tester*, apakah terdapat *bug* atau *error*. *Ready to Deploy* merupakan antrian aktivitas yang sudah lolos dari *bug* dan *error* tetapi masih perlu pemantauan akhir oleh *project manager*. Jika *project manager* menyatakan suatu aktivitas lolos, maka aktivitas tersebut dinyatakan selesai dan siap untuk dipindahkan ke antrian *Done*.



Gambar 3.1 Trello

b. GitLab

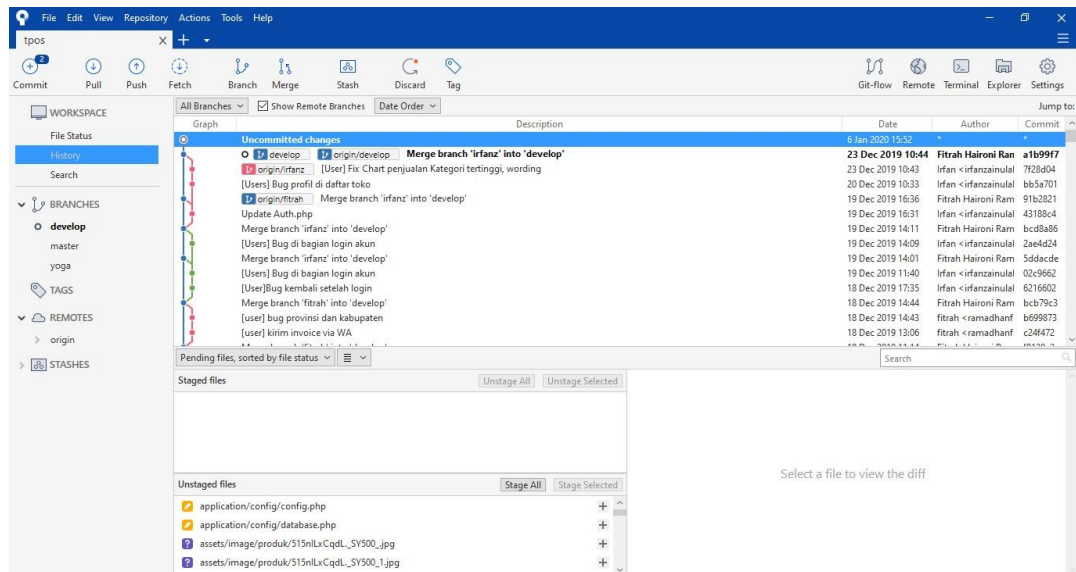
GitLab adalah layanan yang menyediakan *remote access* ke *Git repository*. Proyek TPOS menggunakan layanan GitLab untuk memudahkan penggabungan tugas antar pengembang dan meminimalisir *error* dengan cara pengecekan atau pengambilan kembali versi pengembangan sistem yang sesuai. **Gambar 3.2** merupakan tampilan proyek TPOS pada GitLab.



Gambar 3.2 GitLab

c. Sourcetree

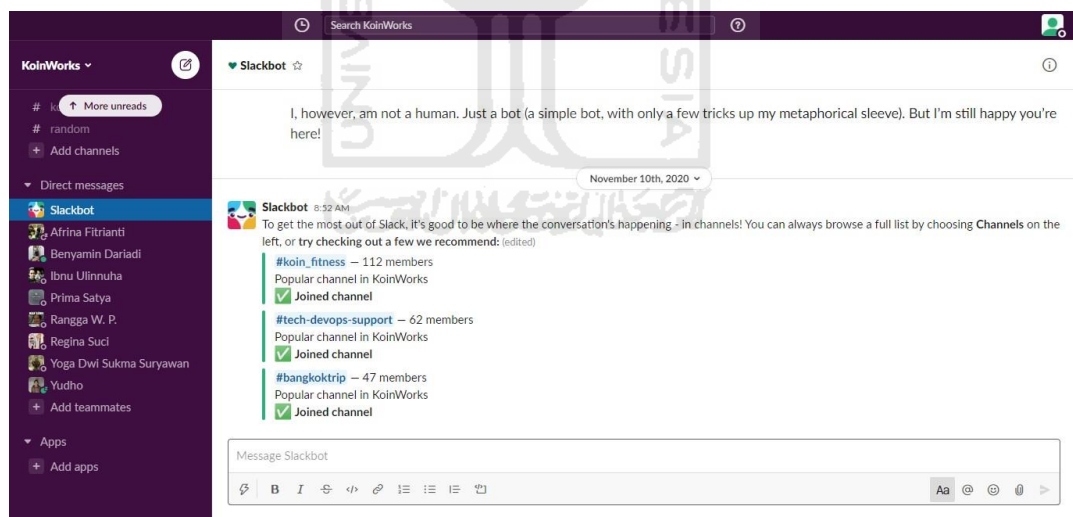
Sourcetree adalah aplikasi yang memudahkan pengembang berinteraksi dengan *Git repository*. Sourcetree memvisualisasikan pengelolaan *Git repository* melalui GUI Git Sourcetree yang sederhana. Pada pengembangan proyek TPOS, penulis menggunakan Sourcetree untuk keperluan *push*, *pull*, dan *commit* kode program. Selain itu Sourcetree juga membantu melihat perubahan yang telah dilakukan pada *file* termasuk penambahan dan pengurangan *file* gambar. **Gambar 3.3** menunjukkan tampilan Sourcetree.



Gambar 3.3 Sourcetree

d. Slack

Slack merupakan aplikasi yang digunakan untuk berkomunikasi antar individu atau tim proyek. Slack juga dapat digunakan untuk berbagi berkas, hal ini lebih mudah dilakukan karena berkas yang berisi kode program dapat diulas secara langsung pada *desktop* tanpa harus mengunduhnya terlebih dahulu. **Gambar 3.4** menunjukkan tampilan aplikasi Slack.



Gambar 3.4 Slack

3.1.2 Pendefinisian Proyek

The Point Of Sale (TPOS) merupakan sebuah aplikasi yang dikembangkan oleh PT. Javasniga Intermedia untuk mengelola penjualan produk pada tiap titik toko di berbagai wilayah di Indonesia melalui situs web. Tujuan dikembangkan sistem tersebut untuk mempermudah *client* mengelola barang, transaksi penjualan, dan pembuatan laporan

penjualan. TPOS diharapkan dapat membantu transaksi penjualan dan perhitungan stok barang. Para admin toko dapat melakukan penjualan dan perhitungan stok barang melalui TPOS dari gadget masing-masing.

Aplikasi *The Point Of Sale* (TPOS) dikembangkan dengan menggunakan *framework* Codeinteger. Bahasa pemrograman *back-end* yang digunakan adalah PHP dan MySQL sebagai basis data. Aplikasi web TPOS ini mempunyai 2 *Dashboard* yaitu *Dashboard* Admin dan *User*. Pada *Dashboard* Admin terdapat 4 tabel yaitu Tabel Kelola Akses untuk mengatur menu-menu yang dapat diaktifkan dan nonaktifkan, Tabel Kelola Pengguna berfungsi mengubah informasi pengguna dan status pengguna, Tabel Kelola Toko yang berguna untuk mengubah dan menghapus toko dan Tabel Kelola Admin di mana untuk melihat dan menghapus admin. Pada *Dashboard user* terdapat 4 tabel yaitu Tabel Pelanggan, Tabel Daftar Produk, tabel Daftar Pesanan.

Terdapat 7 tabel yang tidak semua memiliki fitur yang sama, seperti tabel pada Daftar Pesanan yang memiliki fitur untuk Cetak Laporan dan filter tabel berdasarkan kondisi tertentu. Untuk lebih lengkapnya apa saja yang dibutuhkan ada ada dalam **Tabel 3.1**

Tabel 3.1 *Tabel Analisis Kebutuhan Data*

Dashboard	Nama Tabel	Data yang ditampilkan
Admin	Tabel Kelola Toko	Nama Toko, Deskripsi Toko, Provinsi, Kabupaten, Nama Pengguna, Aksi
	Tabel Kelola Pengguna	Nama Pengguna, Email Pengguna, No Telepon, Status, Aksi
	Tabel Kelola Akses	Title, Url, Icon, Status, Action
	Tabel Kelola Admin	Nama Admin, Email, Aksi
User	Tabel Pelanggan	Gambar, Kode, Nama, Kontak, Alamat, Kota, Provinsi, Status, Aksi

	Tabel Daftar Produk	Foto Produk, Kode, Nama, Kategori, Stok, Harga, Status, Aksi
	Tabel Daftar Pesanan	No, Invoice, Tanggal Pesanan, Pelanggan, Pesan Via, Bayar, Status Pesanan, Total Pesanan, Aksi
	Tabel Pegawai Toko	Gambar, Email, Nama, Toko, Hak Akses, Aksi

The Point Of Sale (TPOS) dikembangkan sebagai suatu sistem aplikasi berbasis web menggunakan spesifikasi sistem seperti yang ditunjukkan oleh **Tabel 3.2**

Tabel 3.2 *Arsitektur Aplikasi TPOS*

Komponen	Spesifikasi
Platform	PHP V.7.4
Framework	CodeIgniter
Javascript Library	DataTables
Database	MariaDB/MySQL
Browser	Firefox, Chrome, Microsoft Edge, Safari
Sistem Operasi	Microsoft Windows 10

Kebutuhan minimal sistem untuk menjalankan Katalon Studio diperlihatkan pada **Tabel**

3.3

Tabel 3.3 *Kebutuhan Sistem*

	Kebutuhan
Sistem Operasi	Windows 7, Windows 8, Windows 10, macOS 10.11+, Linux (Ubuntu based)

CPU	1 GHz or faster 32-bit (x86) or 64-bit (x64) processor
Memori	Minimum: 1 GB RAM (32-bit) or 4 GB RAM (64-bit) Recommended: 4 GB RAM (32-bit) or 8 GB RAM (64-bit)
Hard Drive (hard disk)	minimal 1 GB space tersedia. Diperlukan ruang tambahan tergantung pada <i>source code</i> proyek dan laporan eksekusi yang dihasilkan.

Spesifikasi *browser* yang mendukung Katalon Studio diperlihatkan pada **Tabel 3.4** Lingkungan / *device* yang digunakan pada penulisan laporan yang dilakukan, hanya pada *desktop*.

Tabel 3.4 Spesifikasi Browser

Desktop Browsers	Version on Windows	Version on macOS
Internet Explorer	9, 10, 11	
Microsoft Edge	Current	
Firefox	56+	Untuk Firefox 57, gunakan Katalon Studio v5.1+
Google Chrome	58+	
Opera	Not supported	
Safari	5.1+	9, 10, 11
Opera	Not supported	

Pengujian dilakukan untuk menguji *response time* pada aplikasi TPOS dengan menggunakan *desktop* yang memiliki spesifikasi seperti pada **Tabel 3.5**

Tabel 3.5 Spesifikasi Dekstop

Komponen	Spesifikasi
Sistem Operasi	Windows 10
CPU	2.50GHz 2.71GHz 64-bit (x64) processor
Processor	Intel® Core™ i5-7200 processor
Memori	8 GB RAM
Hard Drive	1 TB
Koneksi Internet	10 Mbps

3.1.3 Perencanaan Proyek

Pengembangan sistem *The Point Of Sale* (TPOS) memiliki beberapa tahapan. Tahapan-tahapan tersebut adalah tahap menentukan kebutuhan sistem, tahap analisis dan desain, tahap implementasi, tahap uji coba dan tahap akhir. Lingkup pengerjaan proyek berupa pengerjaan tugas-tugas yang akan dibagi oleh *project manager*. Waktu pengembangan proyek dilakukan secara berkelanjutan (fleksibel), lebih tepatnya proyek mulai dikerjakan tanggal 17 Juli 2019 dan perkiraan selesai pada akhir bulan April 2020. Setelah tidak ada lagi daftar tugas yang harus dikerjakan oleh pengembang pada antrian tugas, maka CEO akan memperbarui web yang aktif dengan versi sistem web terbaru. Beberapa contoh tugas dalam bentuk kebutuhan fungsionalitas sistem TPOS dapat dilihat pada **Tabel 3.6**

Tabel 3.6 Kebutuhan Fungsionalitas Aplikasi TPOS

Kode	Deskripsi
TF-1	Aplikasi harus dapat melakukan registrasi via nomor telepon dan <i>email</i> .
TF-2	Aplikasi harus dapat <i>login</i> menggunakan nomor telepon dan <i>email</i> .
TF-3	Aplikasi harus dapat menambah, mengedit, menampilkan, dan menghapus toko.
TF-4	Aplikasi harus dapat <i>login</i> berdasar hak akses yang dimiliki pengguna.
TF-5	Aplikasi harus dapat menampilkan jumlah pelanggan baru dalam seminggu, total pelanggan, total pegawai, grafik penjualan produk tertinggi, grafik penjualan kategori tertinggi, dan pengingat stok di halaman <i>dashboard</i> di situs <i>user</i> dan pegawai.

Kode	Deskripsi
TF-6	Aplikasi harus dapat menambah, mengedit, menampilkan, dan menonaktifkan data pelanggan di situs <i>user</i> dan pegawai.
TF-7	Aplikasi harus dapat menambah, mengedit, menampilkan, dan menghapus data produk di situs <i>user</i> dan pegawai.
TF-8	Aplikasi harus dapat membagikan produk via <i>whatsapp</i> di situs <i>user</i> dan pegawai.
TF-9	Sistem harus dapat menyimpan produk sebagai <i>draft</i> di situs <i>user</i> dan pegawai.
TF-10	Aplikasi harus dapat mencari produk berdasar kode atau nama produk di situs <i>user</i> dan pegawai.
TF-11	Aplikasi harus dapat memindahkan produk ke toko lain di situs <i>user</i> dan pegawai.
TF-12	Aplikasi harus dapat menambah dan mengedit kategori dan sub kategori produk di situs <i>user</i> dan pegawai.
TF-13	Aplikasi harus dapat melakukan proses pemesanan produk di situs <i>user</i> dan pegawai.
TF-14	Aplikasi harus dapat menyimpan data pesanan di situs <i>user</i> dan pegawai.
TF-15	Aplikasi harus dapat menampilkan, mencetak, memfilter berdasarkan tanggal, dan mencari data pesanan di situs <i>user</i> dan pegawai.
TF-16	Aplikasi harus dapat mengubah status pesanan dan mengirimkan informasi status ke pelanggan melalui <i>email</i> di situs <i>user</i> dan pegawai.
TF-17	Aplikasi harus dapat mencetak dokumen <i>invoice</i> dan mengirimkan dokumen <i>invoice</i> melalui <i>whatsapp</i> dan <i>email</i> di situs <i>user</i> dan pegawai.
TF-18	Aplikasi harus dapat mengatur pesanan berupa metode pengiriman dan metode pembayaran produk di situs <i>user</i> dan pegawai.
TF-19	Aplikasi harus dapat mengembalikan jumlah stok produk jika pesanan dikembalikan di situs <i>user</i> dan pegawai.
TF-20	Aplikasi harus dapat menampilkan info grafis penjualan, pelanggan, dan lokasi di situs <i>user</i> dan pegawai.
TF-21	Aplikasi harus dapat menambah, mengedit, menampilkan, dan menghapus data pegawai toko di situs <i>user</i> dan pegawai.
TF-22	Aplikasi harus dapat memfilter pegawai toko berdasar hak akses di situs <i>user</i> dan pegawai.
TF-23	Aplikasi harus dapat mengedit profil pengguna di situs <i>user</i> dan pegawai.

Kode	Deskripsi
TF-24	Aplikasi harus dapat mengubah menu akses <i>user</i> dan pegawai di situs <i>user</i> dan pegawai.
TF-25	Aplikasi harus dapat menampilkan jumlah toko dan jumlah pengguna di halaman <i>dashboard</i> di situs super admin.
TF-26	Aplikasi harus dapat menambah, menampilkan, mengedit, dan menghapus toko di situs super admin.
TF-27	Aplikasi harus dapat memfilter, mengedit, menampilkan, dan mengubah status <i>user</i> dan pegawai di situs super admin.
TF-28	Aplikasi harus dapat mengedit dan menghapus menu situs <i>user</i> dan pegawai di situs super admin.
TF-29	Aplikasi harus dapat menambah dan menghapus super admin di situs super admin.
TF-30	Aplikasi harus dapat mengedit profil super admin di situs super admin.

3.1.4 Pelaksanaan Proyek

Dalam pelaksanaan proyek, beberapa aktivitas dari keseluruhan aktivitas yang telah dilaksanakan oleh *quality assurance* (penulis) atau sebagai tester aplikasi *The Point Of Sale* (TPOS) adalah pada **Tabel 3.7**.

Tabel 3.7 Keseluruhan Aktivitas

Kode	Deskripsi
TC-1	[Sprint 3] Share produk to WA
TC -2	[Sprint 3] Remove Pegawai Toko column
TC -3	[Admin] New feature popups mobile web view di bagian kelola toko + kelola akses
TC -4	[Admin] Make a Mobile Web View for Admin
TC-5	[Bug] Add new feature ketika menggunakan web view menu side bar disable, jika menggunakan mobile view menu side bar enable
TC -6	[Sprint 3] Add Edit Kelola Pengguna
TC -7	[Sprint 3] Remove unique visitor column
TC -8	[Sprint 3] Save as Draft on Product
TC -9	[Sprint 3] Add Nomor Telepon column Kelola Pengguna
TC-10	[Sprint 3] Add new Infografis Customer Location
TC-11	[Users] Make a Mobile Web View for Users
TC-12	[User] Bug aksi di bagian daftar produk

Kode	Deskripsi
TC-13	[Users] Bug profil di daftar toko
TC-14	[User] Bug button di bagian registrasi
TC-15	[Invoice] Change date format in Invoice pdf
TC-16	[Invoice] Add Attachment file when send Email notification
TC-17	[Users] Bug di bagian login akun
TC-18	[Users] Add new Menu on Infografis
TC-19	[Users] Bug nomer telpon di bagian edit pelanggan
TC-20	[Users] Bug nomer telpon di bagian pegawai toko
TC-21	[Users] Add Payment Confirmation to Users
TC-22	[Register] Add +62 when Register using numberphone
TC-23	[Users] Bug provinsi di bagian buat toko
TC-24	[Register] Make Register via Numberphone
TC-25	[Users] Filter Data
TC-26	[Users] Add new Menu on Pesanan Dibatalkan
TC-27	[Users] Bug gambar pelanggan di bagian pelanggan
TC-28	[Users] Add filter to Pegawai
TC-29	[Users] Update Invoice Header
TC-30	[Users] Bug icon nomor telpon,email, alamat di edit pegawai
TC-31	[Users] Bug di foto produk bagian daftar produk
TC-32	[Users] Bug di tambah sub kategori + edit
TC-33	[Users] New popups hapus toko di bagian pengaturan
TC-34	[Users] Bug pindah ke toko lain di bagian daftar produk
TC-35	[Users] Bug di bagian daftar pesanan ketika melakukan pembayaran
TC-36	[Users] Add new akses level aksi di bagian pengaturan
TC-37	[Users] Bug Kelola Kategori Produk
TC-38	[Users] Bug nama, nomer telpon, provinsi di bagian pelanggan
TC-39	[Users] Update on Subkategori
TC-40	[User] Add new feature hapus foto profil di bagian edit profile pemilik
TC-41	[Users] Hide some settings on Pengaturan
TC-42	[Users] Add +62 to input No. Telepon on input Pelanggan
TC-43	[Users] Add Filter to Pelanggan
TC-44	[Users] Add new page Detail on Pegawai

Kode	Deskripsi
TC-45	[Users] Add new field on Pegawai
TC-46	[Users] Tambahkan Halaman Detail Pelanggan di menu Pelanggan
TC-47	[Users] Change Color on Menu Cetak Pesanan
TC-48	[Users] Bug on Edit Pelanggan
TC-49	[Users] Add entries filter on Pesanan Page
TC-50	[Admin] Manage Data, Ongkir Manual Can Choose Provinsi & Kabupaten
TC-51	[Users] Add new Metode Pembayaran on Bayar Page
TC-52	[Users] New Menu Infografis
TC-53	[Users] Add new register via email/no telpon
TC-54	[Users] Share Produk to WA
TC-55	[Users] Bug on Pengaturan
TC-56	[Users] New view after Timeout Access
TC-57	[User] Add new notifikasi setelah pesanan di tambah otomatis
TC-58	[User] Add feature popups di bagian daftar produk
TC-59	[Users] New popups di bagian daftar toko
TC-60	[User] Add new feature edit foto profil di bagian edit pegawai + hapus foto profil di bagian edit pegawai
TC-61	[User] Add new feature hapus foto profil di bagian edit pelanggan
TC-62	[Users] Bug di tambah pesanan di bagian stok daftar pesanan
TC-63	[Fix] Remove Columns where export in PDF Excel
TC-64	[Sprint 3] Change Color to Icon
TC-65	[User] must number di bagian registrasi menggunakan nomer telpon
TC-66	[User] new popups di bagian registrasi menggunakan nomer telpon.

Pada **Gambar 3.5** sampai **Gambar 3.27** adalah tampilan yang penulis uji dan penjelasan detail dan fitur-fitur dari aplikasi TPOS mulai dari *Dashboard*, Pelanggan, Produk, Pesanan, Pegawai Toko, Pengaturan, dan Daftar Toko.

Pengguna TPOS dibedakan menjadi 3 aktor yaitu, super admin, *user*, dan pegawai. Setiap aktor memiliki peran yang berbeda seperti yang telah dijabarkan pada BAB 1. Sebelum dapat masuk ke dalam sistem untuk pertama kali, *user* dan pegawai diharuskan melakukan *registrasi* akun melalui *email* atau nomor telepon

Logo: S

Login

Email atau No Telpn
yoga.kosasih172@gmail.com

Kata Kunci

Login

Lupa Kata Kunci

Tidak Punya Akun? Buat Akun

Gambar 3.5 Halaman Login TPOS

Register

Daftar dengan Email | [Daftar Dengan Nomor Handphone](#)

Nama Lengkap
|

Email
|

No HP
+62 Optional |

Kata Kunci | Konfirmasi Kata Kunci |

Register

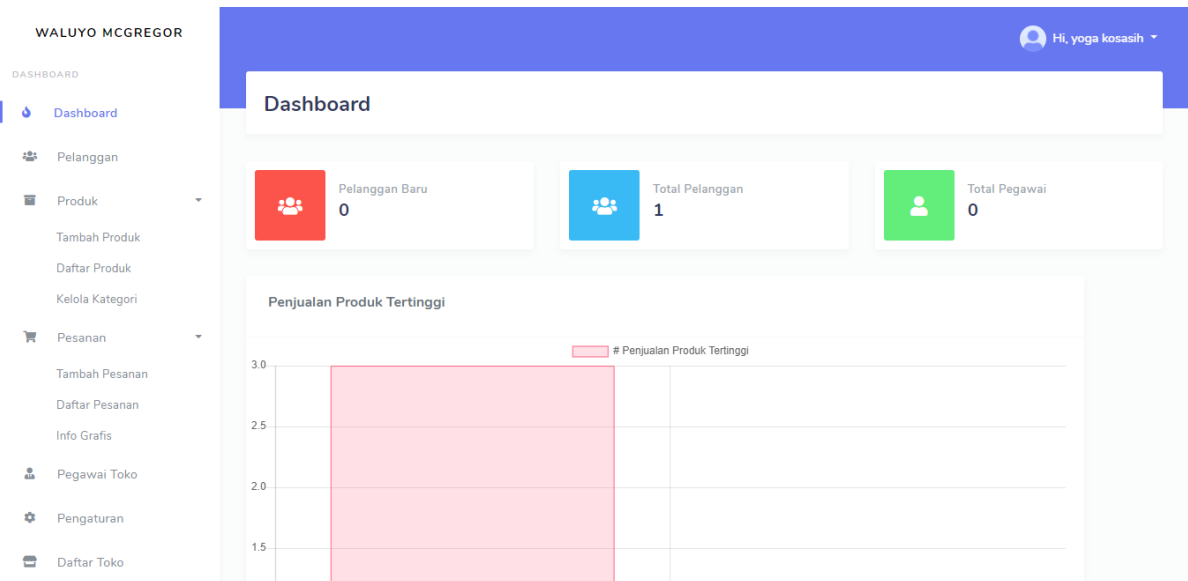
Gambar 3.6 Halaman Reigrasi Email

Gambar 3.7 Halaman Registrasi Telepon

Nama Toko	Ikon	Jumlah Pesanan	Produk Terjual	Omset Penjualan	Aksi
waluyo mcgregor		19	0	Rp. 22.000.000,-	Login

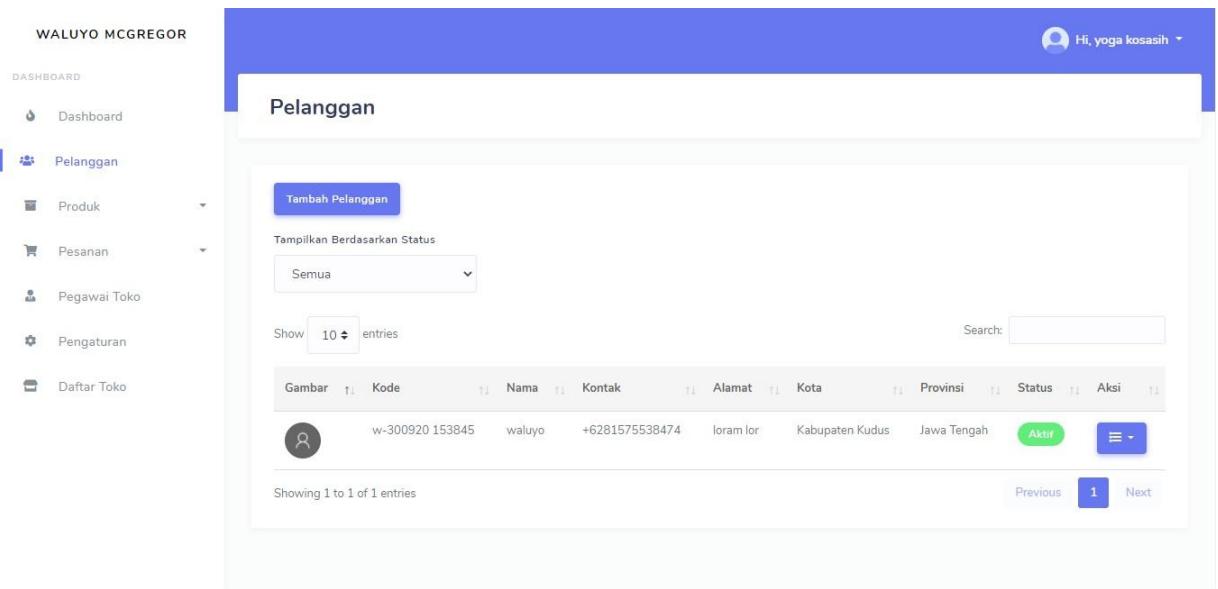
Gambar 3.8 Halaman Daftar Toko

Pada **Gambar 3.9** merupakan *Dashboard* halaman utama TPOS, Halaman utama aplikasi TPOS memiliki berbagai macam komponen seperti dibagian *bar* samping kiri terdapat *Dashboard*, Pelanggan, Produk, Pesanan, Pegawai Toko, Pengaturan, Daftar Toko, dan disamping kanan atas terdapat Profil Pengguna.

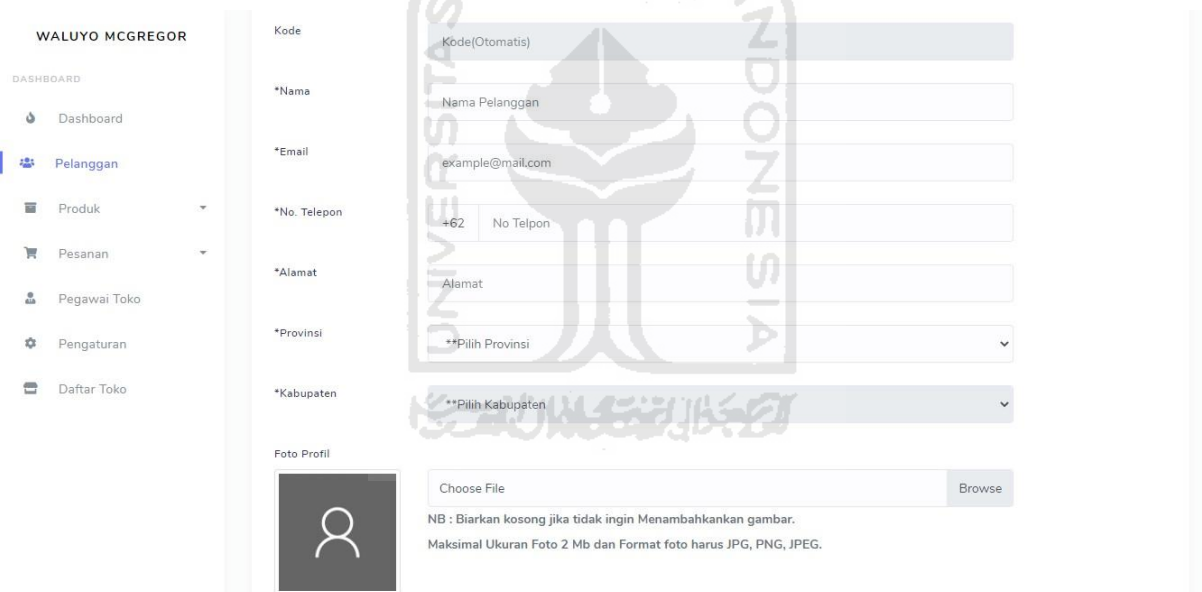


Gambar 3.9 Dashboard TPOS

Gambar 3.10 Merupakan halaman Pelanggan yaitu untuk menambahkan pelanggan dengan mengisi beberapa isian seperti Nama, Email, No Telepon, Alamat, Provinsi, Kabupaten dan foto profil seperti pada **Gambar 3.11 Halaman Tambah Pelanggan**



Gambar 3.10 Halaman Pelanggan



Gambar 3.11 Halaman tambah Pelanggan

Gambar 3.12 Halaman tambah Produk dan ada beberapa sub menu seperti Daftar Produk dan Kelola Kategori. Tambah produk yaitu untuk menambahkan produk dengan mengisi beberapa isian seperti Kode Produk, Nama Produk, Deskripsi Produk, Kategori Produk, bahan, Ukuran, Warna, Link *Instagram*, dan Informasi Penjualan seperti Harga, Produk, Stok produk dan Foto Produk.

WALUYO MCGREGOR

Tambah Produk

DASHBOARD

- Dashboard
- Pelanggan
- Produk
 - Tambah Produk
 - Daftar Produk
 - Kelola Kategori
- Pesanan
- Pegawai Toko
- Pengaturan
- Daftar Toko

Informasi Produk

*Kode Produk: Kode Produk

*Nama Produk: Nama Produk

*Deskripsi Produk: Deskripsi Produk

*Kategori Produk: Pilih Kategori Produk | Pilih Sub Kategori Produk

*Bahan: Bahan

*Ukuran: Ukuran

*Warna: Warna

*Link Instagram: Contoh : <https://instagram.com/p/B8d5INbH9pH/>

Gambar 3.12 Halaman tambah Produk

Gambar 3.13 Halaman Daftar Produk yaitu untuk melihat produk apa saja yang sudah terdaftar dan bisa melihat Foto Produk, Kode Produk, nama Produk, Kategori Produk, Stok Produk, Harga produk, Status Produk, Status Produk dan Aksi dan di aksi terdapat beberapa fitur seperti Detail Produk pada **Gambar 3.14** yaitu untuk melihat detail dari produk itu sendiri, pada **Gambar 3.15** yaitu untuk mengedit informasi produk, Pindah Toko Lain dan pada **Gambar 3.16** yaitu untuk ketika hendak memindahkan produk itu ke toko lain.

WALUYO MCGREGOR Hi, yoga kosasih

DASHBOARD

- Dashboard
- Pelanggan
- Produk**
 - Tambah Produk
 - Daftar Produk
 - Kelola Kategori
- Pesanan
- Pegawai Toko
- Pengaturan
- Daftar Toko

Produk

Re-stok Produk

Tampilkan Berdasarkan Kategori: Semua

Show 10 entries

Foto Produk	Kode	Nama	Kategori	Stok	Harga	
	ip	iphone	hape iphone	7	Rp. 3.000.000,-	Detail Produk Edit Pindah Ke Toko Lain Hapus
	ip	samsul	hape iphone	15	Rp. 4.000.000,-	
	is	ipsul	hape android	8	Rp. 10.000,-	Terbit

localhost/tpos-web/produk/edit_all/122

Gambar 3.13 Halaman Daftar Produk

Detail Produk

Nama Produk: **iphone** Stok: **7**

Harga: **Rp. 3.000.000,-**

Kategori: hape android

Deskripsi: Lainnya

android

Share Produk Ke Whatsapp

Gambar 3.14 Halaman Detail produk

Edit Produk

Informasi Produk

Kode Produk	<input type="text" value="ip"/>
Nama Produk	<input type="text" value="iphone"/>
Deskripsi Produk	<input type="text" value="android"/>
Bahan	<input type="text" value="kayu"/>
Ukuran	<input type="text" value="4.6"/>
Warna	<input type="text" value="rose black"/>
Link Instagram	<input type="text" value="https://www.instagram.com/?hl=id"/>

Gambar 3.15 Halaman Edit Produk

Pindah Ke Toko Lain

Nama Produk

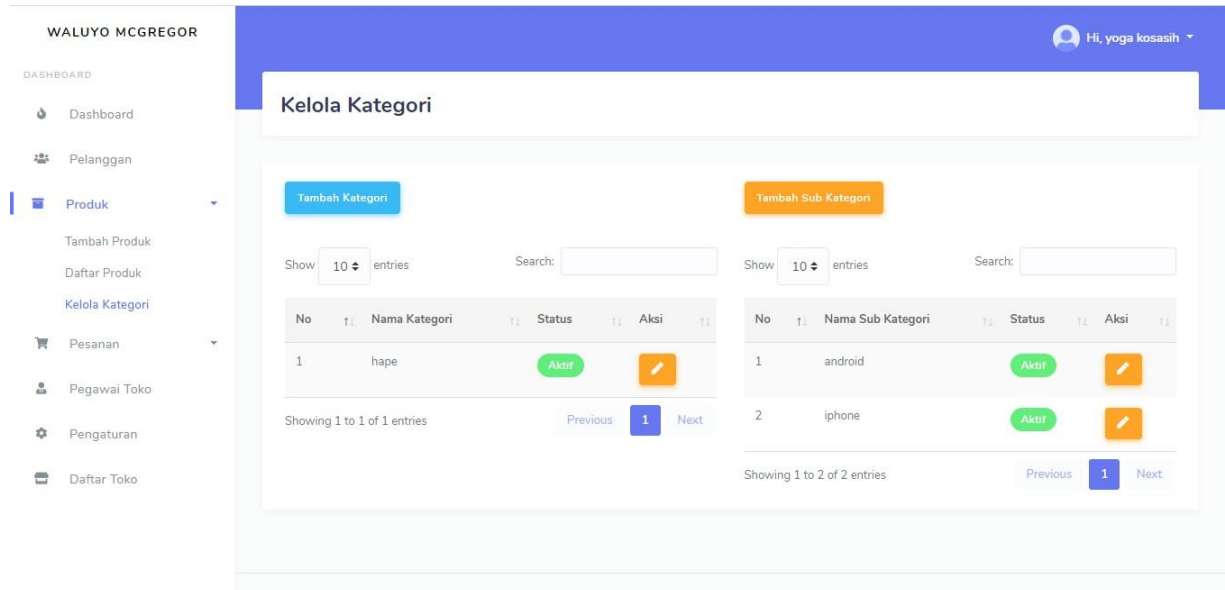
Pindah Ke Toko

Jumlah Yang ingin Dipindah

Harga	Status	Aksi
Rp. 3.000.000,-	Terbit	<input type="button" value="⋮"/>
Rp. 4.000.000,-	Terbit	<input type="button" value="⋮"/>

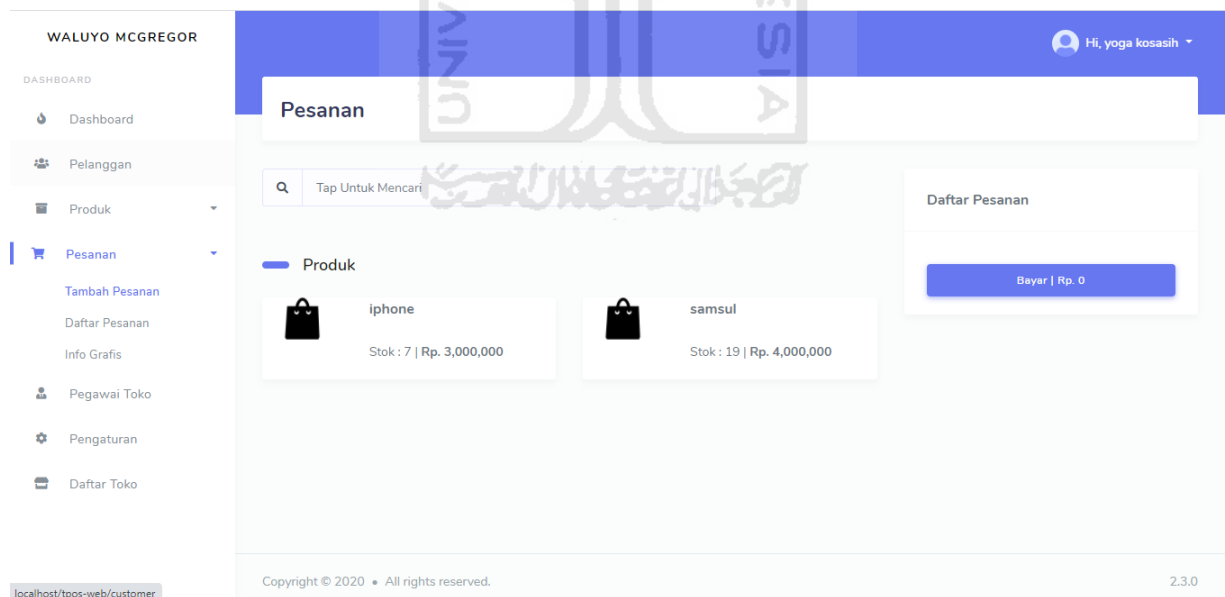
Gambar 3.16 Halaman Pindah Toko Lain

Gambar 3.17 Halaman Kelola kategori yaitu untuk melihat beberapa kategori dan bisa menambahkan kategori dan sub kategori.



Gambar 3.17 Halaman Kelola Kategori

Gambar 3.18 halaman Tambah Pesanan yaitu halaman untuk melihat produk pesanan dan di halaman Tambah Pesanan terdapat Daftar Pesan yang dimana produk yang telah dipilih akan di bayar dengan cara memilih Pelanggan, Pesan via apa terdapat via toko, via Whatsapp, dan Via *Instagram* lalu bisa memilih beberapa pilihan mau Bayar Sekarang, Simpan, atau Batal pesanan.



Gambar 3.18 Halaman Tambah pesanan

Gambar 3.19 Adalah Halaman Daftar Pesanan yaitu Halaman dimana Pesanan yang telah ditambahkan tadi tersimpan dan di Daftar Pesanan terdapat beberapa informasi seperti *Invoice* Pesanan, Tanggal Pesanan, Pelanggan, Via Pesan, Bayar, Status pesanan, Total Pesanan dan Aksi terdapat beberapa menu seperti *Invoice* **Gambar 3.20** yaitu untuk melihat

Invoice bisa untuk cetak Dokumen atau Kirim *Invoice* melalui Whatsapp atau Email, Bayar **Gambar 3.21** yaitu tujuannya untuk melakukan pembayaran dan Atur Pesanan **Gambar 3.22** yaitu untuk melihat metode Pengiriman seperti Kurir, No Resi, Tanggal pengiriman, Alamat kirim dan Metode Pembayaran seperti Via pesanan, Nama Bank, *Note*, dan Pemilik Rekening.

WALUYO MCGREGOR Pesanan

DASHBOARD

- Dashboard
- Pelanggan
- Produk
- Pesanan
 - Tambah Pesanan
 - Daftar Pesanan
 - Info Grafis
- Pegawai Toko
- Pengaturan
- Daftar Toko

Tampilkan Berdasarkan: Semua

Tanggal Awal: Masukkan tanggal awal

Tanggal Akhir: Masukkan tanggal akhir

Cetak Laporan

Show 10 entries

No	Invoice	Tanggal Pesanan	Pelanggan	Pesan Via	Bayar	Status Pesanan
1	wly20201024095217	24 October 2020	waluyo	Instagram	Rp. 0,-	Menunggu Pembayaran
2	wly20201022093547	22 October 2020	waluyo	Toko	Rp. 0,-	Menunggu Pembayaran
3	wly20201022093621	22 October 2020	waluyo	Toko	Rp. 0,-	Menunggu Pembayaran
4	wly20201022093711	22 October 2020	waluyo	Toko	Rp. 0,-	Menunggu Pembayaran

AKSI

- Invoice
- Bayar
- Atur Pesanan

Gambar 3.19 Halaman Daftar Pesanan

Customer :

Nama : waluyo

Email : yoga.kosasih09@gmail.com

No Telp : +6281575538474

Alamat : loram lor

Cetak Dokumen Kirim Invoice

Data Order: wly20201024095217

Status Order : Menunggu Pembayaran

Order Via : Instagram

Tanggal Order: 24-October-2020

Tanggal Pembayaran: Anda Belum Membayar

Order Summary

All items here cannot be deleted.

#	Nama Barang	Harga Per Item	Jumlah Beli	Subtotal
1	samsul	Rp. 4.000.000,-	1	Rp. 4.000.000,-

Gambar 3.20 Halaman Invoice Pesanan

Pesanan

wly20201024095217

Pelanggan : waluyo

Order Via : Instagram

#	Item	Qty	Harga	Subtotal
1	samsul	1	4,000,000	4,000,000
Total				4,000,000

Sudah Dibayar: 0 Jumlah Yang Harus Dibayar: 4000000

Bayar: 4000000 Metode Pembayaran: Transfer

Simpan

Search:

Status Pesanan	Total Pesanan	Aksi
Menunggu Pembayaran	Rp. 4.000.000,-	<input type="button" value="⋮"/>
Menunggu Pembayaran	Rp. 3.000.000,-	<input type="button" value="⋮"/>
Menunggu Pembayaran	Rp.	<input type="button" value="⋮"/>

Gambar 3.21 Halaman Bayar Pesanan

Atur Pesanan

Metode Pengiriman

Kurir:

No Resi:

Tanggal Pengiriman:

Alamat Kirim:

Metode Pembayaran

Via Pesanan:

Nama Bank:

Note:

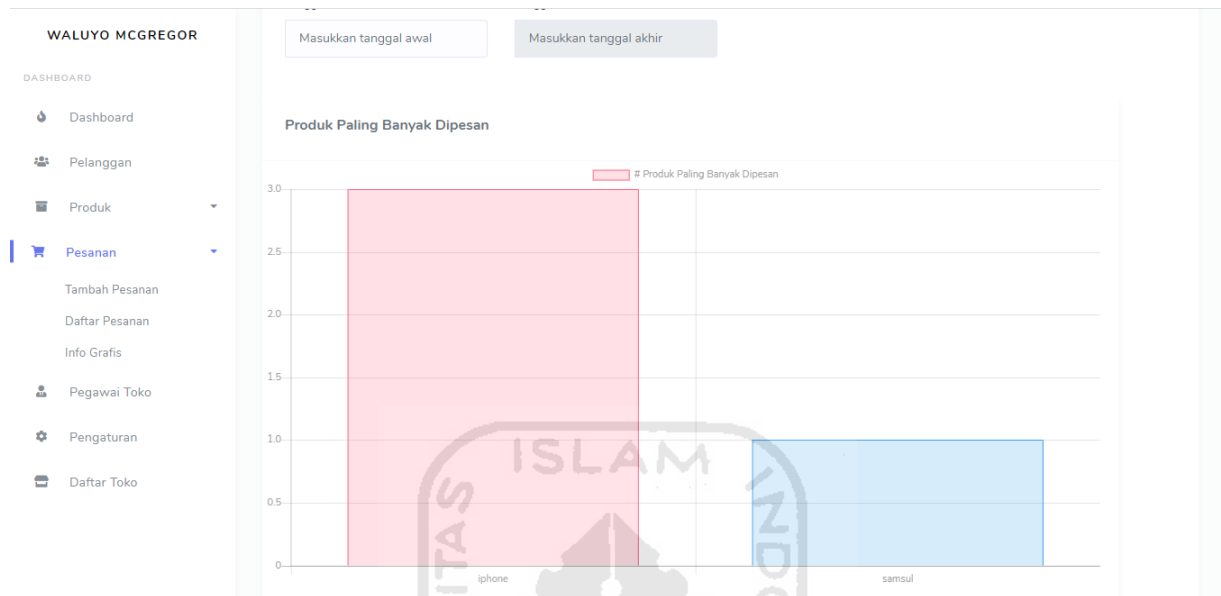
Pemilik Rekening:

Update

Gambar 3.22 Halaman Atur Pesanan

Gambar 3.23 Halaman info Grafis yaitu untuk melihat grafik setiap Penjualan akan terlihat di Halaman Info Grafis, Info grafis menampilkan informasi penjualan, pelanggan, dan lokasi pemesan. Setiap sub menu info grafis menampilkan grafik dan tabel. Info grafis penjualan menampilkan grafik dan tabel produk paling banyak dipesan oleh pelanggan. Info

grafis pelanggan menampilkan grafik dan tabel pelanggan yang paling sering memesan produk. Info grafis lokasi menampilkan grafik dan tabel daerah dengan jumlah pesanan terbanyak. Setiap halaman pada sub menu info grafis memiliki filter tanggal, filter tanggal digunakan untuk menampilkan data terbanyak hingga terkecil berdasar masukan tanggal awal dan akhir.



Gambar 3.23 halaman Info Grafis

Gambar 3.24 Halaman Pegawai toko yaitu untuk melihat dan menambahkan Pegawai Toko Fitur ini dapat menampilkan daftar pegawai berdasarkan hak akses yang dimiliki pegawai. Terdapat empat hak akses yang dimiliki pegawai yaitu, *Customer Service*, Gudang, Keuangan, Manajer. Fitur ini memudahkan pengguna untuk memilih pegawai mana yang ingin ditampilkan berdasarkan hak akses yang sudah disebutkan sebelumnya. **Gambar 3.25** dengan mengisi beberapa isian seperti Nama Pegawai, Email Pegawai, Alamat, No Telpn Pegawai, Kata Kunci Pegawai, Foto Profil Pegawai, dan hak Akses terdapat beberapa pilihan seperti *Customer service*, Gudang, Keuangan, dan Manager.

WALUYO MCGREGOR

Hi, yoga kosasih

DASHBOARD

- Dashboard
- Pelanggan
- Produk
- Pesanan
- Pegawai Toko**
- Pengaturan
- Daftar Toko

Pegawai Toko

[Tambah Pegawai](#)

Tampilkan Berdasarkan

Semua

Show 10 entries Search:

Gambar	Email	Nama	Toko	Hak Akses	Aksi
No data available in table					

Showing 0 to 0 of 0 entries [Previous](#) [Next](#)

Copyright © 2020. All rights reserved.

Gambar 3.24 Halaman Pegawai Toko

Tambah Pegawai

Nama Pegawai

Email Pegawai

Alamat Pegawai

No Telpon Pegawai

Kata Kunci Pegawai

Foto Profil

[Browse](#)

NB : Biarkan kosong jika tidak ingin Menambahkan gambar.
Maksimal Ukuran Foto 2 Mb dan Format foto harus JPG, PNG, JPEG.

Hak Akses waluyo mcgregor

Gambar 3.25 Halaman Tambah Pegawai

Gambar 3.26 Halaman pengaturan yaitu halaman untuk pengaturan umum toko yang telah teregistrasi dan bisa juga untuk ketika pengguna hendak Edit Toko atau Hapus Toko dan dihalaman pengaturan juga terdapat halaman Akses Level **Gambar 3.27** yaitu halaman Akses Level terdapat berbagai akses seperti *Customer Service*, Gudang, Keuangan dan Manager

Gambar 3.26 Halaman Pengaturan

No	Nama Akses	Aksi
1	Customer Service	
2	Gudang	
3	Keuangan	
4	Manager	

Gambar 3.27 Halaman Akses Level

3.1.5 Penutupan Proyek

Setelah tidak ada lagi tugas yang harus dikerjakan oleh pengembang pada antrian tugas, maka CEO akan memperbarui web yang aktif dengan versi sistem web terbaru. Selanjutnya sistem akan digunakan oleh klien. Jika klien menginginkan fitur tambahan, klien akan

berkoordinasi dengan *project manager*. *Project manager* akan menerjemahkan keinginan klien menjadi kebutuhan sistem baru di antrian tugas baru.

Setelah pekerjaan diselesaikan selama 6 bulan, maka proyek pengembangan aplikasi TPOS yang diikuti perlu ditutup. Beberapa kegiatan sebelum penutupan proyek dimulai dari pengujian semua fitur yang telah dibuat. Pengujian dilakukan oleh Quality Assurance penulis dalam pendampingan dua developer. Selanjutnya presentasi hasil pengerjaan kepada bapak Andwi sebagai pengawas proyek aplikasi *The Point Of Sale* (TPOS). Setelah proyek ditutup, diharapkan peserta magang selanjutnya dapat menyempurnakan apa yang belum bisa dikerjakan oleh peserta magang sebelumnya.

32 Pengujian Aplikasi TPOS

Pada pengembangan sistem, khususnya dalam penerapan *tool* Katalon Studio pada aplikasi TPOS, dilakukan beberapa tahapan. Tahapan tersebut adalah sebagai berikut:

1. *Test Case*
2. *Stop Record, Save & Playback*

Pengembangan aplikasi *The Point Of Sale* (TPOS) memiliki beberapa tahapan. Pengujian aplikasi TPOS bisa dieksekusi dengan menggunakan metode *Manual Testing* ataupun dengan metode *Automated software testing* dapat menggunakan salah satu metode atau keduanya untuk kebutuhan test secenario yang kompleks.

Pada penulisan laporan kali ini akan difokuskan kepada *tool* Katalon studio dan dilakukan dua kali pengujian yaitu pengujian *manual* dengan menggunakan metode *Blackbox testing* dan pengujian otomatis dengan menggunakan *tool* Katalon studio untuk melihat *response time* dan *result* dari aplikasi yang diuji, hasil *test case* pengujian terbagi menjadi 2 yaitu *manual* menggunakan Google Spreadsheet dan otomatis menggunakan *test case* yang dibuat dengan metode *record & playback* yang disediakan oleh Katalon Studio. *Record*/rekam merupakan suatu proses perekaman segala *event* yang diterima dari seluruh aktivitas pengguna pada halaman web. Seluruh hasil perekaman nanti akan menjadi sebuah test yang akan disimpan oleh Katalon Studio, dan nantinya akan dijadikan sebagai *test case*. *Test case* tersebut dapat diganti/modifikasi untuk penyempurnaan proses pengujian, *test case* tersebut berikutnya dijalankan secara otomatis menggunakan fungsi *playback*.

3.2.1 Test Case

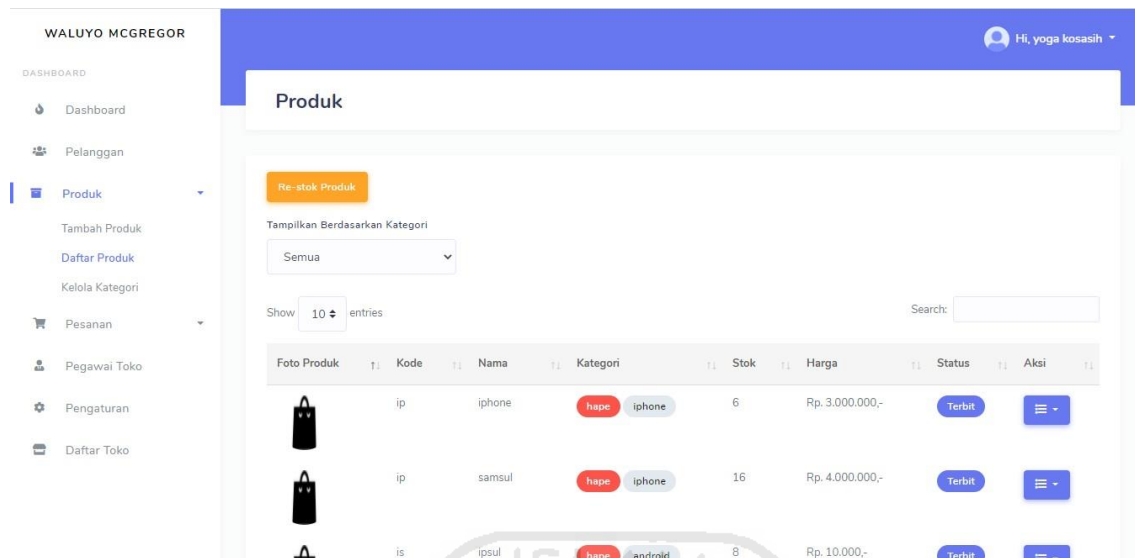
Halaman yang diteliti kali ini adalah halaman untuk menambahkan produk dan pesan produk karena pemesanan produk merupakan fungsi utama dari aplikasi TPOS. berikut adalah beberapa skenario pengujian seperti pada **Tabel 3.8**

Tabel 3.8 Skenario pengujian TPOS

id	Deskripsi Pengujian	Hasil yang Diharapkan
Tc1	Halaman login mengisi Username dengan "yoga.kosasih172@gmail.com" password diisi dengan "bestqa" kemudian klik tombol login atau login menggunakan nomer telpon	Akses login berhasil, sistem akan menutup form login dan menampilkan Dashboard utama TPOS
Tc2	Halaman Tambah produk pengguna mengisi beberapa form seperti kode produk, nama produk, deskripsi produk, kategori produk, bahan, ukuran, warna, link terkait, harga, stock produk, dan foto produk	Tambah produk berhasil dan produk yang telah ditambahkan telah masuk ke daftar Tambah pesanan
Tc3	Halaman Tambah Pesanan pengguna memilih produk lalu membayar produk tersebut	Produk yang dipilih oleh pengguna nantinya akan dibayarkan via toko, Instagram, atau whatsapp
Tc4	Halaman rincian produk pesanan	pengguna akan memilih produk dan lalu melakukan mengatur pemesanan seperti Kurir, No Resi, Tanggal Pengiriman, Alamat Kirim, Metode Pembayaran dan melakukan pembayaran
Tc5	Halaman pembayaran	pesanan yang telah tersimpan tadi akan dibayar Transfer atau Tunai

Gambar 3.27 Adalah tampilan utama dari halaman Tambah produk aplikasi TPOS

Gambar 3.28 Adalah tampilan utama dari halaman Tambah pesanan aplikasi TPOS.



Gambar 3.28 Halaman Tambah Produk

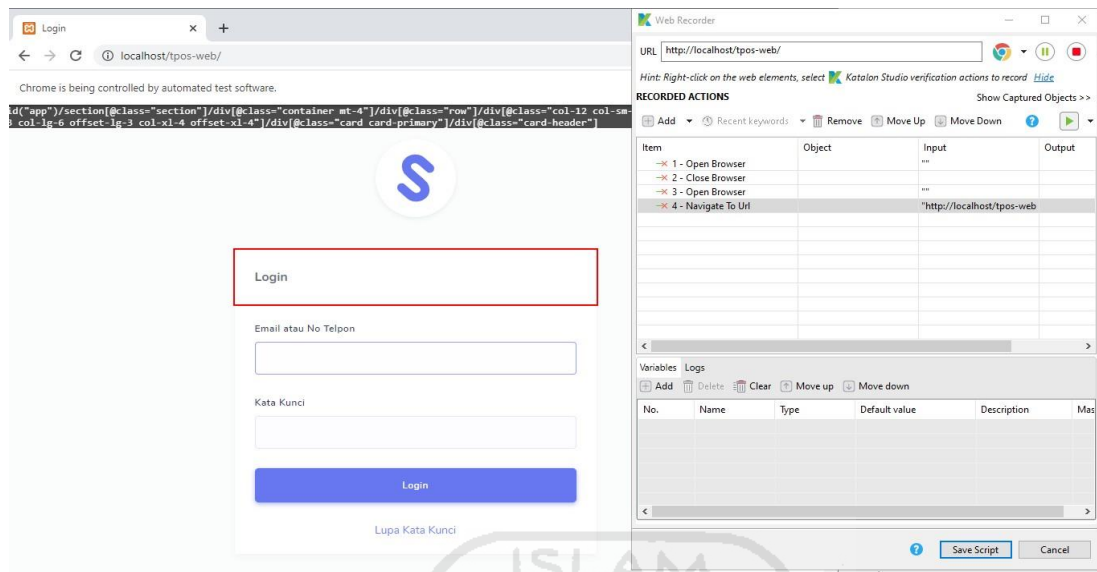


Gambar 3.29 Halaman Tambah Pesanan

3.2.2 Stop Record, Save & Playback

Pada penulisan laporan yang dilakukan, *test case* pengujian dibuat dengan metode *record & playback* yang disediakan oleh Katalon Studio. *Record* adalah proses perekaman semua *event* yang diterima dari aktivitas pengguna pada sebuah halaman web. Semua hasil perekaman akan menjadi test yang akan disimpan oleh Katalon Studio, dan menjadikan sebagai *test case*. *Test case* dapat diubah/modifikasi oleh penguji, untuk penyempurnaan proses pengujian, jika

diperlukan *test case* tersebut selanjutnya dijalankan secara otomatis menggunakan fungsi *playback*. **Gambar 3.29** adalah proses perekaman pada sebuah halaman web



Gambar 3.30 Proseses perekaman event pada katalon

Ketika *test case* dijalankan, log dari setiap langkah secara otomatis akan terbuat. Log tersebut berisi status status berhasil atau gagalnya suatu langkah dan waktu yang dibutuhkan untuk menyelesaikan langkah tersebut. Ketika terdapat langkah yang gagal maka proses menjalankan *test case* akan berhenti dan dapat dilihat rincian dari *test case* tersebut pada *log viewer*.

Kompleksitas halaman *website* bisa diukur dari berbagai macam aspek seperti *browser*, banyaknya *class*, kecepatan data dan lain-lain. TPOS sendiri pasti memiliki kompleksitas yang sangat tinggi. Dalam penulisan laporan kali ini kompleksitas dapat diukur dengan cara mengamati secara langsung objek yang dimuat dalam suatu halaman.

33 Hasil Pengujian

Pengujian dilakukan dalam beberapa tahapan yang pertama yaitu membuat *test case*. *Test case* yang dibuat digunakan untuk menguji *response time* dari halaman utama TPOS, halaman *login*, *Dashboard*, *Pelanggan*, *Produk*, *Pesanan*, *Pegawai Toko*, *Pengaturan*, dan *Daftar Toko*. Hipotesa pada penulisan laporan ini adalah belum tentu TPOS dengan kompleksitas tinggi akan memiliki *response time yang tinggi*. Hal ini karena waktu akses dan eksekusi sebuah proses dipengaruhi juga oleh banyak hal salah satunya adalah algoritma yang digunakan pada tiap *event* di balik komponen tersebut. Sangat mungkin *event* yang lebih kompleks merespon lebih

cepat dibandingkan *event* yang kompleksitasnya rendah, jika didukung dengan algoritma yang *advance* seperti pada **Tabel 3.9**

Tabel 3.9 Hasil Skenario pengujian TPOS

Id	Hasil pengujian	Kesimpulan
T1	Open Browser	Sesuai 15:10:21.640 INFO Starting'Chrome' driver 15:10:21.711 INFO Action delay is set to 0 milliseconds 15:10:26.886 PASSED Browser is opened with url: "
T2	Sistem akan muncul alret berhasil login dan setelah itu masuk ke halaman Dashboard TPOS	Sesuai 15:10:29.146 PASSED Text 'yoga.kosasih172@gmail.com' is set on object 'Object Repository/Page_Login/input_Email atau No Telpon_email'
T3	Produk yang ditambahkan telah berada di halaman Daftar produk	Sesuai 15:10:52.600 PASSED Object: 'Object Repository/Page_Tambah Produk POS/input_Please fill in the first name_submit' is clicked on
T4	Tambah pesanan memilih produk yang akan dibayar melalui toko, Instagram, atau whatsapp	Sesuai 15:10:54.861 PASSED Object: 'Object Repository/Page_Produk POS/a_Tambah Pesanan' is clicked on
T5	Pesanan yang telah tersimpan di Daftar pesanan lalu memilih produk dan mengatur pemesanan seperti Kurir, No Resi, Tanggal Pengiriman, Alamat	Sesuai 15:10:57.941 PASSED Object: 'Object Repository/Page_Order POS/td_w-300920 153845' is clicked on

	Kirim, Metode Pembayaran dan melakukan pembayaran	
T6	Melakukan pembayaran Transfer atau Tunai	Sesuai 15:10:57.941 PASSED Object: 'Object Repository/Page_Order POS/td_w-300920 153845' is clicked on
T7	Close Browser	Sesuai 15:11:01.550 PASSED Browser is closed

Proses pengujian aplikasi TPOS dapat dilakukan dengan cara *manual* ataupun otomatis. Namun pengujian aplikasi dengan cara *manual* memiliki banyak kelemahan diantaranya cakupan pengujian aplikasi yang dilakukan tidak cukup luas, karena penguji cenderung melakukan hal yang sama berulang-ulang sehingga antarmuka yang lainnya tidak diuji. Kelemahan lainnya yaitu sulit mereproduksi kegagalan karena urutan perintah pengujian yang tidak dicatat dan pengujian *manual* tidak melakukan perekaman *response time* secara otomatis.

Pengujian secara otomatis dapat menutupi kelemahan tersebut. Otomatis *testing* pada setiap pengujian mengeksekusi setiap proses pengujian oleh perangkat lunak (*tool*). Terdapat beberapa keuntungan otomatis *testing* pada aplikasi TPOS yaitu memungkinkan pengulangan urutan perintah pengujian, sehingga kegagalan yang terjadi dapat direproduksi, pengujian dapat lebih cepat, serta pengujian otomatis memungkinkan pencatatan *response time* secara otomatis.

Eksperimen dilakukan dalam beberapa tahap yang pertama yaitu pembuatan *test case*. *test case* yang dibuat digunakan untuk menguji *response time* dari tiap halaman utama TPOS mulai dari halaman login, halaman daftar toko, halaman *Dashboard*, Halaman Pelanggan, Halaman Produk, Halaman Pesanan, Halaman Pegawai Toko, dan Halaman Pengaturan. *Test case* untuk pengujian memanfaatkan *test case* yang dihasilkan dari perekaman tiap *event* yang dicoba penguji pada halaman tersebut. *Test case* yang telah dihasilkan lalu dimodifikasi kembali karena terdapat *event* yang tidak bisa terekam oleh Katalon Studio seperti *scroll mouse*.

Tahapan berikutnya adalah *test case* tadi yang telah dimodifikasi selanjutnya di eksekusi untuk mendapatkan *response time* dan hasil dari tiap-tiap halaman yang diuji. Eksekusi *test case* di jalankan secara otomatis oleh Katalon Studio, *test case* yang digunakan untuk menguji mulai dari halaman utama sampai dengan pesanan produk seperti **Gambar 3.30**

Item	Object	Input	Output	Description
→x 1 - Open Browser		""		
→x 2 - Navigate To Url		"http://localhost/tpos-web/"		
→x 3 - Click	input_Email atau No Telpon_email			
→x 4 - Double Click	input_Email atau No Telpon_email			
→x 5 - Set Text	input_Email atau No Telpon_email	"yoga.kosasih172@gmail.com"		
→x 6 - Set Encrypted Text	input_Kata Kunci_password	"o4eleAFY7XY="		
→x 7 - Send Keys	input_Kata Kunci_password	Keys.chord(Keys.ENTER)		
→x 8 - Click	a_Login			
→x 9 - Click	span_Produk			
→x 10 - Click	a_Tambah Produk			
→x 11 - Set Text	input_Kode Produk_kode_produk	"ian"		
→x 12 - Set Text	input_Nama Produk_nama_produk	"ipon androa"		
→x 13 - Set Text	textarea_Deskpripsi Produk_deskpripsi_	"hape tok"		
→x 14 - Select Option By Value	select_Pilih Kategori Produk_khape	"29"; true		
→x 15 - Select Option By Value	select_androidiphone	"66"; true		
→x 16 - Set Text	input_Bahan_bahan	"aluminium foil"		
→x 17 - Set Text	input_Ukuran_ukuran	"6.10"		
→x 18 - Set Text	input_Warna_warna	"rose blek"		
→x 19 - Click	div_Informasi ProdukKode ProdukNa			
→x 20 - Set Text	input_Link Instagram_link	"https://www.instagram.com/?hl=id		
→x 21 - Set Text	input_Rp_harga_beli	"11000000"		
→x 22 - Set Text	input_Stok Produk_stok_produk	"10"		
→x 23 - Click	input_Please fill in the first name out			

Gambar 3.31 Test Case Untuk Menguji TPOS

Pada *test case* tersebut item 1 dan 2 berfungsi untuk menguji *response time* dari halaman utama TPOS, item 3 sampai item 8 berfungsi untuk menguji halaman *login* aplikasi, item 8 sampai 23 berfungsi untuk *input* tambah produk seperti kode produk, nama produk, deskripsi produk, kategori produk, bahan, ukuran, warna, alamat *website/link*, dan informasi penjualan seperti harga, stok produk dan foto produk, item 24 sampai item 42 berfungsi untuk menguji pemesanan produk.

Pada *test case* tersebut item 1 sampai 42 berfungsi untuk menguji *response time* mulai dari *login* sampai pemesanan produk. Total keseluruhan fungsi yang diuji ada 5 fungsi dan 42 item/object yaitu mulai dari pada saat pengguna *login*, tambah produk, tambah pesanan, pesanan yang tersimpan, lalu melakukan pembayaran hingga pengguna berhenti melakukan *testing*, jumlah fungsi yang ditemukan *error* 0 atau tidak ditemukan adanya *error* pada saat pengujian, **Gambar 3.31**, **Gambar 3.32** Menunjukkan *response time* dan *log view* yang diperoleh dari proses pengujian otomatis menggunakan *tool* Katalon Studio dan bisa dikatakan *test case* tersebut berhasil karena *test case* semuanya berwarna hijau, **Gambar 3.33** *Test case manual* diperoleh dari proses pengujian *manual* bisa dibilang bahwa *test case* tersebut juga berhasil

tpos Test Log

Execution Environment

Host name: waluyo - DESKTOP-20807A6
 OS: Windows 10 64bit
 Katalon version: 7.7.1.205
 Browser: Chrome 86.0.4240.111

Test Execution Log

TEST SUITE: tpos Expand All	
Full Name:	tpos
Start / End / Elapsed:	2020-10-24 14:51:34.044 / 2020-10-24 14:52:25.404 / 00:00:51.360
Status:	1 test total, 1 passed, 0 failed, 0 error, 0 incomplete
TEST CASE: Test Cases/tpos Expand All	
Full Name:	tpos/Test Cases/tpos
Start / End / Elapsed:	2020-10-24 14:51:34.506 / 2020-10-24 14:52:25.404 / 00:00:50.898
Status:	PASSED
TEST STEP:	openBrowser("")
TEST STEP:	navigateToUrl("http://localhost/tpos-web/")
TEST STEP:	click(findTestObject("Object Repository/Page_Login/Input_Email atau No Telpn_email"))
TEST STEP:	doubleClick(findTestObject("Object Repository/Page_Login/Input_Email atau No Telpn_email"))
TEST STEP:	setText(findTestObject("Object Repository/Page_Login/Input_Email atau No Telpn_email"), "yoga.kosasih172@gmail.com")
TEST STEP:	setEncryptedText(findTestObject("Object Repository/Page_Login/Input_Kata Kunci_password"), "o4eleAFY7XY=")
TEST STEP:	sendKeys(findTestObject("Object Repository/Page_Login/Input_Kata Kunci_password"), Keys.chord(ENTER))
TEST STEP:	click(findTestObject("Object Repository/Page_Daftar Toko POS/a_Login"))
TEST STEP:	click(findTestObject("Object Repository/Page_Dashboard POS/span_Produk"))
TEST STEP:	click(findTestObject("Object Repository/Page_Dashboard POS/a_Tambah Produk"))

Gambar 3.32 Report Automation Test Case






Gambar 3.31 Memperlihatkan *response time* dalam satuan detik yang diperoleh dari pengujian menggunakan Katalon Studio.



```

Test Cases/tpos (40.148s)
├── 1 - openBrowser("") (6.278s)
│   ├── 2 - navigateToUrl("http://localhost/tpos-web/") (0.990s)
│   ├── 3 - click(findTestObject("Object Repository/Page_Login/input_Email atau No Telpn_email")) (0.867s)
│   ├── 4 - doubleClick(findTestObject("Object Repository/Page_Login/input_Email atau No Telpn_email")) (0.591s)
│   ├── 5 - setText(findTestObject("Object Repository/Page_Login/input_Email atau No Telpn_email"), "yoga.kosasih172@gmail.com") (0.827s)
│   ├── 6 - setEncryptedText(findTestObject("Object Repository/Page_Login/input_Kata Kunci_password"), "o4eleAFY7XY=") (0.504s)
│   ├── 7 - sendKeys(findTestObject("Object Repository/Page_Login/input_Kata Kunci_password"), Keys.chord(ENTER)) (1.288s)
│   ├── 8 - click(findTestObject("Object Repository/Page_Daftar Toko POS/a_Login")) (2.402s)
│   ├── 9 - click(findTestObject("Object Repository/Page_Dashboard POS/span_Produk")) (0.491s)
│   ├── 10 - click(findTestObject("Object Repository/Page_Dashboard POS/a_Tambah Produk")) (1.491s)
│   ├── 11 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Kode Produk_kode_produk"), "ian") (0.880s)
│   ├── 12 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Nama Produk_nama_produk"), "ipon androa") (0.944s)
│   ├── 13 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/textarea_Desripsi Produk_deskripsi_produk"), "hape tok") (0.746s)
│   ├── 14 - selectOptionByValue(findTestObject("Object Repository/Page_Tambah Produk POS/select_Pilih Kategori Produk_hape"), "29", true) (1.529s)
│   ├── 15 - selectOptionByValue(findTestObject("Object Repository/Page_Tambah Produk POS/select_androidiphone"), "66", true) (1.240s)
│   ├── 16 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Bahan_bahan"), "aluminium foil") (0.859s)
│   ├── 17 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Ukuran_ukuran"), "6.10") (0.858s)
│   ├── 18 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Warna_warna"), "rose blek") (0.675s)
│   ├── 19 - click(findTestObject("Object Repository/Page_Tambah Produk POS/div_Informasi ProdukKode ProdukNama ProdukD_5b29c1")) (0.750s)
│   ├── 20 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Link Instagram_link"), "https://www.instagram.com/?hl=id") (0.817s)
│   ├── 21 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Rp_harga_beli"), "11000000") (0.801s)
│   ├── 22 - setText(findTestObject("Object Repository/Page_Tambah Produk POS/input_Stok Produk_stok_produk"), "10") (0.774s)
│   ├── 23 - click(findTestObject("Object Repository/Page_Tambah Produk POS/input_Please fill in the first name_submit")) (1.234s)
│   └── 24 - click(findTestObject("Object Repository/Page_Produk POS/span_Pesanan")) (0.993s)
└── 25 - click(findTestObject("Object Repository/Page_Produk POS/a_Tambah Pesanan")) (1.205s)
    
```

Gambar 3.33 Log View

No Test Case	Sub_Module	Test Scenario	Steps To Reproduce	Prerequisite	Expected Result	Screen Shots	Status	Remarks	Start Date	Target Date
4	Web view Mobile view	feature ketika menggunakan web view menu side bar diaktif, jika menggunakan mobile view menu side bar enable	web view 1. launch the web 2. click login tab Mobile view 1. launch the web 2. click login tab 3. right click mouse 4. inspect	Must login web http://tpos.troli.id/	Data should be appropriate	 	Pass	the data is appropriate, has been updated.	02/03/2020	
5	Admin Mobile View	menambah feature popups di bagian ketika toko + ketika akses di admin	Mobile view as admin 1. launch the web 2. click ketika toko akses (red button) 1. launch the web 2. click ketika akses 3. klik (web button)	Must login web http://tpos.troli.id/	Data should be appropriate		Pass	the data is appropriate, has been updated.	11 February 2020	
6	Pesanan	Remove column Aksi, Invoice	1. launch the web 2. click login toko 3. click pesanan 4. click daftar pesanan 5. click cetak laporan (blue button) 6. click PDF / Excel	Must login web http://tpos.troli.id/	Data should be appropriate	 	Pass	the data is appropriate, has been updated.	26/02/2020	

Gambar 3.34 Report Manual Test Case

Dari segi kompleksitas hampir semua memiliki kompleksitas yang seimbang semua memproses suatu proses rata-rata 5 detik dengan total keseluruhan 40.148s. Dalam pengujian dengan menjalankan *test case response* tiap *test case* berbeda-beda apabila dijalankan beberapa kali. Dalam pengujian dengan menjalankan *test case* yang ada, *response time* pada pada satu *test case* berbeda-beda apabila dijalankan beberapa kali

Katalon Studio memiliki fitur *record*/merekam agar membantu penggunaanya dalam membuat *test case* agar lebih mudah dan Katalon Studio sangat sensitif terhadap *error* pada *test case* terkadang dalam merekam aktivitas untuk *test case* banyak *event* yang coba dilakukan

oleh *user* yang tidak berjalan dalam *test case* atau banyak bagian-bagian yang terekam oleh sendirinya. Maka *test case* tersebut perlu diperbaiki secara *manual* agar *test case* berjalan dengan semestinya.



BAB IV

REFLEKSI PELAKSANAAN MAGANG

41 Hasil Pemanfaatan Katalon Studio

Implementasi pengujian menggunakan Katalon Studio merupakan hal yang paling sering dikerjakan penulis selama kegiatan magang. Implementasi ini memanfaatkan *Automation testing tool* akan sangat membantu tester untuk mengerjakan beberapa pengerjaan untuk tes aplikasi atau *web* dengan lebih mudah, ada yang perlu membuat *scripted test* untuk membuat *flow* pengerjaan *testing* atau bisa juga menggunakan “*record and playback*” lebih mudah dilakukan karena sudah disediakan prosedur yang jelas. Katalon membagi *folder* kerja sesuai fungsinya masing-masing, sehingga akan sangat membantu proses pengembangan sistem. intinya tinggal rekam *web* dan bisa langsung disimpan secara otomatis dan bisa di putar ulang untuk mengetahui apakah hasil sudah sesuai dengan apa yang sudah dibuat sebelumnya pada *record*. Jadi *automation testing* ini bisa membantu tester dalam pengerjaan suatu tes yang berulang dikarenakan jika tester melakukan tes yang berulang mungkin ada beberapa *bug* yang terlewat dikarenakan si tester sudah terbiasa dan hafal mengenai alur dari aplikasi atau *web* tersebut.

Implementasi penggunaan *automation testing tool* kurang ditekankan di matakuliah kurikulum 2016. Pembahasan mengenai pengujian web dilakukan menggunakan *Blackbox testing* atau pengujian *manual* saja. Sehingga akan lebih sulit untuk melakukan manajemen sistem jika sistem yang dikerjakan cukup kompleks. Selama proses magang ditemui bahwa masalah-masalah tersebut dapat diminimalisir menggunakan Katalon Studio alat yang menyediakan *automation testing*. pengujian sistem secara otomatis menggunakan Katalon Studio akan lebih mudah dilakukan, karena Katalon telah menyediakan semua fitur yang dibutuhkan oleh penguji.

Selama pengembangan aplikasi TPOS menggunakan Katalon Studio, penulis juga mengalami beberapa kendala dalam penerapannya. Beberapa diantaranya adalah kendala di bagian RAM karena katalon memakan banyak sekali RAM yang dibutuhkan dan sering sekali mengalami kegagalan pengujian stop di tengah *running testing* pada saat menggunakan RAM 4gb.

42 Perbandingan Solusi dengan Alternatif Lain

Metode lain pengembangan aplikasi *The Point Of Sale* (TPOS) dapat dilakukan tanpa menggunakan *automation testing tool* Katalon Studio yaitu murni menggunakan Bahasa pemrograman seperti Selenium webdriver. Perbandingan pengembangan aplikasi menggunakan Katalon Studio dengan Selenium akan diperlihatkan sebagai berikut:

Selenium webdriver adalah adalah aplikasi pengujian secara otomatis yang *open source* dengan kelebihan dapat digunakan dalam berbagai Bahasa pemrograman seperti C#, Java, php, python, dsb/ Hasil uji dari pengujian otomatis dapat ditampilkan menggunakan *Hypertext Markup Language(HTML)* yang ditampilkan pada *browser* dan dapat dilihat menggunakan format PDF. Umumnya setiap orang memiliki gayanya tersendiri saat menulis kode program menggunakan selenium, hal ini akan membuat orang lain kesulitan memahai kode program yang sudah dibuat. Namun pengujian menggunakan selenium webdriver memiliki beberapa manfaat di antaranya adalah:

1. Cocok digunakan untuk pengembangan web bersekala kecil
2. Pemrograman yang dibangun atas dasar pemikiran pengembang itu sendiri
3. Ukuran *file* relative lebih kecil dibanding dengan Katalon Studio

Selenium webdriver memang mudah digunakan untuk mengembangkan aplikasi web yang bersekala kecil tetapi penggunaanya tidak diannjurkan untuk pengembangan sistem dengan kompleksitas yang cukup tinggi. Terlebih pengembangan secara modular akan sulit dilakukan jika menggunakan selenium webdriver, karena terkadang tidak dipetakan dengan jelas.

Contoh kasus penulisan kode program adalah saat hendak melakukan pengujian web menggunakan selenium webdriver **Gambar 4.1**

```
from selenium import webdriver
from selenium.webdriver.common.keys import keys

browser = webdriver.Chrome()

browser.get('http://localhost/tpos-web')
assert 'TPOS' in browser.tittle

elem = browser.find_element_by_name('p')
elem.send_keys('seleniumhq' + keys.RETURN)

browser.quit()
```

Gambar 4.1 Konfigurasi menggunakan selenium webdriver

43 Hasil dari Magang pada Kualitas Diri

Setelah menjalani proses magang selama kurang lebih 6 bulan di PT. JAVASIGN INTERMEDIA dan mengerjakan berbagai pekerjaan yang diberikan dapat diambil refleksi pada kualitas pribadi yang dijabarkan dalam poin-poin berikut:

1. **Wawasan mengenai dunia testing.** Selama magang yang dilakukan kurang lebih 6 bulan lebih di PT. JAVASIGN INTERMEDIA , banyak hal yang dapat dipelajari dari *Quality Assurance* dalam pengujian *website* dan banyak hal lainnya terkait dunia IT.
2. **Komunikasi dengan satu tim.** selama magang dilakukan komunikasi dengan pihak-pihak pendukung dan instansi terkait, termasuk menjaga hubungan dengan rekan kerja satu tim dan ilmu baru yang didapatkan melalui mereka.
3. **Manajemen waktu.** Selama magang mendapatkan beban pekerjaan dan juga tidak ketinggalan kewajiban sebagai mahasiswa yang wajib diselesaikan. Pada kondisi itu manajemen waktu sangat berharga dan harus dimanfaatkan sebaik-baiknya. Supaya waktu tidak terbuang sia-sia dengan menelantarkan pekerjaan dan kewajiban yang ada
4. **Meningkatkan kemampuan diri.** Setelah melalui berbagai macam pekerjaan yang ada di tempat magang, tidak lengkap rasanya bila tidak ada perubahan pada diri dan kemampuan yang dimiliki. Tentunya selama magang, banyak hal yang telah dipelajari untuk meningkatkan kualitas diri agar menjadi pribadi yang lebih baik dari sebelum dimulainya magang.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Telah berhasil dilaksanakan implementasi Katalon Studio pada proyek aplikasi *The Point Of Sale* (TPOS). Implementasi Katalon Studio Terdiri dari beberapa tahapan, yaitu *test case*, *record & playback and result*. Pada tahap *test case* itu adalah *event* apa saja yang terekam oleh Katalon Studio, pada tahap *record & playback* itu adalah *event* yang telah terekam dan tersimpan dalam bentuk *test case* tadi di putar ulang dan yang terakhir adalah *result* itu hasil dari proses pengujian menggunakan Katalon Studio yang filenya telah tersimpan otomatis di *local disc* masing masing yang dapat di unduh.

Pengujian otomatis menggunakan Katalon Studio cukup efektif dari segi waktu dibandingkan pengujian *manual*, kelebihan dari penggunaan Katalon Studio untuk melakukan pengujian otomatis adalah *interface* yang mudah digunakan dan hasil pengujian yang dapat di *generate* secara otomatis

Berdasarkan eksperimen yang dilakukan dapat diambil kesimpulan bahwa kesimpulan tergantung dari kebutuhan pengujian yang dilakukan jika membutuhkan pengujian yang repetitif atau harus menguji banyak *platform* data yang besar sebaiknya itu di buat *automation script* dan jika sifatnya cuma 1 kali *testing* atau dibutuhkan perasaan atau *eksperience* langsung biasanya sifatnya *emergency* sebaiknya *manual* juga bisa dipertimbangkan.

5.2 Saran

Aplikasi *The Point Of Sale* (TPOS) yang sedang dikerjakan oleh tiga orang yang terdiri dari dua *developer* dan satu *quality assurance* masih sangat bisa dikembangkan dan untuk analisis lebih lanjut dapat dilakukan pengujian otomatis dan pengujian *manual* secara bersamaan dengan menggunakan *desktop* yang lebih proper karena pada saat menjalankan banyak *test case* maka akan terasa sangat lambat ketika spesifikasi *desktop* menggunakan 4gb ram Karena Katalon Studio terlalu memakan ram yang cukup besar maka permasalahan itu bisa terselesaikan seperti menambah ram minimal 8gb atau menggunakan SSD. Dan Katalon Studio masih dalam tahap pengembangan dan itu adalah hal yang sangat wajar sangat disarankan untuk lebih bersabar .

DAFTAR PUSTAKA

- Manno, M. (2004, September 16). *United States Patent No. US 2004/0181454 A1*.
- Matin Mujduddin, M. D. (2019). *Analisis GUI Testing pada Aplikasi E-Commerce*. Retrieved from Analisis GUI Testing pada Aplikasi E-Commerce: <https://jurnal.polban.ac.id/ojs-3.1.2/proceeding/article/view/1443/1205>
- Mirza, A. (2020, 01 15). *Spotify*. Retrieved from Spotify.
- Mirza, A. (2020, 01 29). *Spotify*. Retrieved from Spotify.
- Sani, A. S., Pradana, F., & Rusdianto, D. S. (2018). Pembangunan Sistem Informasi Point Of Sales Terintegrasi Dalam Lingkup Rumah Makan Beserta Cabangnya (Studi Kasus: RM. Pecel Pincuk Bu Tinuk). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2, 3249-3257.
- Studio, K. (2020, 09 28). *katalon studio*. Retrieved from katalon studio: <https://www.katalon.com/>
- Sukandar, C. A. (2019, April 23). *Warta Ekonomi*. Retrieved Desember 18, 2019, from <https://www.wartaekonomi.co.id/read224883/apa-itu-point-of-sale.html>
- Sung Im, Oakton, & VA (US). (2012, November 22). *United States Patent No. US 2012/0296679 A1*.
- Zakiah, F. (2018, 09 27). *Mengenal Software Quality Assurance*. Retrieved from medium: <https://medium.com/qa-malang/mengenal-software-quality-assurance-c3d86a8936be>