

## **BAB III**

### **PELAKSANAAN KERJA MAGANG**

#### **3.1 Kedudukan dan Koordinasi**

Dalam menjalankan kegiatan program kerja magang di PT Quantum Infra Solusindo dengan posisi sebagai *Application Quality Control Intern*, penulis memperoleh kesempatan berharga untuk terlibat dalam tim yang terdiri dari empat orang dan berada di bawah bimbingan *Project Controller*. Tim ini memiliki tanggung jawab dalam mengelola dan memastikan kualitas dari aplikasi yang dikembangkan oleh perusahaan. Sebagai *Application Quality Control Intern*, posisi ini secara khusus ditetapkan untuk melaksanakan implementasi *automation testing*, suatu aspek yang dibutuhkan oleh perusahaan dalam rangka memastikan kehandalan aplikasi yang dikembangkan. Peran ini memiliki fokus pada pengembangan dan penerapan *automation testing*, yang kemudian akan digunakan dalam situasi implementasi rumah sakit baru dan *patch update*.

Selama pelaksanaan program kerja magang penulis dibimbing oleh Kak Budi Setiawan selaku mentor yang memberikan arahan mengenai alur *testing* yang dilakukan pada aplikasi. Dalam perjalanan magangnya, penulis juga mendapat bimbingan secara langsung oleh Pak Julius Saviordi, yang memegang peran penting sebagai *Project Controller* pada PT Quantum Infra Solusindo. Sebagai pembimbing, Pak Julius memberikan arahan dan panduan yang mendalam terkait proses *Quality Control* menggunakan 2 opsi *tools*, yaitu Katalon Studio dan Selenium IDE. Penggunaan Katalon Studio tidak mendapatkan hasil yang efektif karena alur sistem pada aplikasi terlalu kompleks sehingga dalam melakukan pengujian memakan waktu yang sangat lama dan kurangnya perangkat atau hardware yang mendukung dalam menjalankan pengujian menggunakan *tools* Katalon Studio. Setelah mempertimbangkan lebih lanjut sehingga Selenium IDE menjadi pilihan perusahaan berdasarkan kebutuhan proyek. Ketika menggunakan Selenium

IDE, pengujian dapat dilakukan dengan efektif dalam menjaga dan meningkatkan kualitas aplikasi yang dikembangkan oleh perusahaan.

Penting juga untuk dicatat bahwa koordinasi kerja magang dilakukan secara efektif melalui platform WhatsApp, di mana tim dapat berkomunikasi dan berbagi informasi dengan cepat. Meskipun demikian, terdapat juga pertemuan langsung (*onsite meeting*) untuk memastikan pemahaman yang lebih mendalam terkait arahan dan tinjauan kinerja. Kombinasi antara komunikasi daring dan pertemuan langsung menjadi landasan untuk kolaborasi yang efisien dan efektif di dalam tim.

### 3.2 Tugas dan Uraian Kerja Magang

Kegiatan pelaksanaan kerja magang pada PT Quantum Infra Solusindo sebagai *Application Quality Control Intern* sejak tanggal 10 Juli 2023 sampai dengan 8 Desember 2023. Berikut ini merupakan deskripsi tugas yang dikerjakan selama praktik kerja :

1. Melakukan alur testing pada aplikasi demo menggunakan *tools* Selenium pada modul Rawat Jalan dan EMR.
2. Melakukan testing terhadap issue *bugs/error*.
3. Mendokumentasikan skrip Selenium IDE.
4. Membuat dan memperbaharui *user guide*.

Pada Tabel 3.1 dibawah ini merupakan tabel uraian tugas yang dilakukan selama melakukan program kerja magang pada PT Quantum Infra Solusindo:

Tabel 3. 1 Kegiatan Kerja Magang

No.	Kegiatan	Minggu ke-	Tanggal Mulai	Tanggal Selesai
1.	<b>On boarding</b>			
A.	Orientasi lingkungan perusahaan dan mempelajari	1	10-07-2023	14-07-2023

No.	Kegiatan	Minggu ke-	Tanggal Mulai	Tanggal Selesai
	tentang alur pengujian aplikasi demo Medinfrs.			
B.	Pembuatan dokumentasi dan pembaharuan panduan untuk user ( <i>User Guide</i> ) modul EMR.	2-3	17-07-2023	26-07-2023
C.	Perkenalan proyek baru yaitu “QC Automation Testing SIM-RS Medinfrs” dan melakukan instalasi <i>tools</i> Selenium IDE pada browser.	4	27-07-2023	31-07-2023
<b>2.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada menu Pendaftaran Rawat Jalan</b>			
A.	Melakukan <i>testing</i> pada bagian <i>zip code</i> .	5	1-08-2023	7-08-2023
B.	Melakukan <i>testing</i> pendaftaran pasien baru tipe VIP dan pemeriksaan terhadap gambar VIP yang muncul disebelah No. Rekam Medis (RM).	6	8-08-2023	11-08-2023
<b>3.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada menu <i>Appointment</i></b>			
A.	Melakukan <i>testing</i> pada button yang terdapat di <i>appointment</i> , seperti <i>Change Appointment</i> , <i>Void</i> , <i>Void By Paramedic</i> , <i>Registrasi</i> , dan <i>Reschedule All</i> .	7-10	14-08-2023	08-09-2023
<b>4.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada menu Transaksi Klinik</b>			

No.	Kegiatan	Minggu ke-	Tanggal Mulai	Tanggal Selesai
A.	Melakukan <i>testing</i> pada fitur <i>linkbutton</i> untuk menambahkan transaksi klinik menggunakan <i>Quick Picks</i> .	11-13	11-09-2023	20-09-2023
B.	Melakukan pemeriksaan input <i>linkbutton</i> tambah data transaksi klinik.	14	21-09-2023	22-09-2023
<b>5.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada menu Ringkasan Tagihan dan Pembayaran</b>			
A.	Membuat pendaftaran pasien baru dengan tipe pribadi dan instansi serta melakukan propose transaksi dan pembuatan tagihan.	15-16	25-09-2023	10-10-2023
<b>6.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada tab <i>Outpatient Soap</i> menu Asesmen Umum</b>			
A.	Melakukan <i>testing</i> pada <i>fieldset</i> status pasien.	17-18	11-10-2023	27-10-2023
<b>7.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada tab <i>Planning</i> menu <i>Online Prescription</i></b>			
A.	Melakukan <i>testing</i> pada <i>button</i> yang terdapat pada <i>Add Online Prescription</i> , <i>Quick Picks</i> , <i>Quick Picks Template</i> , dan <i>Compound</i> .	19-20	30-10-2023	10-11-2023
<b>8.</b>	<b>Pembuatan <i>command script</i> Selenium IDE pada menu Kamar Operasi</b>			

No.	Kegiatan	Minggu ke-	Tanggal Mulai	Tanggal Selesai
A.	Melakukan testing pada sub menu Jadwal Kamar Operasi.	21	13-11-2023	17-11-2023
B.	Melakukan testing pada sub menu Pengkajian Kamar Operasi.	22-23	20-11-2023	1-12-2023
C.	Melakukan testing pada menu Form Pengkajian Kamar Operasi.	24	4-12-2023	8-12-2023

### 3.2.1 On Boarding

- A. Orientasi lingkungan perusahaan dan mempelajari tentang alur pengujian aplikasi demo Medinfrs.

Pada minggu pertama merupakan masa orientasi lingkungan perusahaan dalam memperkenalkan diri selaku program magang dan diberikan gambaran umum tentang perusahaan. Setelah itu, diarahkan untuk memahami struktur organisasi perusahaan yang bertujuan agar pekerja magang memahami posisinya, yaitu sebagai *Quality Control Intern* dan mengenal tim atau rekan kerja yang lainnya. Di samping orientasi lingkungan perusahaan, minggu pertama pelaksanaan program kerja magang juga mencakup pemahaman tentang alur pengujian aplikasi demo Medinfrs. Medinfrs ini merupakan aplikasi dalam bidang medis atau infrastruktur teknologi kesehatan. Pentingnya aplikasi demo Medinfrs ditekankan melalui pengujian fungsionalitas dan bagaimana aplikasi tersebut dapat berjalan dan berfungsi sebagaimana mestinya.

- B. Pembuatan dokumentasi dan pembaharuan panduan untuk user (*User Guide*) modul EMR.

Pembuatan dokumentasi dan pembaharuan panduan untuk pengguna (*User Guide*) modul *Electroinic Medical Record* (EMR) merupakan suatu proses yang krusial dalam mendukung penggunaan efektif dan efisien dari sistem tersebut. Modul EMR digunakan oleh dokter/perawat memainkan peran sentral dalam manajemen data kesehatan pasien, baik itu pasien yang berasal dari *emergency, outpatient, inpatient, diagnositic support & ancillaries*, maupun *medical checkup*. Panduan pengguna yang jelas menjadi kunci utama dalam memastikan bahwa pengguna mampu memanfaatkannya sebuah sistem aplikasi dengan baik. Oleh karena itu, kegiatan pembuatan dan pembaruan panduan pengguna atau *user guide* menjadi aspek penting dalam peranan seorang *quality control*.

Dalam melaksanakan aktivitas ini, langkah pertama adalah pemahaman mendalam terhadap modul EMR yang akan didokumentasikan. Memahami alur serta fungsi dan penggunaan fitur-fitur yang sudah ada bahkan terbaru yang akan dimasukkan ke dalam modul sejalan kebutuhan pengguna, sehingga pembuatan dan pembaharuan panduan pengguna harus dilakukan secara berkala. Panduan pengguna harus berusaha memberikan wawasan sebanyak mungkin, termasuk mereka yang memiliki tingkat pemahaman teknologi yang berbeda. Oleh karena itu, pembuatan panduan harus melibatkan pengguna yang tidak memiliki latar belakang teknis yang mendalam dan memastikan bahwa istilah atau konsep teknis yang digunakan dapat dikomunikasikan secara sederhana dan jelas.

- C. Perkenalan proyek baru yaitu “QC Automation Testing SIM-RS Medinfrs” dan melakukan instalasi *tools* Selenium IDE pada browser.

Perkenalan proyek baru bernama “QC Automation Testing SIM-RS Medinfrs” merupakan langkah maju yang signifikan dalam pengembangan dan pemeliharaan sistem informasi di PT Quantum

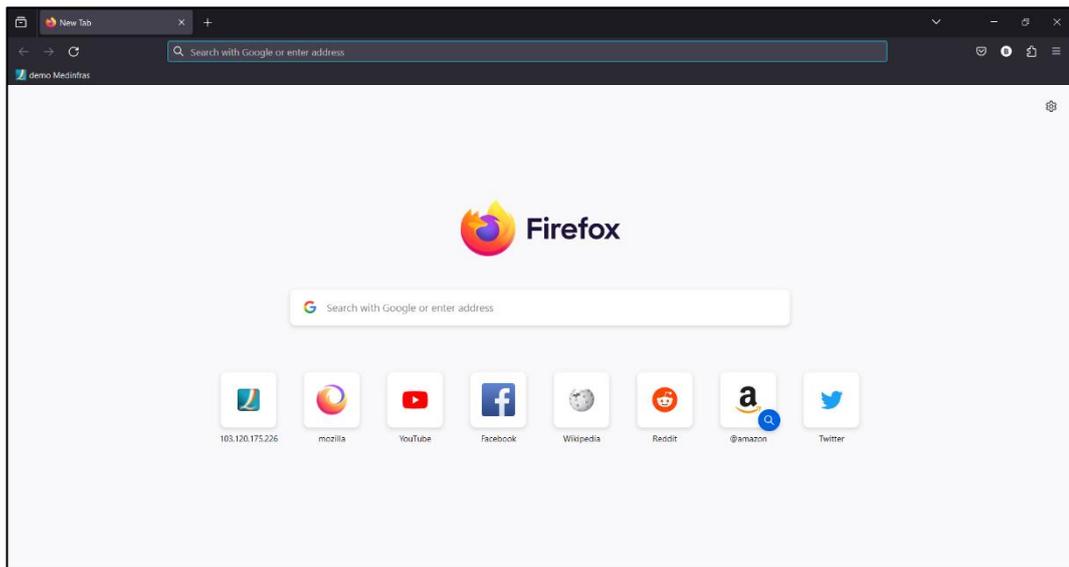
Infra Solusindo. Proyek ini dibuat khusus untuk menjaga tingkat kualitas yang optimal dalam Sistem Informasi Manajemen Rumah Sakit Medinfrs (SIM-RS), sebuah aplikasi penting dalam mendukung operasional rumah sakit. Tujuan utama dari proyek ini adalah untuk meningkatkan efisiensi dan akurasi pengujian aplikasi guna memastikan SIM-RS Medinfrs dapat bekerja dengan baik tanpa bug atau kegagalan yang tidak diinginkan.

Menginstal Selenium IDE di browser Mozilla Firefox adalah tindakan pertama yang dilakukan sebagai bagian dari tahap awal proyek. Skrip pengujian dapat dibuat dan dijalankan secara otomatis dengan Selenium IDE. Tim *Quality Control* mampu membuat rangkaian pengujian otomatis yang efektif untuk memvalidasi berbagai fitur dan modul Medinfrs SIM-RS dengan mengintegrasikan Selenium IDE. Berikut ini merupakan langkah-langkah untuk melakukan instalasi selenium IDE pada browser Mozilla Firefox :

1. Siapkan browser firefox

Pastikan bahwa Mozilla Firefox sudah terinstal di komputer. Jika belum, *download* dan *install* Mozilla Firefox terlebih dahulu.

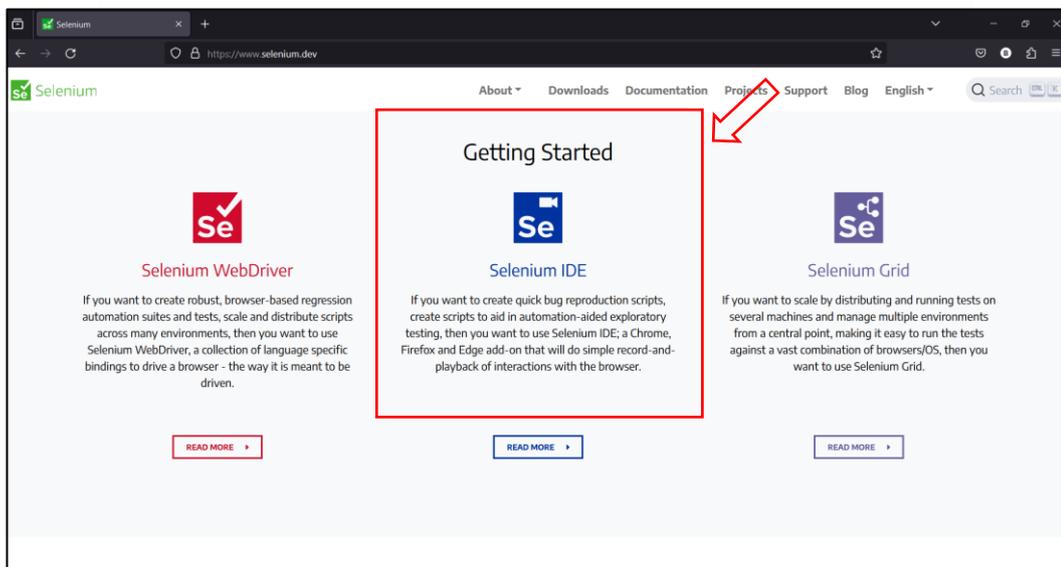
2. Buka browser firefox



Gambar 3. 1 Halaman utama firefox

Pastikan untuk membuka browser Mozilla Firefox sebelum memulai proses instalasi Selenium IDE pada Gambar 3. 1.

### 3. Kunjungi situs resmi selenium

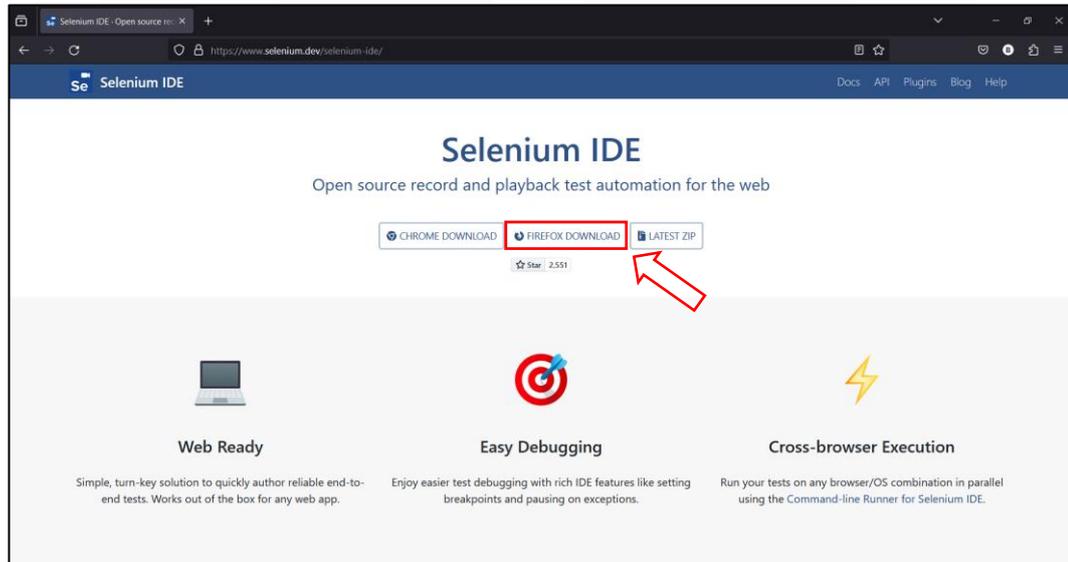


Gambar 3. 2 Halaman website resmi Selenium.

Akses situs resmi Selenium di <https://www.selenium.dev/> pada Gambar 3. 2. Ini adalah sumber utama untuk mendapatkan berbagai

alat dan sumber daya Selenium. Kemudian navigasikan ke bagian Selenium IDE.

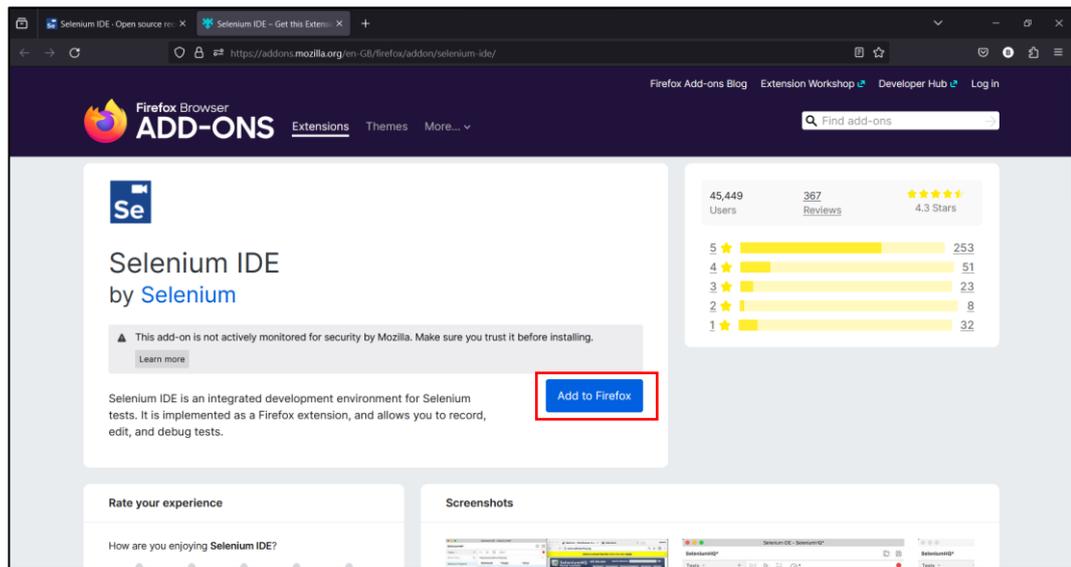
#### 4. Pemilihan browser



Gambar 3. 3 Pemilihan browser untuk install Selenium

Setelah navigasi ke Selenium IDE, silahkan pilih browser yang digunakan, yaitu Mozilla Firefox pada Gambar 3. 3.

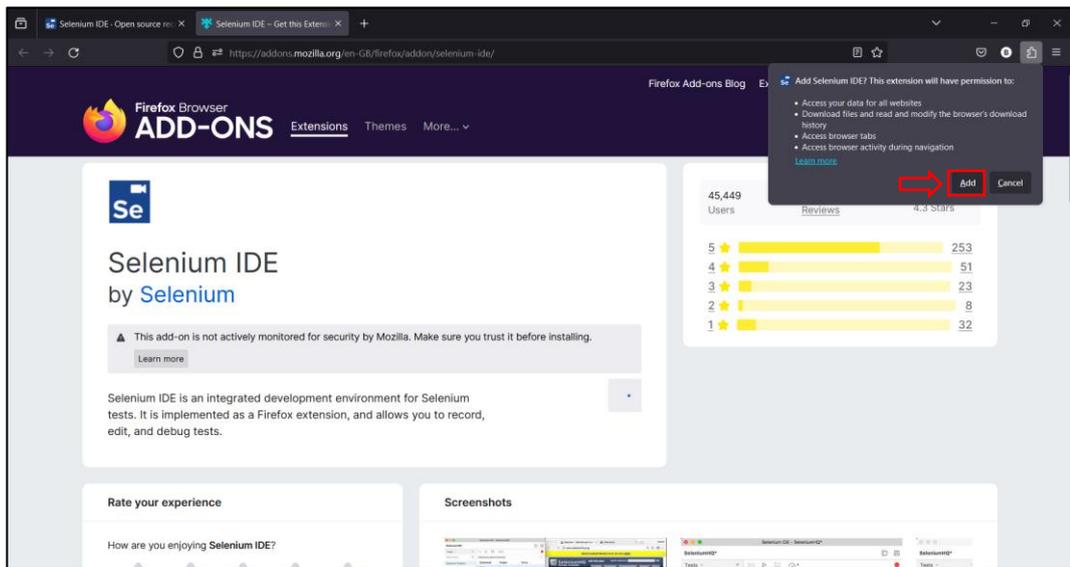
#### 5. Pilih Add to Firefox



Gambar 3. 4 Button add to firefox

Pada halaman utama website untuk mengunduh Selenium IDE, dapat mencari tombol "Download" atau "Add to Firefox" pada Gambar 3. 4. Klik pada tombol tersebut untuk memulai proses pengunduhan dan instalasi. Tunggu sampai proses download selesai dan buka hasil download.

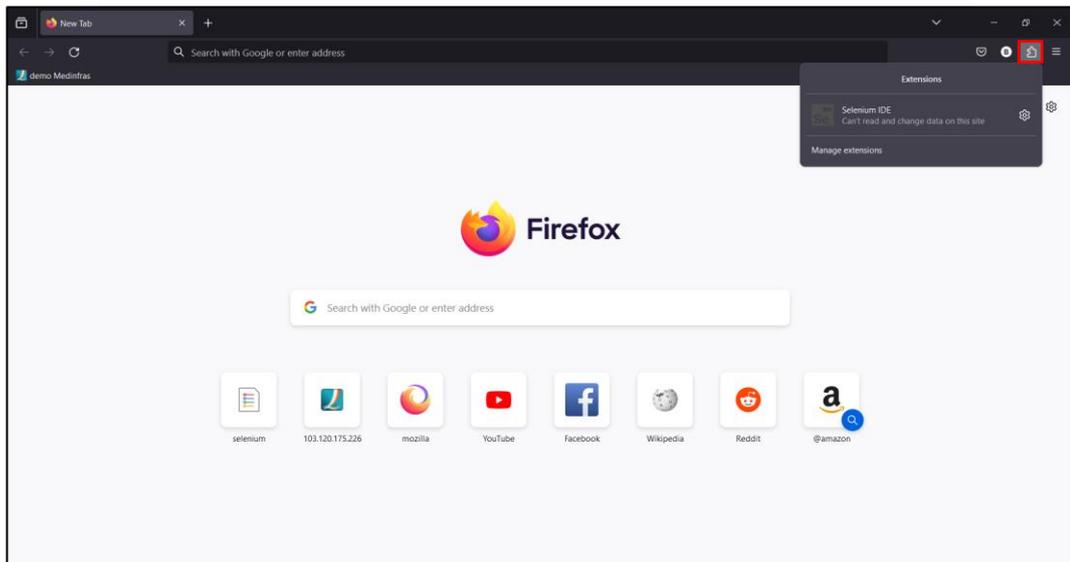
## 6. Konfirmasi instalasi



Gambar 3. 5 Konfirmasi add Selenium

Setelah mengklik tombol "Add to Firefox," Anda mungkin akan diminta untuk mengkonfirmasi instalasi. Klik "Add" atau "Allow" untuk melanjutkan pada Gambar 3. 5.

## 7. Akses Selenium IDE



Gambar 3. 6 Extension Selenium IDE

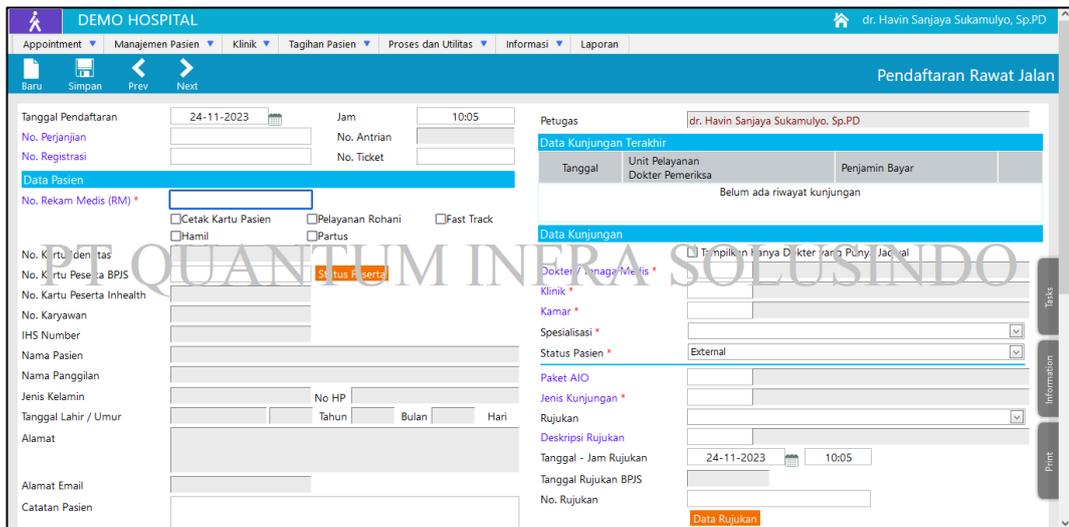
Setelah instalasi selesai, akan diminta untuk me-*restart* Firefox. Setelah Firefox di-*restart*, dapat mengakses Selenium IDE menggunakan menu ekstensi atau toolbar di bagian atas browser pada Gambar 3. 6.

Penerapan otomatisasi pengujian ini bertujuan untuk meningkatkan efisiensi dan ketepatan dalam mendeteksi bug atau ketidaksesuaian dalam fungsionalitas aplikasi. Dengan rangkaian skrip otomatis yang tepat, tim QC dapat melakukan pengujian berulang tanpa keterlibatan manual yang berlebihan, menghemat waktu dan sumber daya. Selain itu, otomatisasi juga meningkatkan akurasi pengujian, mengurangi risiko kesalahan manusia dan memastikan bahwa setiap aspek SIM-RS Medinfrans diuji secara menyeluruh.

### **3.2.2 Pembuatan command script Selenium IDE pada menu Pendaftaran Rawat Jalan.**

#### **A. Melakukan *testing* pada bagian *zip code*.**

Pada tahap awal implementasi *automation testing* menggunakan selenium IDE pada aplikasi demo Medinfrans, proses login merupakan langkah awal yang krusial untuk mengakses berbagai fitur dan modul yang disediakan. Pengguna akan diminta untuk mengisi username dan password pada halaman login. Setelah berhasil login, pengguna akan diarahkan ke halaman aplikasi yang terdapat beberapa modul yang tersedia pada Gambar 3. 7.



Gambar 3. 7 Halaman pendaftaran pasien baru

Modul rawat jalan adalah salah satu fitur penting dalam aplikasi ini, memungkinkan para tenaga medis untuk mendaftarkan pasien baru dengan efisien. Untuk memulai proses pendaftaran pasien, pengguna dapat mengakses modul rawat jalan dari menu utama. Setelah masuk ke modul tersebut, terdapat formulir pendaftaran yang harus diisi dengan informasi pasien. Hal ini melibatkan pengisian data seperti nama lengkap, tanggal lahir, alamat, dan informasi medis pada Gambar 3. 7. Berikut ini merupakan skrip untuk melakukan otomatisasi pendaftaran rawat jalan.

	Command	Target	Value
1	//	Data Pendaftaran Pasien	
2	✓ store	PENYAKIT DALAM (PAGI)	namaKlinik
3	✓ store	havin	namaDokter

Gambar 3. 8 Store data pendaftaran pasien

Pada Gambar 3.8 merupakan beberapa data yang disimpan ke dalam sebuah variabel untuk menyimpan nama klinik dan nama dokter. Variabel yang sudah di *store* pada *test case* yang akan digunakan untuk

melakukan pendaftaran pasien baru. Data-data tersebut berguna apabila sewaktu-waktu tim QC ingin mengubah data pada nama klinik dan nama dokter di kemudian hari.

1	//	TEST LOGIN		
2	✓	open		
3	✓	set window size	1156x678	
4	✓	store attribute	id=ctl00_plhMPBase_pnlUserLoginInformation@style	loginInfo
5	✓	if	\${loginInfo} == "display: block;"	
6		click	id=ctl00_plhMPBase_InkLogout	
7	✓	end		
8	✓	wait for element visible	id=btnLogin	30000
9	✓	type	id=ctl00_plhMPBase_txtUserName	
10	✓	click	id=ctl00_plhMPBase_txtPassword	
11	✓	type	id=ctl00_plhMPBase_txtPassword	
12	✓	click	id=btnLogin	

Gambar 3. 9 Skrip halaman login

Pada Gambar 3. 9 merupakan skrip untuk halaman *user login*, pada baris ke-5 merupakan *if statement* yang digunakan untuk memeriksa keberadaan *style "display: block,"* dari elemen dengan variabel *loginInfo* untuk menentukan apakah pengguna telah *login* atau tidak. Jika pengguna sudah *login*, maka akan melanjutkan tindakan pada baris ke-6, yaitu untuk mengklik *log out* pada halaman.

1	//	TEST PENDAFTARAN PASIEN BARU		
2	✓	open		
3	✓	click	css=.enabled:nth-child(1) > .imgOpenModule	
4	✓	select window	handle=\${win3704}	
5	✓	set window size	1366x1500	
6	✓	wait for element present	xpath=//div/ul/li[2]/a	30000
7	✓	click	xpath=//div/ul/li[2]/a	
8	✓	click	linkText=Pendaftaran Rawat Jalan	
9	✓	click	css=.divOpenRightPanelContent:nth-child(1).textRightPanel	
10	✓	click	linkText=Go	

Gambar 3. 10 Skrip pilih menu pendaftaran rawat jalan

Pada Gambar 3. 10 merupakan skrip yang digunakan untuk memilih menu “Pendaftaran Rawat Jalan”, pengguna harus mengarahkan atau *hover* kursor *mouse* ke menu manajemen pasien. Pada baris ke-5

menggunakan *command set window size* untuk mengatur ukuran halaman menjadi 1366x1500, kemudian Selenium akan mengklik elemen dengan menggunakan Xpath sebagai metode identifikasi. XPath adalah cara untuk menemukan elemen dalam dokumen XML atau HTML. Dalam kasus ini, elemen yang akan diklik berada di dalam div, ul, dan li, dan kemudian mengandung tautan (a). Hal ini akan memilih elemen kedua dalam daftar yaitu, menu “Manajemen Pasien”. Pada baris ke-8 Selenium akan mencari tautan dengan teks "Pendaftaran Rawat Jalan" dan mengkliknya menggunakan CSS Selector yang merupakan anak pertama ":nth-child(1)" dari elemen tersebut.

	Command	Target	Value
10	✓ <i>click</i>	linkText=Go	
11	✓ <i>click</i>	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_txtIdentityCardNo	
12	✓ <i>execute script</i>	return Math.floor(Math.random()*1000000000000000000)	noKTP
13	✓ <i>type</i>	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_txtIdentityCardNo	\$(noKTP)

Gambar 3. 11 Skrip random nomor KTP

Pada Gambar 3. 11 merupakan skrip yang menggunakan metode “Math.random()” untuk menghasilkan nilai acak. Selanjutnya, nilai tersebut dikalikan dengan 1000000000000000000 (16-digit nol) dan dibulatkan menggunakan “Math.floor()”. Hasilnya adalah nomor KTP acak yang memiliki panjang maksimum 16-digit dan akan disimpan kedalam sebuah variabel dengan nama noKTP. Variabel ini akan dipanggil pada baris ke-13 menggunakan perintah *type* ke dalam input HTML dengan id yang disebutkan.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

	Command	Target	Value
14	✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_btFirstName	
15	✓ type	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_btFirstName	Pasien
16	✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_btFamilyName	
17	✓ execute script	return Math.floor(Math.random()*10000)	numRandom
18	✓ type	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_btFamilyName	\${numRandom}
19	✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_cboGender_1	
20	✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_cboGender_DDD_L_LB11T0	

Gambar 3. 12 Skrip random nomor pasien

Pada Gambar 3. 12 juga menggunakan metode “Math.random()” dan “Math.floor()”, namun dengan panjang maksimum 4-digit dan disimpan ke dalam variabel numRandom.

	Command	Target	Value
38	✓ click	css=#ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_btnMPEntypopupSave > img	
39	✓ click	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbpMPEntypopupContent_plhEntry_iblPhysician	
40	✓ click	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	
41	✓ type	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	\${namaDokter}
42	✓ click	css=.imgSearch	
43	✓ click	css=#ctl00_ctl00_ctl00_ctlGeneral_pcSearchDialog_cbpSearchDialog_grdSearch td:nth-child(3)	

Gambar 3. 13 Skrip memanggil variabel nama dokter

Pada Gambar 3. 13 di baris ke-40 dan ke-41 merupakan skrip yang metode identifikasi elemen berdasarkan atribut “name”. Pada baris ke-41 digunakan untuk mengisi nilai pada elemen tersebut, yaitu kolom nama dokter. Nilai yang diisi berasal dari variabel “\${namaDokter}” yang sudah didefinisikan pada Gambar 3. 8.

Command	Target	Value
44 ✓ <i>click</i>	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbp MPEntryContent_plhEntry_lblServiceUnit	
45 ✓ <i>click</i>	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	
46 ✓ <i>type</i>	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	`\${namaKlinik}`
47 ✓ <i>send keys</i>	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	`\${KEY_ENTER}`
48 ✓ <i>click</i>	css=#ctl00_ctl00_ctl00_ctlGeneral_pcSearchDialog_cbpSearchDialog_grdSearch td:nth-child(2)	
49 ✓ <i>click</i>	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbp MPEntryContent_plhEntry_lblRoom	
50 ✓ <i>click</i>	css=#ctl00_ctl00_ctl00_ctlGeneral_pcSearchDialog_cbpSearchDialog_grdSearch tr:nth-child(7) > td:nth-child(2)	

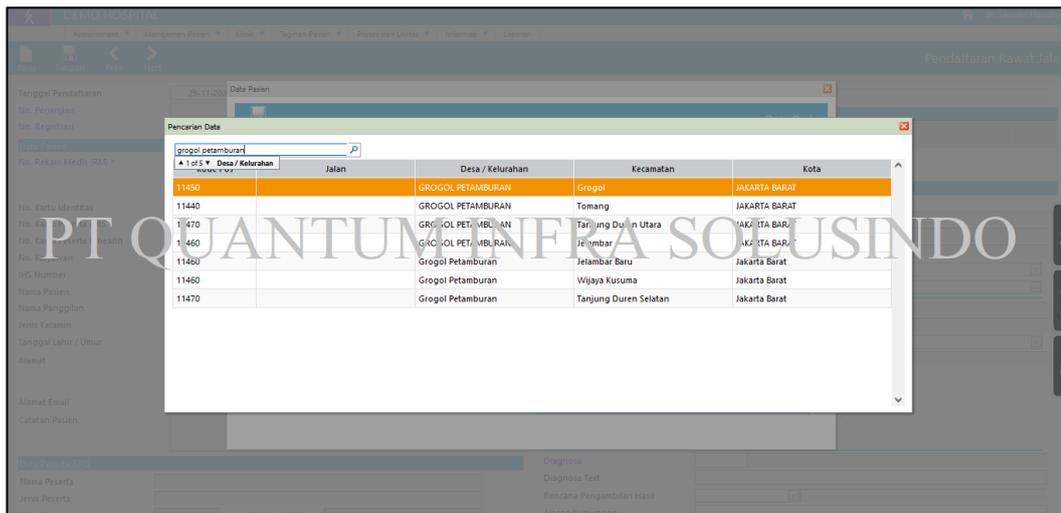
Gambar 3. 14 Skrip memanggil nama klinik

Pada Gambar 3. 14 di baris ke-46 dan ke-47 juga menggunakan skrip yang sama seperti pada Gambar 3.13 yang membedakan hanya element yang digunakan. Pada baris ke-46 digunakan untuk pengisian nama klinik menggunakan perintah “*type*” dengan nilai yang berasal dari variabel “`\${namaKlinik}`” yang sudah didefinisikan pada Gambar 3. 8.

51 ✓ <i>click</i>	css=#ctl00_ctl00_ctl00_plhMPBase_plhMPMain_btnMPEntrySave > img	
52 ✓ <i>store value</i>	name=ctl00\$ctl00\$ctl00\$plhMPBase\$plhMPMain\$cbpMPEntryContent\$plhEntry\$txtPatientName	namaPasien
53 ✓ <i>echo</i>	`\${namaPasien}`	

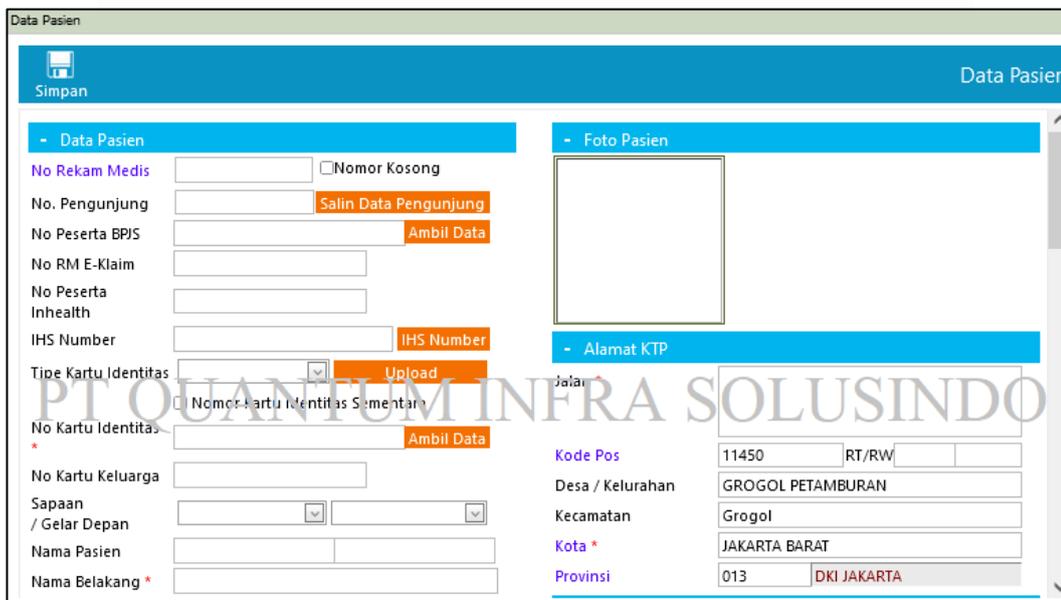
Gambar 3. 15 Store value nama pasien

Pada Gambar 3.15 melakukan perintah *store value* pada baris ke-52 untuk menyimpan nilai nama pasien yang sudah berhasil melakukan pendaftaran dengan variabel yang diberi nama “namaPasien”. Kemudian pada baris ke-53 menggunakan perintah untuk mencetak pesan nilai dari “namaPasien”. Sedangkan untuk halaman pengisian kode pos dapat dilihat pada gambar dibawah ini.



Gambar 3. 16 Halaman pencarian data kode zip

Pada Gambar 3. 16 merupakan tampilan untuk pengisian data zip code atau kode pos dengan melakukan pencarian data di layar pop-up yang tampil. Pengguna akan mengisi textbox pencarian data bisa menggunakan *search paramter* ataupun langsung mengisi dengan nama kota.



Gambar 3. 17 Pengisian otomatis data kode pos

Pada Gambar 3. 17 merupakan halaman setelah pengguna melakukan pencarian data kode pos. Setelah berhasil memilih kode pos, layar pop-up sebelumnya pada Gambar 3. 18 akan otomatis tertutup dan mengisi otomatis data terkait seperti kode pos, desa/kelurahan, kecamatan, kota, dan provinsi pada Gambar 3. 19. Berikut ini merupakan skrip untuk melakukan pemeriksaan kode pos.

	Command	Target	Value
1	// Data Test Kode Pos Link Button		
2	// ***Ubah "Target" pada "Value" : alamat***		
3	✓ store	grogol petamburan	alamat

Gambar 3. 18 Store data pemeriksaan kode pos

Pada Gambar 3. 18 merupakan skrip untuk menyimpan nilai “grogol petamburan” ke dalam sebuah variabel dengan nama “alamat” yang nantinya akan digunakan dalam sebuah *test suites*.

	Command	Target	Value
14	✓ click	linkText=Go	
15	✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntryPopUpContent_ctl03_tblZipCode	
16	✓ click	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	
17	✓ type	name=ctl00\$ctl00\$ctl00\$ctlGeneral\$pcSearchDialog\$ctl24	\${alamat}
18	✓ click	css=.imgSearch	
19	✓ store text	xpath=/html/body/form/div[18]/table/tbody/tr[1]/td[1]/table/tbody/tr[2]/td/div/div/div[1]/div/div/table/tbody/tr[2]/td[4]	yangDiklik
20	✓ click	css=tbody:nth-child(1) > tr:nth-child(5) > td:nth-child(3)	
21	✓ store value	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntryPopUpContent_ctl03_txtCounty	setelahKlik

Gambar 3. 19 Skrip memanggil variabel alamat kode pos

Pada Gambar 3. 19 di baris ke-16 dan ke-17 merupakan skrip yang metode identifikasi elemen berdasarkan atribut “name”. Pada baris ke-17 digunakan untuk mengisi nilai pada elemen tersebut, yaitu kolom pencarian data alamat kode pos pada *pop-up*. Nilai yang diisi berasal dari variabel “\${alamat}” yang sudah didefinisikan pada Gambar 3. 18.

Kemudian pada baris ke-18 melakukan perintah untuk mengklik element yang digunakan untuk melakukan pencarian. Pada baris ke-19 melakukan perintah *store text* yang terletak di posisi yang ditentukan oleh XPath tersebut, kemudian menyimpan teks dari elemen tersebut ke dalam variabel yang ditentukan, yaitu “yangDiklik”. Setelah itu menjalankan perintah *click* terhadap elemen tersebut. Pada baris-19 menggunakan perintah *store value* untuk menyimpan nilai dari suatu elemen di halaman web ke dalam sebuah variabel yang diberi nama “setelahKlik”.

	Command	Target	Value
20	✓ <i>click</i>	css=tbody:nth-child(1) > tr:nth-child(5) > td:nth-child(3)	
21	✓ <i>store value</i>	id=ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntPopUpContent_ctl03_btCounty	setelahKlik
22	✓ <i>echo</i>	\${yangDiklik}	
23	✓ <i>echo</i>	\${setelahKlik}	
24	✓ <i>if</i>	\${setelahKlik}==\${yangDiklik}	
25	✓ <i>echo</i>	QC PASSED (Pengisian Kode Pos Sesuai)	
26	<i>else</i>		
27	<i>echo</i>	QC FAILED (Pengisian Kode Pos Tidak Sesuai)	
28	✓ <i>end</i>		

Gambar 3. 20 Skrip membandingkan nilai kode pos

Pada Gambar 3. 20 merupakan skrip yang digunakan untuk membandingkan dua variabel yang sebelumnya sudah disimpan pada Gambar 3. 18. Pada baris ke-24 menggunakan perintah *if* untuk membandingkan variabel “setelahKlik” sama dengan variabel “yangDiklik”, maka jika kondisi terpenuhi akan mencetak pesan "QC PASSED (Pengisian Kode Pos Sesuai)", apabila tidak terpenuhi akan mencetak pesan "QC FAILED (Pengisian Kode Pos Tidak Sesuai)".

- B. Melakukan *testing* pendaftaran pasien baru tipe VIP dan pemeriksaan terhadap gambar VIP yang muncul disebelah No. Rekam Medis (RM).

Testing pendaftaran pasien dengan status VIP merupakan langkah penting dalam memastikan fungsionalitas dan keakuratan sistem pendaftaran pasien dalam aplikasi demo Medinfras.

Gambar 3. 21 Halaman pilih status pasien

Pada Gambar 3. 21 merupakan tahap pendaftaran pasien tipe VIP, pengguna melakukan ceklis checkbox pada VIP dan memilih jenis VIP. Pengujian ini dilakukan untuk memastikan bahwa proses pendaftaran pasien dengan status VIP.

Gambar 3. 22 Halaman status pasien VIP

Pada Gambar 3. 22 merupakan tanda status pasien VIP disebelah Nomor Rekam Medis (RM). Gambar VIP akan muncul ketika pengguna melakukan registrasi dengan memilih status pasien VIP pada Gambar 3. 21.

Command	Target	Value
45	✓ <i>click</i>	upContent_cti03_chkCopyKTP id=cti00_cti00_cti00_ctiGeneral_pcRightPanelContent_cbpRightPanelContent_cti04_cbpMPEntPop upContent_cti03_chkIsVIP
46	✓ <i>execute script</i>	return document.querySelector("input[type='checkbox']") return document.querySelector("input[id='cti00_cti00_cti00_ctiGeneral_pcRightPanelContent_cbpRightPanelContent_cti04_cbpMPEntPop']") return document.querySelector("input[id='cti00_cti00_cti00_ctiGeneral_pcRightPanelContent_cbpRightPanelContent_cti04_cbpMPEntPop']").checked
47	✓ <i>echo</i>	cekIsVIP
48	✓ <i>if</i>	\$(cekIsVIP)
49	✓ <i>echo</i>	QC PASSED (Ceklis pasien VIP berhasil)
50	<i>else</i>	
51	<i>echo</i>	QC FAILED (Ceklis pasien VIP tidak berhasil)
52	✓ <i>end</i>	

Gambar 3. 23 Skrip pemeriksaan ceklis VIP

Pada Gambar 3. 23 di baris ke-46 merupakan skrip yang menggunakan metode “*document.querySelector()*” untuk mencari elemen pada halaman web yang merupakan sebuah checkbox (kotak centang) dengan ID tertentu. Setelah elemen tersebut ditemukan, kode “*.checked*” diakhir dari skrip tersebut digunakan untuk mengakses properti *checked* dari elemen checkbox tersebut. Dengan demikian, akan mengembalikan nilai *true* jika kotak centang tersebut sedang tercentang, dan nilai *false* jika kotak centang tersebut tidak tercentang. Pada baris ke-47 menggunakan *if* statement untuk membandingkan apakah kotak sudah centang atau tidak, jika mengembalikan nilai *true* maka akan mencetak pesan “QC PASSED (Ceklis pasien VIP berhasil), apabila mengembalikan nilai *false* maka akan mencetak pesan “QC FAILED (Ceklis pasien tidak berhasil)”.

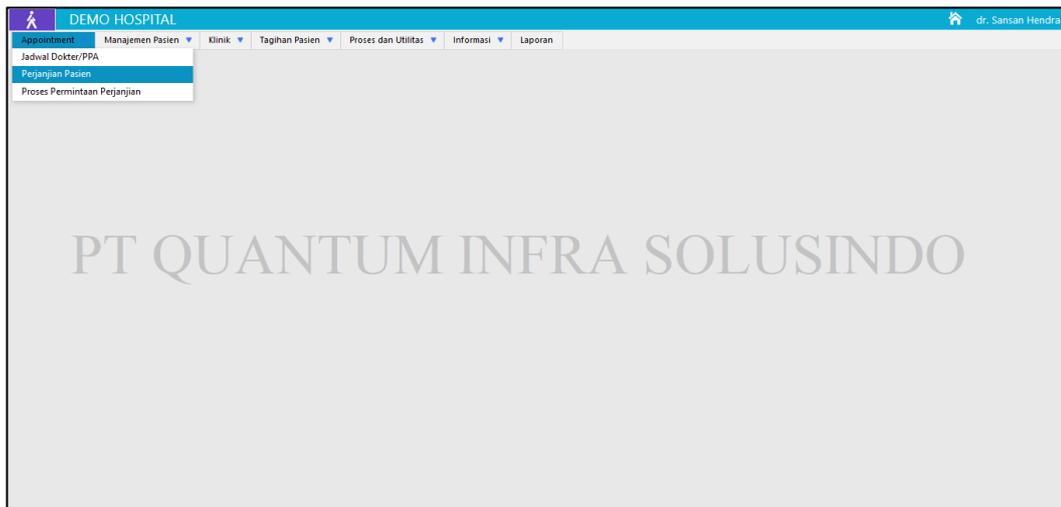
	Command	Target	Value
52	✓ end		
53	✓ click	css=#ct100_ct100_ct100_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ct104_btnMPEntypopuSave > img	
54	✓ store xpath count	xpath=//*[@id="ct100_ct100_ct100_plhMPBase_plhMPMain_cbpMPEntypContent_plhEntry_tdlsvIP"]	gambarVIP
55	✓ echo	`\${gambarVIP}`	
56	✓ if	`\${gambarVIP}` == 1	
57	✓ echo	QC PASSED (Ada Ikon Ceklis VIP)	
58	else		
59	echo	QC FAILED (Tidak Ada Ikon Ceklis VIP)	
60	✓ end		

Gambar 3. 24 Skrip pemeriksaan gambar VIP

Pada Gambar 3. 24 di baris ke-54 digunakan perintah *store xpath count* untuk melakukan penghitungan jumlah elemen menggunakan XPath pada halaman web. Perintah ini digunakan karena sesuai dengan *test scenario* untuk memeriksa gambar VIP yang muncul disebelah Nomor Rekam Medis (RM). Gambar VIP yang muncul hanya berjumlah 1 (satu). Sehingga pada baris ke-56 menggunakan *if statement* untuk membandingkan jika jumlah elemen bernilai 1, maka akan menjalankan perintah mencetak pesan “QC PASSED (Ada Ikon Ceklis VIP)”, jika tidak berjumlah 1 akan mencetak pesan “QC FAILED (Tidak Ada Ikon Ceklis VIP)”.

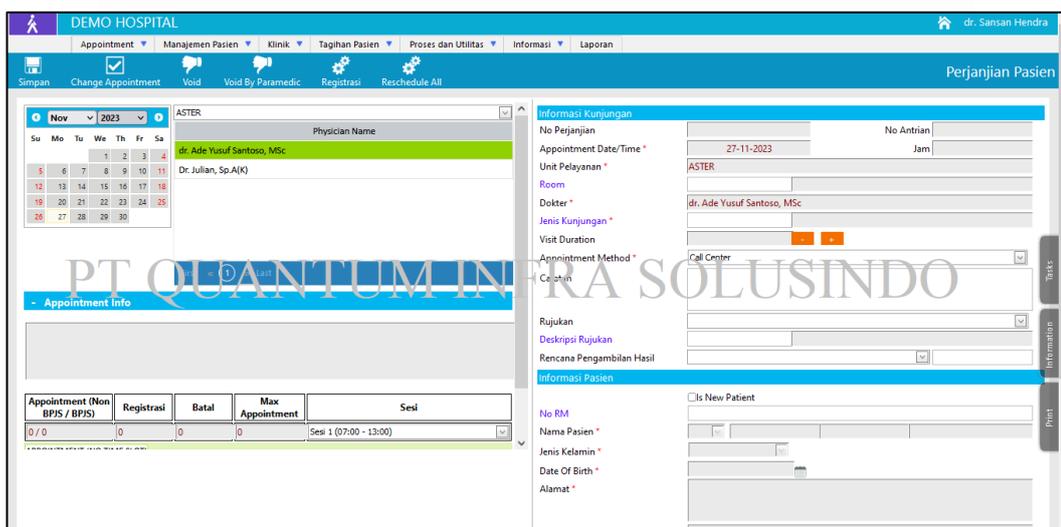
### 3.2.3 Pembuatan command script Selenium IDE pada menu Appointment

- A. Melakukan *testing* pada button yang terdapat di appointment, seperti *Change Appointment*, *Void*, *Void By Paramedic*, *Registrasi*, dan *Reschedule All*.
  - Appointment



Gambar 3. 25 Lokasi menu appointment

Pada Gambar 3. 25 merupakan menu “*Appointment*” dalam aplikasi demo menyediakan akses terhadap berbagai fitur terkait manajemen janji pertemuan pasien dengan dokter. Bagian *appointment* memungkinkan pengguna untuk mengelola dan mengatur janji temu mereka. Dalam aplikasi demo, lokasi untuk melakukan perjanjian pasien berada didalam menu “*Appointment*”.



Gambar 3. 26 Halaman utama menu appointment

Pada Gambar 3. 26 dan biasanya menu ini dirancang dengan antarmuka pengguna yang fungsional. Appinment pada aplikasi demo Medinfras dibagi menjadi 2 tipe, yaitu *time slot* dan *no time slot*. Tipe *time slot* memungkinkan pasien untuk memilih janji temu berdasarkan jadwal waktu yang telah ditentukan sebelumnya. Sistem akan menampilkan jadwal yang tersedia dalam bentuk blok waktu, contohnya setiap 3 menit.

	Command	Target	Value
1	//	Data Test Appoinment Pasien No Time Slot	
2	//	***Ubah "Target" pada "Value" : jumPasien, namaKlinik, tanggal, loopBaris***	
3	✓ store	27	tanggal
4	✓ store	3	jumPasien
5	✓ store	ELDER SYARAF	namaKlinik
6	✓ store	10	loopBaris

Gambar 3. 27 Store data menu appoinment

Pada Gambar 3. 27 merupakan skrip yang digunakan untuk menyimpan semua nilai yang digunakan untuk melakukan pengisian data *appoinment* dalam *test suites*. Adapun data-data yang disimpan dalam variabel “tanggal”, “jumPasien”, “namaKlinik”, dan “loopBaris”. Variabel “tanggal” digunakan untuk mengisi tanggal melakukan *appoinment*, “jumPasien” digunakan untuk membuat *appoinment* sebanyak jumlah pasien yang ditentukan, “namaKlinik” digunakan untuk melakukan pengisian nama klinik, dan “loopBaris” digunakan untuk melakukan pengulangan dan memeriksa baris pada tabel *appoinment* pasien.

	Command	Target	Value
1	//	Test Apoinment Pasien No Time Slot	
2	✓ open		
3	✓ click	css= enabled:nth-child(1) > .imgOpenModule	
4	✓ select window	handle=\${win1390}	
5	✓ wait for element present	xpath=//div/ul/li[1]/a	30000
6	✓ click	xpath=//div/ul/li[1]/a	
7	✓ click	linkText=Perjanjian Pasien	
8	✓ click	linkText=\${tanggal}	
9	✓ type	id=ct100_ct100_ct100_plhMPBase_plhMPMain_cbp MPEntryContent_plhEntry_cboServiceUnit_I	\$(namaKlinik)
10	✓ click	id=ct100_ct100_ct100_plhMPBase_plhMPMain_cbp MPEntryContent_plhEntry_cboServiceUnit_I	

Gambar 3. 28 Skrip pilih menu appoinment

Pada Gambar 3.28 merupakan skrip yang digunakan untuk memilih menu “*Appointment*”, pengguna harus mengarahkan atau *hover* kursor *mouse* ke menu *appointment*. Pada baris ke-5 menggunakan *command wait for element present* untuk menunggu element yang dimaksud muncul, setelah element muncul Selenium akan mengklik elemen dengan menggunakan Xpath sebagai metode identifikasi. Pada baris ke-7 Selenium akan mencari tautan dengan teks "Perjanjuan Pasien" dan mengkliknya. Pada baris ke-8 melakukan perintah *click* yang awalnya “linkText=27” menjadi “linkText=\${tanggal}” untuk mengklik tanggal *appoinment* dengan dengan memanggil variabel yang sudah disimpan pada langkah sebelumnya pada Gambar. 3. 27. Kemudian pada baris ke-9 juga memanggil variabel “namaKlinik” untuk melakukan pengisian dan pemilihan layanan klinik yang dituju.

	Command	Target	Value
8	✓ <i>click</i>	linkText=\${tanggal}	
9	✓ <i>type</i>	id=ct00_ct00_ct00_plhMPBase_plhMPMain_cbp MPEntryContent_plhEntry_cboServiceUnit_I	\$(namaKlinik)
10	✓ <i>click</i>	id=ct00_ct00_ct00_plhMPBase_plhMPMain_cbp MPEntryContent_plhEntry_cboServiceUnit_I	
11	✓ <i>times</i>	\${jumPasien}	
12	✓ <i>store</i>	\${loopBaris}	nomorBaris
13	✓ <i>do</i>		
14	✓ <i>execute script</i>	return Number(\${nomorBaris})-1	nomorBaris
15	✓ <i>store xpath count</i>	xpath=/tr[\${nomorBaris}]/td[3]/ol/li	jumlahXPath
16	✓ <i>repeat if</i>	\$(jumlahXPath)==0	
17	✓ <i>click</i>	xpath=/tr[\${nomorBaris}]/td[3]/ol/li	

Gambar 3. 29 Skrip looping pemilihan baris kosong tipe no time slot

Pada Gambar 3. 29 merupakan skrip untuk memilih baris kosong pada tabel appointment. Memilih baris kosong dalam tabel janji temu dengan tipe no time slot berarti berhasil menemukan slot yang belum terisi oleh janji temu sebelumnya. Sehingga proses ini akan menghasilkan pencatatan janji temu baru dalam sistem aplikasi pada menu “Appointment”. Pada baris ke-11 menunjukkan iterasi yang akan dijalankan sebanyak “\${jumPasien}”, yaitu 3 kali yang dapat dilihat pada Gambar 3. 27. Setiap iterasi akan memperbarui variabel “nomorBaris” dengan nilai loop saat ini. Pada baris ke-14 mengembalikan nilai hasil pengurangan 1 dari nilai variabel “nomorBaris” dan dilanjutkan menghitung jumlah elemen yang cocok di baris ke-15 dengan ekspresi XPath yang diberikan pada baris tertentu dari tabel. Pada baris ke-16 menggunakan perintah *repeat if* untuk melakukan pengulangan iterasi jika jumlah elemen yang ditemukan adalah 0, artinya jika tidak ada elemen yang cocok dengan ekspresi XPath yang diberikan. Pada baris ke-17 jika jumlah elemen sesuai akan menjalankan perintah *click* pada elemen yang sesuai dengan ekspresi XPath yang diberikan pada baris yang diidentifikasi oleh “nomorBaris” pada tabel.

	Command	Target	Value
41	✓ <i>pause</i>	3000	
42	✓ <i>store text</i>	<code>xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldset/table/body/tr[td[1]/div/table/body/tr[2]/td/div[1]/div[2]/div[2]/div/table/body/tr[{\$nomorBaris...</code>	getValue
43	✓ <i>execute script</i>	<code>return "Pasien "+\${numRandom}</code>	cekPasien
44	✓ <i>echo</i>	<code>\${getValue}</code>	
45	✓ <i>echo</i>	<code>\${cekPasien}</code>	
46	✓ <i>if</i>	<code>\${cekPasien}==\${getValue}</code>	
47	✓ <i>echo</i>	QC PASSED (Appointment Pasien Berhasil)	
48	<i>else</i>		
49	<i>echo</i>	QC FAILED (Appointment Pasien Tidak Berhasil)	
50	✓ <i>end</i>		
51	✓ <i>end</i>		

Gambar 3. 30 Skrip membandingkan pasien yang melakukan appointment

Pada Gambar 3. 30 di baris ke-46 menggunakan *if statement* untuk membandingkan nilai variabel “getValue” yang diperoleh dari perintah *store text* di baris ke-42 dan variabel “cekPasien” yang diperoleh mengembalikan nilai berupa teks “Pasien” ditambah dengan nilai dari variabel “numRandom”.

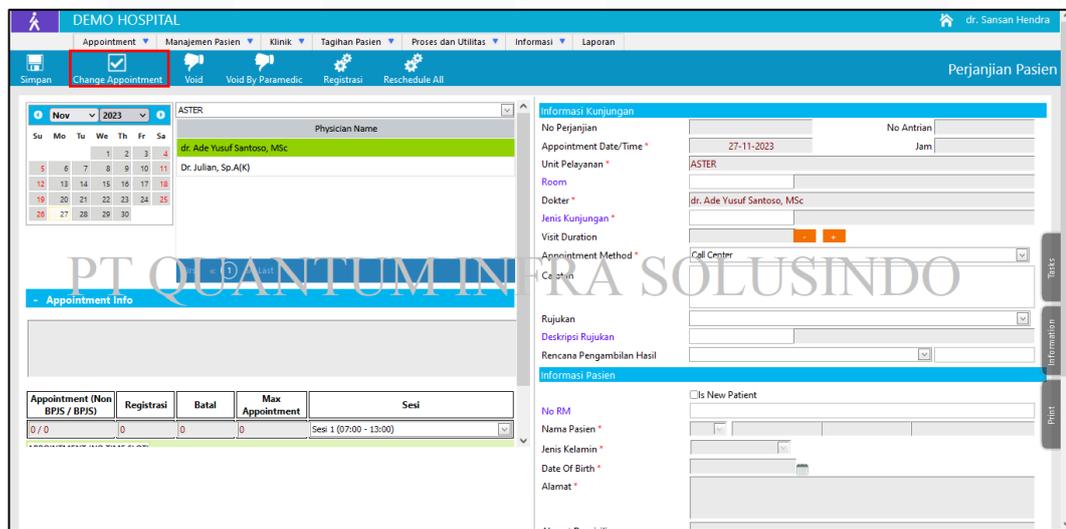
	Command	Target	Value
13	✓ <i>times</i>	<code>\${jumPasien}</code>	
14	✓ <i>store text</i>	<code>xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldset/table/body/tr[td[1]/div/table/body/tr[2]/td/div[1]/div[2]/div[2]/div/table/body/tr[{\$loopBaris}]t...</code>	isiLoopBaris
15	✓ <i>while</i>	<code>\${isiLoopBaris}=""</code>	
16	✓ <i>execute script</i>	<code>return Number(\${loopBaris})+1</code>	loopBaris
17	✓ <i>store text</i>	<code>xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldset/table/body/tr[td[1]/div/table/body/tr[2]/td/div[1]/div[2]/div[2]/div/table/body/tr[{\$loopBaris}]t...</code>	isiLoopBaris
18	✓ <i>end</i>		
19	✓ <i>store</i>	<code>\${loopBaris}</code>	nomorBarisBaru
20	✓ <i>click</i>	<code>xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldset/table/body/tr[td[1]/div/table/body/tr[2]/td/div[1]/div[2]/div[2]/div/table/body/tr[{\$loopBaris}]t...</code>	

Gambar 3. 31 Skrip looping pemilihan baris kosong tipe time slot

Pada Gambar 3. 31 merupakan cara yang serupa untuk memilih baris kosong, namun pada tabel dengan tipe time slot. Pada baris ke-14 dilakukan *store text* untuk menyimpan teks dari elemen yang dipilih menggunakan ekspresi XPath ke dalam variabel bernama “isiLoopBaris”. Pada baris ke-15, perintah *while* ini menunjukkan bahwa operasi yang terkandung di dalamnya akan diulang selama nilai

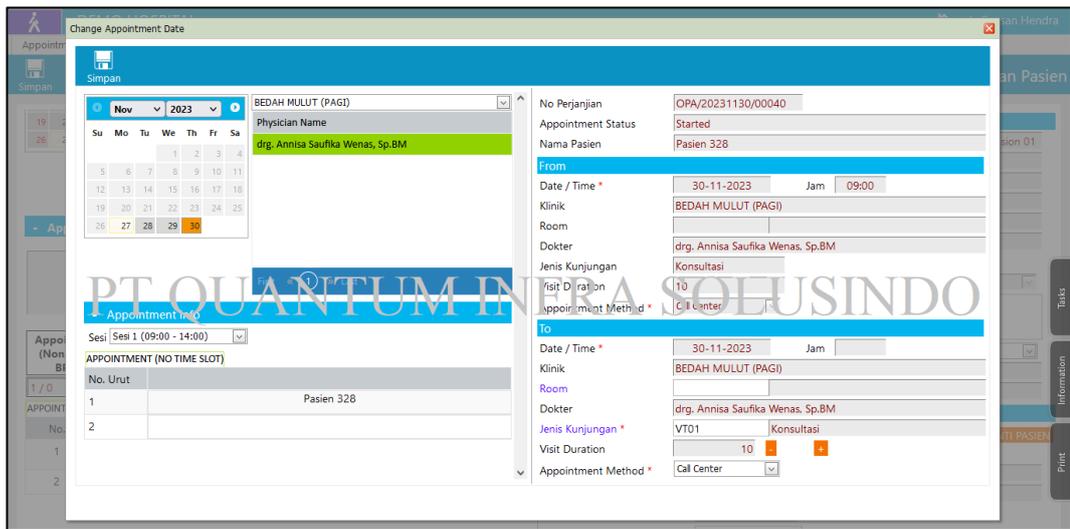
variabel “isiLoopBaris” tidak kosong. Kemudian mengembalikan nilai berupa penjumlahan dari nilai variabel “loopBaris” yang telah diubah menjadi tipe numerik dengan 1. Nilai ini kemudian akan dijadikan sebagai nilai baru untuk variabel “loopBaris”, yang kemungkinan digunakan untuk mengontrol perulangan *while* sampai menemukan nilai “loopBaris” tidak kosong dan mengkliknya.

- Change appointment

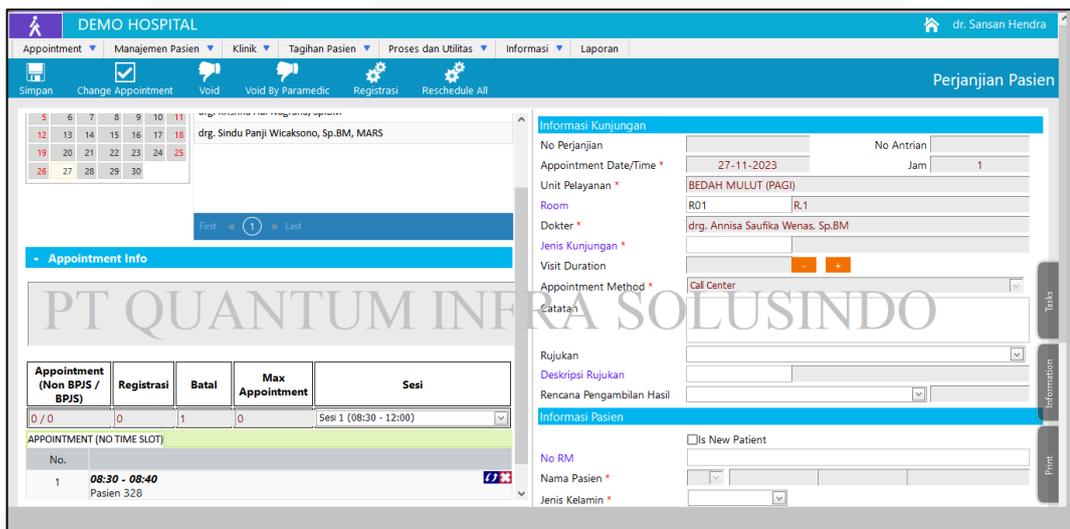


Gambar 3. 32 Button change appointment

Pada Gambar 3. 32 merupakan lokasi button change appointment untuk melakukan perubahan jadwal janji temu. Setelah pengguna mengklik button, akan dinavigasi ke halaman *pop-up* yang bertujuan untuk mengisi data yang diperlukan untuk melakukan perubahan jadwal. Pada Gambar 3. 33 dibawah ini merupakan halaman memilih pasien yang ingin melakukan perubahan jadwal. Pada halaman ini pengguna mengisi data untuk pengganti jadwal *appointment*, seperti tanggal, klinik, kamar, dan jenis kunjungan.

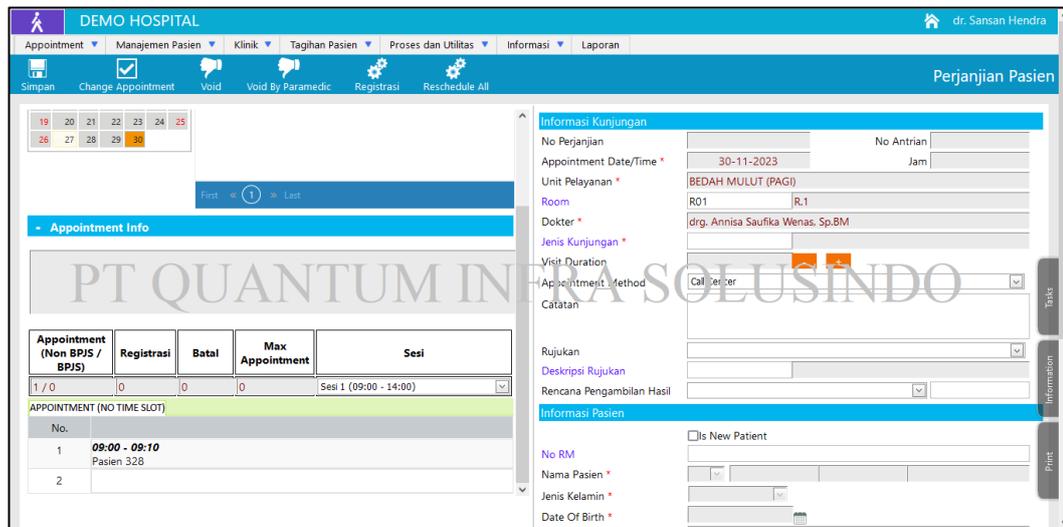


Gambar 3. 33 Layar pop up change appointment



Gambar 3. 34 Sebelum dilakukan change appointment

Pada Gambar 3. 34 merupakan halaman yang menunjukkan pasien sudah berhasil melakukan perubahan jadwal di tanggal 27 yang ditandai dengan adanya icon disebelah kanan baris nama pasien.



Gambar 3. 35 Setelah dilakukan change appointment

Pada Gambar 3. 35 merupakan halaman yang menunjukkan perubahan jadwal *appointment* yang baru. Jadwal awal *appointment* yang awalnya dilakukan pada tanggal 27 secara otomatis akan berubah ke tanggal yang baru.

Command	Target	Value
1 //	Data Test Change Appointment Pasien	
2 //	***Ubah "Target" pada "Value": namaKlinik, loopBaris, tanggal, tanggal2, bulan***	
3	✓ store	BEDAH MULUT (PAGI) namaKlinik
4	✓ store	27 tanggal
5	✓ store	30 tanggal2
6	✓ store	Nov bulan
7	✓ store	10 loopBaris

Gambar 3. 36 Store data change appointment

Pada Gambar 3. 36 terdapat penambahan variabel “tanggal2” yang digunakan untuk memilih tanggal pergantian *appointment* dan variabel “bulan” untuk memilih bulan pada menu dropdown.

Command	Target	Value
49 ✓ click	xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldsel/table/body/tr[1]/div/table/body/tr[2]/td/div[1]/div[2]/div[3]/div/table/body/tr[1]/td	
50 ✓ click	id=ctl00_ctl00_plhMPBase_plhMPMain_plhCustomButtonToolbar_btnAppointmentChangeAp pointment	
51 ✓ select	xpath=/html/body/form/div[7]/div[1]/table/body/tr[1]/td[1]/table/body/tr[2]/td/div/div[1]/div/table/body/tr[1]/td[1]/div/table/body/tr[1]/td[1]/div/div/div/div/select[1]	\$(bulan)
52 ✓ click	linkText=\${tanggal2}	
53 ✓ store	\$(loopBaris)	nomorBaris2
54 ✓ do		
55 ✓ execute script	return Number(\$(nomorBaris2))-1	nomorBaris2
56 ✓ store xpath count	xpath=/html/body/form/div[7]/div[1]/table/body/tr[1]/td[1]/table/body/tr[2]/td/div/div[1]/div/table/body/tr[1]/td[1]/div/table/body/tr[2]/td/div/div[1]/div/div[4]/div/t...	jumlahXPath2

Gambar 3. 37 Skrip memanggil variabel change appoinment

Pada Gambar 3. 37 merupakan skrip yang digunakan untuk memanggil variabel yang sudah disimpan pada Gambar 3. 36. Baris ke-51 menggunakan perintah *select* untuk memilih nama bulan yang terdapat pada menu *dropdown list*. Kemudian pada baris ke-52 digunakan perintah untuk mengklik tanggal pergantian *appointment* dengan memanggil variabel “tanggal2”.

Command	Target	Value
55 ✓ execute script	return Number(\$(nomorBaris2))-1	nomorBaris2
56 ✓ store xpath count	xpath=/html/body/form/div[7]/div[1]/table/body/tr[1]/td[1]/table/body/tr[2]/td/div/div[1]/div/table/body/tr[1]/td[1]/div/table/body/tr[2]/td/div/div[1]/div/div[4]/div/t...	jumlahXPath2
57 ✓ repeat if	\$(jumlahXPath2)==0	
58 ✓ click	xpath=/html/body/form/div[7]/div[1]/table/body/tr[1]/td[1]/table/body/tr[2]/td/div/div[1]/div/table/body/tr[1]/td[1]/div/table/body/tr[2]/td/div/div[1]/div/div[4]/div/t...	
59 ✓ click	css=#btnMPEnterSave > img	
60 ✓ click	linkText=\${tanggal2}	
61 ✓ store	\$(nomorBaris2)	barisBaru2
62 ✓ execute script	return 'Pasien '+\$(numRandom)	cekPasien2

Gambar 3. 38 Skrip looping baris kosong dalam pop up change appoinment

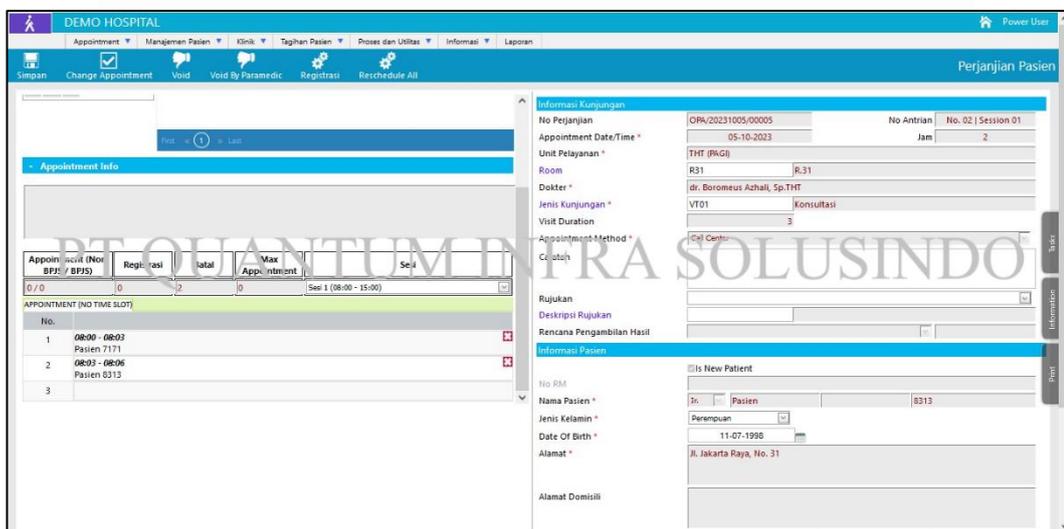
Pada baris ke-55 mengembalikan nilai hasil pengurangan 1 dari nilai variabel “nomorBaris2” dan di baris ke-56 dilanjutkan dengan menghitung jumlah elemen yang cocok di baris dengan ekspresi XPath yang diberikan pada baris tertentu dari tabel. Baris ke-57 menggunakan perintah *repeat if* untuk melakukan pengulangan iterasi jika jumlah elemen yang ditemukan adalah 0.

	Command	Target	Value
60	✓ click	linkText=\${tanggal2}	
61	✓ store	\${nomorBaris2}	barisBaru2
62	✓ execute script	return 'Pasien '+\${numRandom}	cekPasien2
63	✓ echo	\${cekPasien2}	
64	✓ echo	\${cekPasien}	
65	✓ if	\${cekPasien2}==\${cekPasien}	
66	✓ echo	QC PASSED (Change Appointment Pasien Baru Berhasil)	
67	else		
68	echo	QC FAILED (Change Appointment Pasien Baru Tidak Berhasil)	
69	✓ end		

Gambar 3. 39 Skrip membandingkan change appointment

Pada Gambar 3. 39 merupakan skrip yang digunakan untuk membandingkan nama pasien yang melakukan *change appointment* pada tanggal awal dengan tanggal yang baru. Jika kondisi sesuai maka akan mencetak pesan “QC PASSED (Change Appointment Pasien Baru Berhasil)”.

- Void



Gambar 3. 40 Halaman void appointment

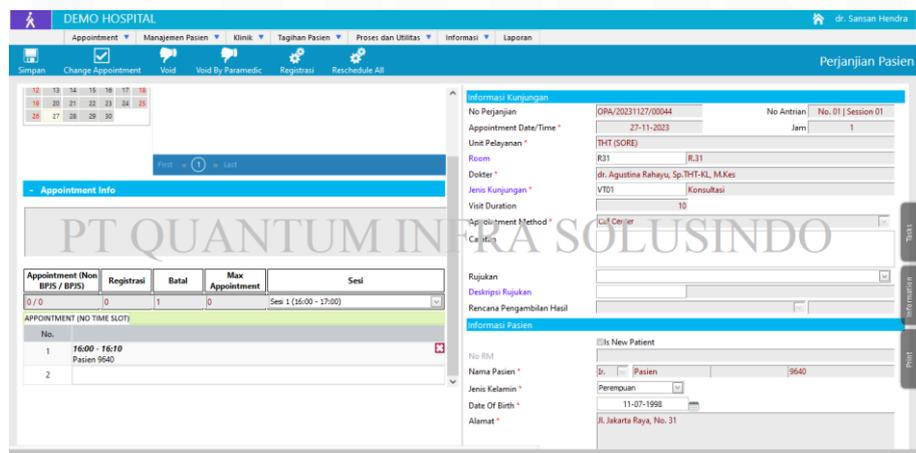
Pada Gambar 3. 40 merupakan halaman utama void yang merupakan tindakan pembatalan atau penghapusan janji temu yang telah dijadwalkan sebelumnya.

Command	Target	Value
53	type	type
54	click	css=#ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_btnMPEEntryPopupSave > img
55	store xpath count	jumlahImage
56	if	QC PASSED (Void Pasien Berhasil, Ada Icon Delete)
57	echo	QC PASSED (Void Pasien Berhasil, Ada Icon Delete)
58	else	QC FAILED (Void Pasien Tidak Berhasil, Tidak Ada Icon Delete)
59	echo	QC FAILED (Void Pasien Tidak Berhasil, Tidak Ada Icon Delete)
60	end	

Gambar 3. 41 Skrip void appointment

Pada Gambar 3. 41 merupakan skrip yang digunakan untuk memeriksa pembatalan appointment berhasil atau tidak dengan mengecek icon “x” berwarna merah disebelah nama pasien. Jika terdapat icon tersebut maka void yang dilakukan telah berhasil menggunakan perintah store xpath count untuk menghitung jumlah elemen yang muncul pada baris dengan ekspresi XPath yang diberikan.

- Void by paramedic



Gambar 3. 42 Halaman void by paramedic

Pada Gambar 3. 42 merupakan halaman utama *void by paramedic* yang merupakan tindakan pembatalan atau penghapusan janji temu yang telah dijadwalkan sebelumnya, namun proses pembatalan janji temu dilakukan oleh petugas paramedis atau tenaga medis yang bertanggung jawab atas janji temu tersebut.

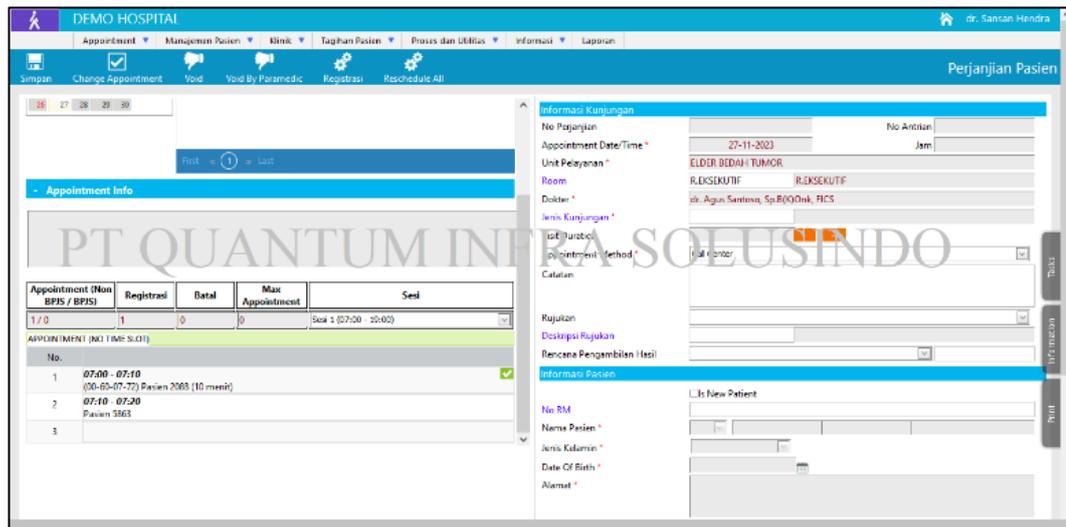
Command	Target	Value
✓ click	plhCustomButtonToolbar_btnAppointmentVoidByParamedic > img	
51 ✓ click	id=ctf00_ctf00_ctf00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctf04_btnMPEnterPopUpSave	
52 ✓ store xpath count	xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldset/table/tbody/tr[1]/div/table/tbody/tr[2]/td/div[1]/div[2]/div[3]/div/table/tbody/tr[1]/td	jumlahImage
53 ✓ if	\$(jumlahImage) == 1	
54 ✓ echo	QC PASSED (Void By Paramedic Berhasil, Ada Icon Delete)	
55 else		
56 echo	QC FAILED (Void By Paramedic Tidak Berhasil, Tidak Ada Icon Delete)	
57 ✓ end		

Gambar 3. 43 Skrip void by paramedic

Pada Gambar 3. 43 merupakan skrip yang digunakan untuk *void by paramedic* berhasil atau tidak dengan mengecek icon “x” berwarna merah disebelah nama pasien. Jika terdapat icon tersebut maka *void by paramedic* yang dilakukan telah berhasil menggunakan perintah store xpath count untuk menghitung jumlah elemen yang muncul pada baris dengan ekspresi XPath yang diberikan.

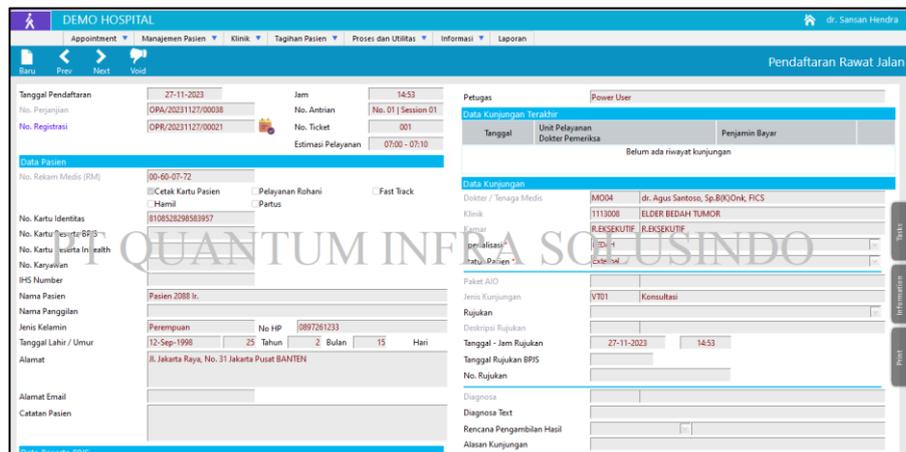
- Registrasi

Pada halaman registrasi ini, pasien diminta untuk memberikan data pribadi seperti nama lengkap, alamat, nomor telepon, dan informasi asuransi kesehatan. Selain itu, halaman ini juga mencakup sejumlah pertanyaan kesehatan dasar yang membantu tenaga medis memahami kondisi kesehatan pasien, riwayat penyakit, serta alergi yang perlu diperhatikan.



Gambar 3. 44 Button registrasi appoinment

Pada Gambar 3. 44 merupakan letak button “Registrasi” pada halaman utama *appointment*. Registrasi appointment merupakan proses di mana seorang pasien mendaftarkan diri mereka untuk mendapatkan layanan kesehatan atau membuat janji temu dengan penyedia layanan medis atau dokter tertentu.



Gambar 3. 45 Halaman registrasi appoinment

Pada Gambar 3. 45 merupakan halaman registrasi appointment, di mana pengguna dapat mendaftar atau membuat janji temu dengan penyedia layanan, seperti dokter atau layanan kesehatan lainnya.

Halaman ini memungkinkan pengguna untuk mengisi informasi pribadi mereka dan mengatur janji temu untuk mendapatkan layanan yang mereka butuhkan

Command	Target	Value
50 ✓ click	xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/fieldset/table/tbody/tr[td[1]/div/table/tbody/tr[2]/td/div[1]/div[2]/div[3]/div/table/tbody/tr[1]/td	
51 ✓ store value	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbpMPEntryContent_plhEntry_btAppointmentNo	noPerjanjian
52 ✓ store value	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbpMPEntryContent_plhEntry_btPhysician	dokter
53 ✓ click	css=#ctl00_ctl00_ctl00_plhMPBase_plhMPMain_plhCustomButtonToolBar_btnAppointmentProcessRegistration > div	
54 ✓ click	css=btnClose:nth-child(2)	
55 ✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntryPopupContent_ctl03_txtIdentityCardNo	
56 ✓ execute script	return Math.floor(Math.random()*1000000000000000)	numRandom
57 ✓ type	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntryPopupContent_ctl03_txtIdentityCardNo	\${numRandom}

Gambar 3. 46 Skrip store value nomor perjanjian dan dokter

Pada Gambar 3. 46 di baris ke-51 merupakan skrip yang digunakan untuk menyimpan nomor perjanjian yang berhasil dibuat dengan variabel yang diberi nama “noPerjanjian”, kemudian di baris ke-52 menyimpan nama dokter dengan variabel “dokter”.

Command	Target	Value
80 ✓ click	PopupSave > img css=#ctl00_ctl00_ctl00_plhMPBase_plhMPMain_btnMPEntrySave > img	
81 ✓ store value	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbpMPEntryContent_plhEntry_btAppointmentNo	noPerjanjian2
82 ✓ store value	id=ctl00_ctl00_ctl00_plhMPBase_plhMPMain_cbpMPEntryContent_plhEntry_btPhysicianName	dokter2
83 ✓ echo	\${noPerjanjian} (Nomor Perjanjian 1)	
84 ✓ echo	\${noPerjanjian2} (Nomor Perjanjian 2)	
85 ✓ if	\${noPerjanjian}=\${noPerjanjian2}	
86 ✓ echo	QC PASSED (Nomor Perjanjian Sama)	
87 else		
88 echo	QC FAILED (Nomor Perjanjian Tidak Sama)	
89 ✓ end		

Gambar 3. 47 Skrip membandingkan nomor perjanjian

Pada Gambar 3. 47 merupakan skrip yang digunakan untuk membandingkan nomor perjanjian. Baris ke-81 melakukan perintah untuk menyimpan nilai dengan variabel yang diberi nama

“noPerjanjian2”, kemudian pada baris ke-82 juga menyimpan nama dokter dengan variabel yang diberi nama “dokter2”. Pada baris ke-86 digunakan perintah if statement untuk membandingkan variabel “noPerjanjian” sama dengan “noPerjanjian2”. Jika kondisi terpenuhi akan mencetak pesan “QC PASSED (Nomor Perjanjian Sama)”.

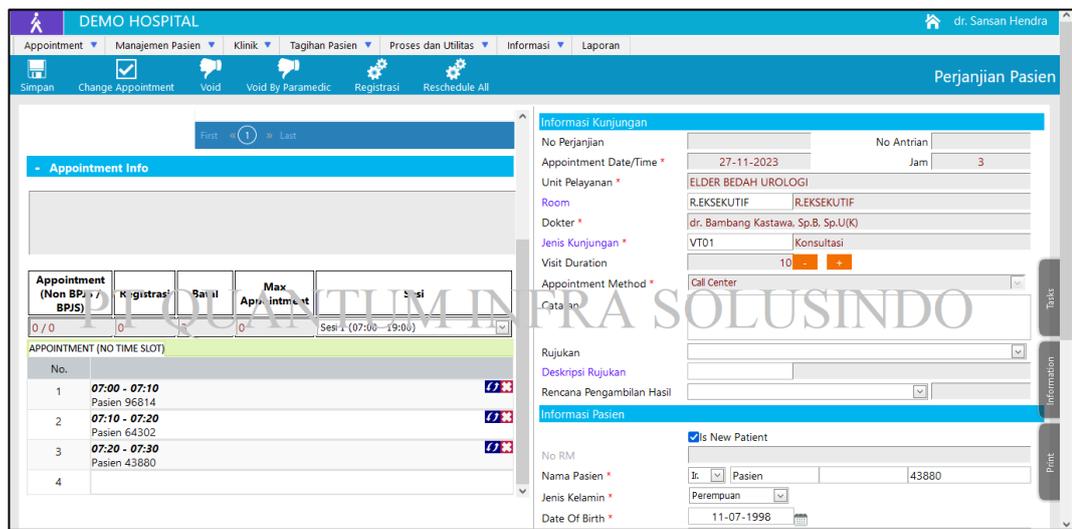
	Command	Target	Value
86	✓ echo	QC PASSED (Nomor Perjanjian Sama)	
87	else		
88	echo	QC FAILED (Nomor Perjanjian Tidak Sama)	
89	✓ end		
90	✓ echo	\${dokter} (Dokter 1)	
91	✓ echo	\${dokter2} (Dokter 2)	
92	✓ if	\${dokter}==\${dokter2}	
93	✓ echo	QC PASSED (Dokter Sama)	
94	else		
95	echo	QC FAILED (Dokter Tidak Sama)	
96	✓ end		

Gambar 3. 48 Skrip membandingkan nama dokter

Pada Gambar 3. 48 merupakan skrip yang digunakan untuk membandingkan nilai dari variabel “dokter” sama dengan “dokter2”. Jika kondisi terpenuhi akan mencetak pesan “QC PASSED (Dokter Sama)”.

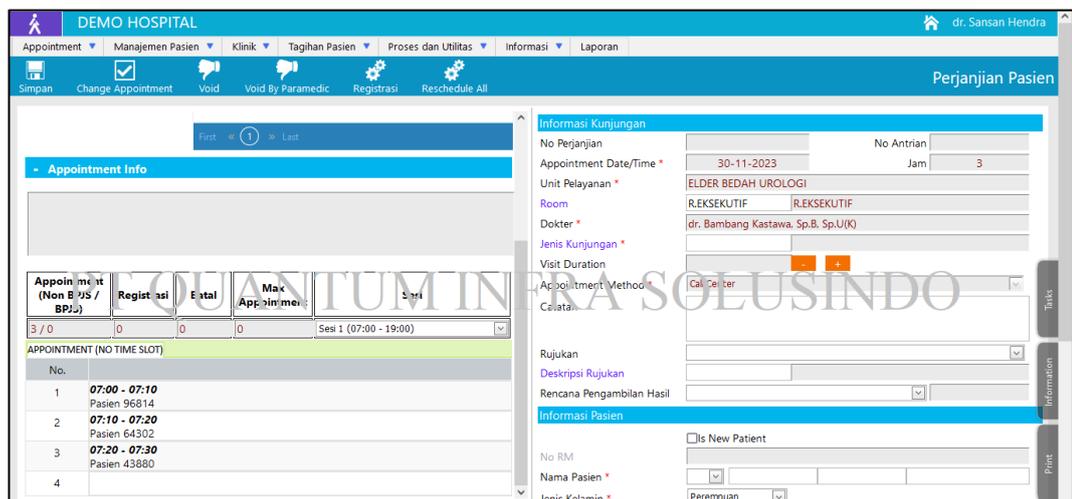
- Reschedule all

Fitur “Reschedule All” pada *appointment* pasien bertujuan untuk memberikan kemudahan bagi pengguna untuk mengatur ulang atau mengubah jadwal seluruh janji temu yang telah dijadwalkan. Fitur ini sangat bermanfaat dalam situasi di mana perlu dilakukan penyesuaian jadwal secara kolektif, seperti perubahan jam operasional, penjadwalan ulang massal, atau peristiwa khusus yang memengaruhi *appointment*.



Gambar 3. 49 Jadwal appointment sebelum reschedule all

Pada Gambar 3. 49 merupakan jadwal awal pasien yang ingin melakukan appointment. Ketika dilakukan reschedule all pada jadwal yang sudah dibuat akan muncul *icon* disebelah kanan baris nama pasien dan akan muncul jadwal baru sesuai dengan tanggal baru yang sudah ditetapkan.



Gambar 3. 50 Jadwal appointment setelah reschedule all

Pada Gambar 3. 50 merupakan jadwal baru appointment yang sudah dilakukan reschedule all. Pada jadwal baru tidak menampilkan *icon*

disebelah kanan baris nama pasien yang menjadi penanda status bahwa jadwal telah mengalami perubahan.

	Command	Target	Value
1	//	Data Test Reschedule All Pasien	
2	//	***Ubah "Target" pada "Value": jumPasien, namaKlinik, tanggal, loopBaris***	
3	✓ store	3	jumPasien
4	✓ store	ELDER BEDAH UROLOGI	namaKlinik
5	✓ store	27	tanggal
6	✓ store	ELDER BEDAH UROLOGI	toKlinik
7	✓ store	30	tanggal2
8	✓ store	Nov	bulan2
9	✓ store	10	loopBaris

Gambar 3. 51 Store data reschedule all

Pada Gambar 3. 51 merupakan sejumlah data yang sama seperti *appointment*, namun terdapat beberapa tambahan seperti vairabel “toKlinik” yang digunakan untuk mengubah nama klinik tujuan, variabel “tanggal2” dan “bulan2” yang digunakan untuk menentukan pengganti tanggal dan bulan untuk melakukan appointment.

	Command	Target	Value
55	✓ click	id=ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_cboServiceUnitRescheduleToI	
56	✓ type	id=ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_cboServiceUnitRescheduleToI	\$(toKlinik)
57	✓ send keys	id=ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopupContent_ctl03_cboServiceUnitRescheduleToI	\$(KEY_ENTER)
58	✓ select	xpath=/html/body/form/div[7]/div[1]/table/tbody/tr[1]/td[1]/table/tbody/tr[2]/td/div/div[1]/div/div[2]/div[1]/fieldset/div/table/tbody/tr[2]/table[1]/tbody/tr[td[1]...	\$(bulan2)
59	✓ click	xpath=//a[contains(text(),\${tanggal2})][2]	
60	✓ click	id=ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_btnMPEntypopupSave	

Gambar 3. 52 Skrip memanggil variabel reschedule all

Pada Gambar 3. 52 merupakan skrip untuk memanggil variabel yang sudah disimpan pada Gambar 3. 51. Variabel ini akan digunakan untuk pengisian data perubahan jadwal kunjungan appointment.

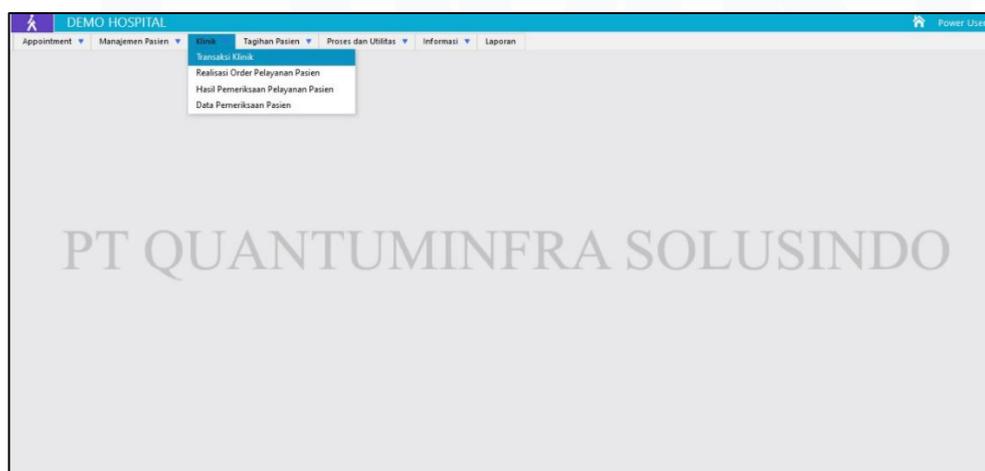
Command	Target	Value
59 ✓ click	<code>xpath=//a[contains(text(),\${tanggal2})][2]</code>	
60 ✓ click	<code>id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_btnMPEnterPopUpSave</code>	
61 ✓ store xpath count	<code>xpath=/html/body/form/div[27]/div/div[2]/div[2]/div[1]/table/tbody/tr/td[1]/div/table/tbody/tr[2]/td/div/div[1]/div[2]/div[3]/div/table/tbody/tr[1]/td</code>	jumlahImgReschedule
62 ✓ if	<code>\$(jumlahImgReschedule)=1</code>	
63 ✓ echo	QC PASSED (Reschedule Berhasil, Ada Icon Reschedule)	
64 else		
65 echo	QC FAILED (Reschedule All Tidak Berhasil, Tidak Ada Icon Reschedule)	
66 ✓ end		

Gambar 3. 53 Skrip memeriksa icon reschedule all

Pada Gambar 3. 53 merupakan skrip yang digunakan untuk memeriksa apakah jadwal yang sudah dilakukan reshchedule all terdapat icon atau tidak. Cara pemeriksaannya dilakukan pada baris ke-61, yaitu dengan perintah store xpath count untuk menghitung jumlah elemen yang muncul pada salah satu baris dengan ekspresi XPath yang diberikan.

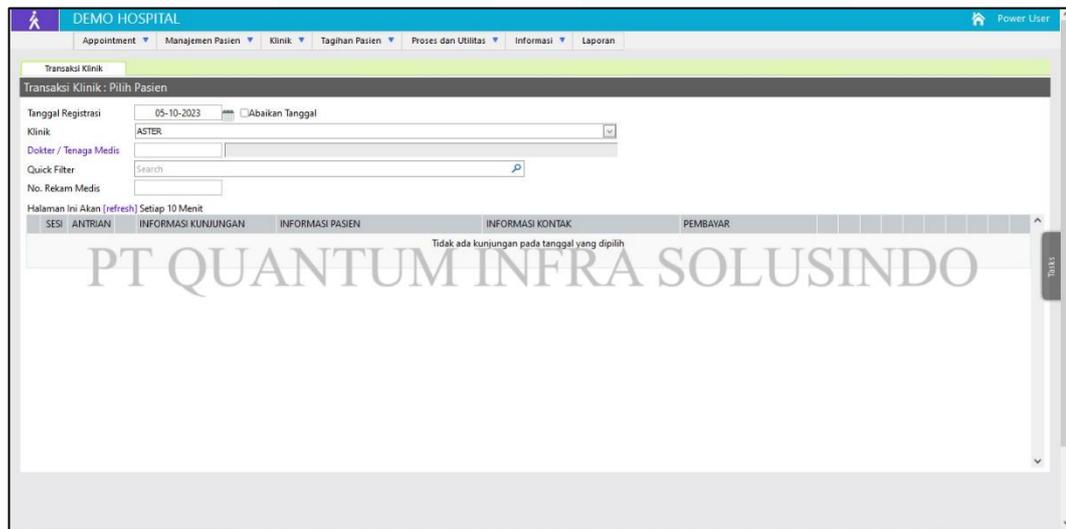
### 3.2.4 Pembuatan command script Selenium IDE pada menu Transaksi Klinik

A. Melakukan *testing* pada fitur *linkbutton* untuk menambahkan transaksi klinik menggunakan *Quick Picks*.



Gambar 3. 54 Lokasi menu transaksi klinik

Pada Gambar 3. 54 merupakan halaman aplikasi demo yang menunjukkan lokasi menu “Transaksi Klinik”, yaitu ketika pengguna melakukan *hover* kursor ke menu “Klinik”.



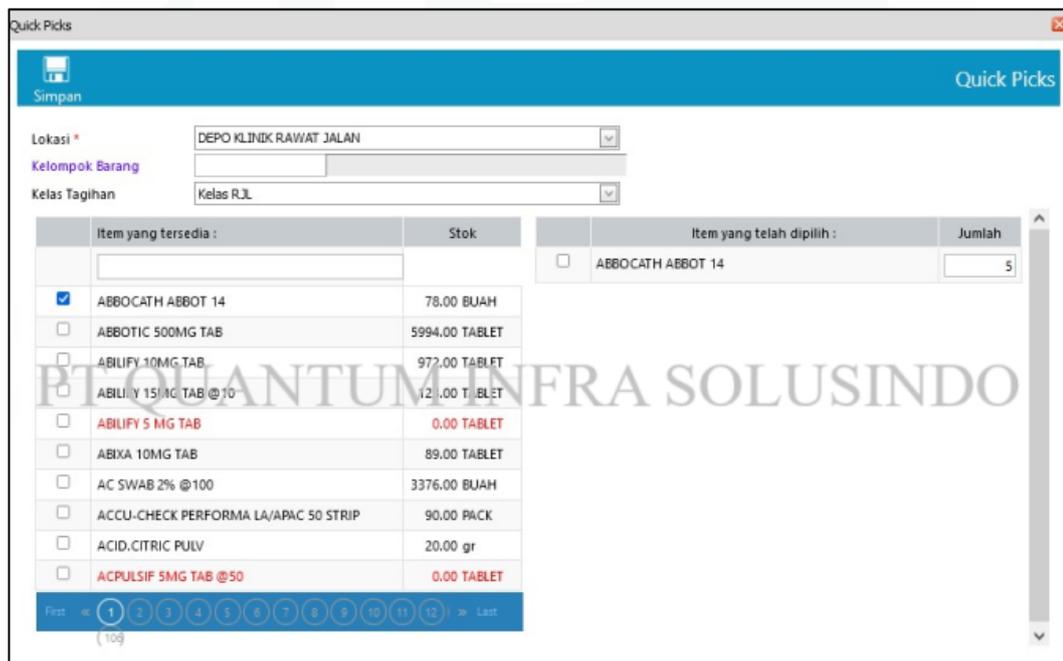
Gambar 3. 55 Halaman memilih pasien untuk transaksi klinik

Pada Gambar 3. 55 merupakan halaman yang digunakan untuk memilih asal pasien yang sudah didaftarkan. Pasien yang dipilih nantinya akan diarahkan ke halaman untuk melakukan transaksi klinik.



Gambar 3. 56 Halaman utama transaksi klinik

Pada Gambar 3. 56 merupakan halaman utama untuk melakukan transaksi klinik. Pada halaman ini terdapat 2 fitur *linibutton*, yaitu “Tambah Data” dan “Quick Picks” yang kegunaan masing-masing juga untuk menambahkan item untuk transaksi klinik.



Gambar 3. 57 Pop up linkbutton quick picks transaksi klinik

Pada Gambar 3. 57 merupakan halaman pop-up ketika pengguna mengklik *linkbutton*. Halaman ini digunakan untuk memilih item dan jumlah.



Gambar 3. 58 Berhasil melakukan transaksi klinik menggunakan quick picks

Pada Gambar 3. 58 merupakan halaman ketika berhasil melakukan transaksi klinik menggunakan *linkbutton* “Quick Picks. Pengguna dapat melihat rincian item dan harga obat yang digunakan selama masa perawatan di rumah sakit.

	Command	Target	Value
17	✓ <i>click</i>	css=#ulTabClinicTransaction > li:nth-child(2)	
18	✓ <i>click</i>	id=lblDrugMSQuickPick	
19	✓ <i>execute script</i>	return document.getElementById("ctl00_ctl00_ctl00__ctlGeneral_pcRightPanelContent_CLW-1") != null	popUpDrugQuickPick
20	✓ <i>echo</i>	`\${popUpDrugQuickPick}`	
21	✓ <i>if</i>	`\${popUpDrugQuickPick}`==true	
22	✓ <i>echo</i>	QC PASSED (Muncul pop up add quick picks)	
23	<i>else</i>		
24	<i>echo</i>	QC FAILED (Tidak muncul pop up add quick picks)	
25	✓ <i>end</i>		

Gambar 3. 59 Skrip pemeriksaan layar pop up

Pada Gambar 3.59 di baris ke 19 merupakan skrip yang digunakan untuk melakukan pemeriksaan muncul layar *pop-up* atau tidak. Skrip tersebut menggunakan metode “document.getElementById” untuk mencari elemen dengan ID yang ditentukan, sedangkan “!=null” merupakan operator perbandingan memeriksa apakah hasil dari pemanggilan getElementById tidak sama dengan null. Jika elemen dengan ID yang ditentukan ditemukan, nilai selain null akan dikembalikan, dan pernyataan ini akan bernilai true. Sebaliknya, jika elemen tidak ditemukan, getElementById akan mengembalikan null, dan pernyataan ini akan bernilai false.

	Command	Target	Value
26	✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_cbpView_grdView_ctl02_chkls...	
27	✓ click	css=btQty	
28	✓ type	css=btQty	`\${jumlahItem}`
29	✓ echo	`\${jumlahItem}`	
30	✓ click	css=#ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_btnMPEnterPopSave > img	
31	✓ execute script	return Number(`\${jumlahItem}`).toFixed(2)	jumlahItemFormatted
32	✓ store text	xpath=/html/body/form/div[29]/div/div[6]/div[2]/div[2]/div[1]/fieldset/div[1]/table[1]/tbody/tr[2]/td/div[3]/div[2]/div/table/tbody/tr[3]/td[5]/div	jumlahItem2

Gambar 3. 60 Skrip mengonversi nilai variabel

Pada Gambar 3. 60 merupakan skrip yang digunakan untuk mengonversi nilai variabel “jumlahItem” menjadi sebuah angka dan kemudian mengonversinya menjadi string dengan dua digit di belakang koma menggunakan metode “.toFixed(2)” kemudian disimpan ke dalam variabel baru dengan nama “jumlahItemFormatted”.

	Command	Target	Value
31	✓ execute script	return Number(`\${jumlahItem}`).toFixed(2)	jumlahItemFormatted
32	✓ store text	xpath=/html/body/form/div[29]/div/div[6]/div[2]/div[2]/div[1]/fieldset/div[1]/table[1]/tbody/tr[2]/td/div[3]/div[2]/div/table/tbody/tr[3]/td[5]/div	jumlahItem2
33	✓ echo	`\${jumlahItemFormatted}`	
34	✓ echo	`\${jumlahItem2}`	
35	✓ if	`\${jumlahItemFormatted}`==`\${jumlahItem2}`	
36	✓ echo	QC PASSED (Jumlah Item Sama)	
37	else		
38	echo	QC FAILED (Jumlah Item Tidak Sama)	
39	✓ end		

Gambar 3. 61 Skrip membandingkan jumlah item transaksi klinik

Pada Gambar 3. 61 di baris ke-35 merupakan skrip yang digunakan untuk membandingkan variabel “jumlahItem2” jika sama dengan “jumlahItemFormatted” akan mencetak pesan “QC PASSED (Jumlah Item Sama)”.

Command	Target	Value
42 ✓ <i>store xpath count</i>	<code>xpath=/html/body/form/div[7]/div[1]/table/tbody/tr[1]/td[1]/table/tbody/tr[2]/td/div/div[1]/div/div[2]/div[1]/fieldset/div/div[1]/table/tbody/tr[2]/td/fieldset/ta...</code>	ceklis
43 ✓ <i>echo</i>	<code>\$(ceklis)</code>	
44 ✓ <i>if</i>	<code>\$(ceklis) == 1</code>	
45 ✓ <i>echo</i>	QC PASSED (Pilih Item Berhasil)	
46 <i>else</i>		
47 <i>echo</i>	QC FAILED (Pilih Item Berhasil)	
48 ✓ <i>end</i>		

Gambar 3. 62 Skrip pemeriksaan check item transaksi klinik

Pada Gambar 3. 62 merupakan skrip yang digunakan untuk melakukan pemriksaan ketika pengguna melakukan pilih item di layar *pop-up* pada Gambar 3. 57. Pada baris ke-42 menggunakan *store xpath count* untuk menghitung jumlah elemen pada checkbox, jika bernilai 1 maka akan mencetak pesan “QC PASSED (Pilih Item Berhasil)”.

50 ✓ <i>store xpath count</i>	<code>xpath=/html/body/form/div[7]/div[1]/table/tbody/tr[1]/td[1]/table/tbody/tr[2]/td/div/div[1]/div/div[2]/div[1]/fieldset/div/div[1]/table/tbody/tr[2]/td/fieldset/ta...</code>	ceklis2
51 ✓ <i>echo</i>	<code>\$(ceklis2)</code>	
52 ✓ <i>if</i>	<code>\$(ceklis2) == 0</code>	
53 ✓ <i>echo</i>	QC PASSED (Uncheck Berhasil)	
54 <i>else</i>		
55 <i>echo</i>	QC FAILED (Uncheck Tidak Berhasil)	
56 ✓ <i>end</i>		

## B. Melakukan pemeriksaan input linkbutton tambah data transaksi klinik.

Gambar 3. 63 Halaman transaksi klinik menggunakan linkbutton tambah data

Pada Gambar 3. 63 merupakan halaman transaksi klinik menggunakan linkbutton tambah data. Pengguna melakukan pengisian data, seperti obat, kelas tagihan, harga satuan, dan jumlah item.



Gambar 3. 64 Berhasil melakukan transaksi klinik menggunakan linkbutton tambah data

Pada Gambar 3. 64 merupakan halaman ketika berhasil melakukan transaksi klinik menggunakan linkbutton tambah data. Pada halaman ini pengguna dapat melihat rincian item dan harga.

Command	Target	Value
32 ✓ store value	id=ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPa ge_cbpMPEntryContent_plhEntry_ctDrugMS_btDr ugMSPayer	instansi
33 ✓ click	id=ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPa ge_cbpMPEntryContent_plhEntry_ctDrugMS_btDr ugMSTotal	
34 ✓ store value	id=ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPa ge_cbpMPEntryContent_plhEntry_ctDrugMS_btDr ugMSTotal	total
35 ✓ execute script	return \${total}.replace(/,/g,"")	totalFormatted

Gambar 3. 65 Skrip menghapus tanda baca koma

Pada Gambar 3. 65 di baris ke-20 melakukan store variabel dengan nama “total”, selanjutnya pada baris ke-21 menggunakan metode “.replace” yang bertujuan untuk menghapus semua koma dari nilai variabel “total”. Hal ini berguna jika nilai variabel “harga” adalah angka yang diformat dengan pemisah menggunakan koma, dan menghilangkan koma agar dapat menggunakannya dalam operasi matematika atau pemrosesan data lainnya.

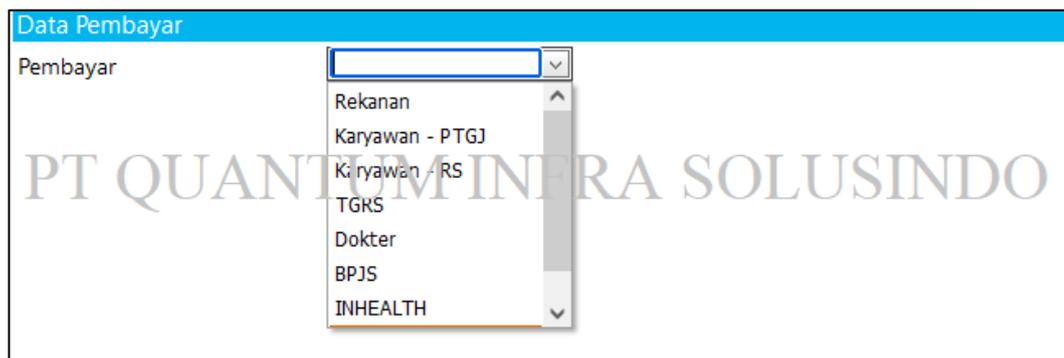
	Command	Target	Value
36	✓ <i>execute script</i>	return \${hargaFormatted}*\${digunakanDecimal}-\${diskon}	jumlahTotal
37	✓ <i>execute script</i>	return Number(\${jumlahTotal}).toFixed(2)	jumlahTotalFormatted
38	✓ <i>echo</i>	\${jumlahTotalFormatted}	
39	✓ <i>echo</i>	\${totalFormatted}	
40	✓ <i>if</i>	\${jumlahTotalFormatted}==\${totalFormatted}	
41	✓ <i>echo</i>	QC PASSED (Jumlah Total Sama)	
42	<i>else</i>		
43	<i>echo</i>	QC FAILED (Jumlah Total Tidak Sama)	
44	✓ <i>end</i>		

Gambar 3. 66 Skrip membandingkan jumlah total transaksi klinik

Pada Gambar 3. 66 di baris ke-37 menggunakan skrip yang sama pada Gambar 3. 65 untuk menghapus semua koma. Pada baris ke-40 menggunakan if statement untuk membandingkan variabel “jumlahTotalFormatted” sama dengan “totalFormatted” akan mencetak pesan “QC PASSED (Jumlah Total Sama)”.

### 3.2.5 Pembuatan command script Selenium IDE pada menu Ringkasan Tagihan dan Pembayaran

A. Membuat pendaftaran pasien baru dengan tipe pribadi dan instansi serta membuat ringkasan tagihan dan pembayaran



Gambar 3. 67 Menu pilih tipe pembayar

Pada Gambar 3. 67 merupakan menu data pembayar yang digunakan pengguna untuk menyediakan pilihan-pilihan berupa jenis pembayaran yang dapat dipilih oleh pengguna ketika melakukan

transaksi atau pembayaran dalam suatu sistem manajemen layanan kesehatan.

	Command	Target	Value
1	//	Data Ubah Diskon Pembayar	
2	//	✓ echo	Author: Billie Andhika Febrian
3	//	✓ echo	Tanggal: 25/09/2023
4	✓ store	PENYAKIT DALAM (PAGI)	namaKlinik
5	✓ store	havin	namaDokter
6	✓ store	pribadi	tipePembayar

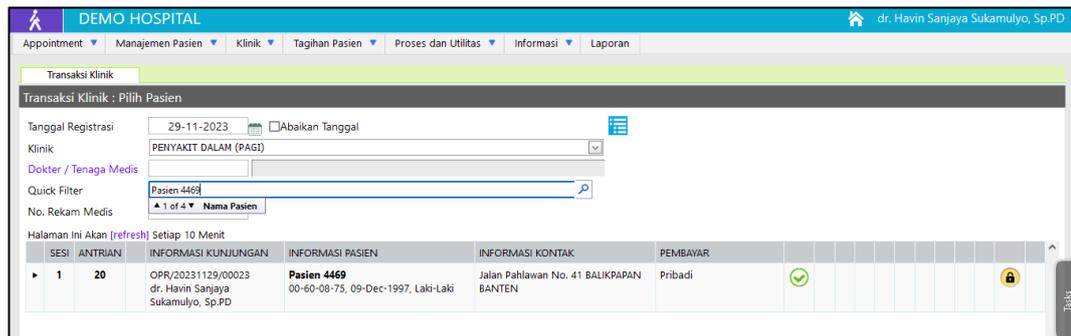
Gambar 3. 68 Store data ringkasan tagihan dan pembayaran

Pada Gambar 3. 68 merupakan data yang diperlukan untuk melakukan pemeriksaan tagihan dan pembayaran. Data tersebut disimpan ke dalam sebuah variabel yang akan digunakan untuk melakukan pemeriksaan tagihan dan pembayaran.

	Command	Target	Value
51	✓ click	id=ct100_ct100_ct100_phMPBase_phMPMain_cbpMPEntryContent_plhEntry_ibIRoom	
52	✓ click	css=#ct100_ct100_ct100_ctGeneral_pcSearchDialog_cbpSearchDialog_grdSearch tr:nth-child(7) > td:nth-child(2)	
53	✓ click	id=ct100_ct100_ct100_phMPBase_phMPMain_cbpMPEntryContent_plhEntry_cboRegistrationPayer_I	
54	✓ type	id=ct100_ct100_ct100_phMPBase_phMPMain_cbpMPEntryContent_plhEntry_cboRegistrationPayer_I	\$(tipePembayar)
55	✓ send keys	id=ct100_ct100_ct100_phMPBase_phMPMain_cbpMPEntryContent_plhEntry_cboRegistrationPayer_I	\$(KEY_ENTER)
56	✓ click	css=#ct100_ct100_ct100_phMPBase_phMPMain_bInMPEntrySave > img	
57	✓ store value	name=ct100\$ct100\$ct100\$phMPBase\$phMPMain\$cbpMPEntryContent\$plhEntry\$tdPatientName	namaPasien
58	✓ echo	\$(namaPasien)	

Gambar 3. 69 Skrip memanggil variabel tipe pembayar

Pada Gambar 3. 69 merupakan skrip yang digunakan untuk memanggil variabel “tipePembayar” dengan *value* pribadi yang sebelumnya sudah disimpan pada Gambar 3. 68.



Gambar 3. 70 Halaman filter transaksi klinik pilih pasien

Pada Gambar 3. 70 merupakan halaman yang menampilkan beberapa pasien yang sudah didaftarkan sebelumnya. Pengguna dapat menggunakan fitur “Quick Filter” untuk melakukan filter pasien dengan mengisi nama pasien tertentu yang sudah melakukan transaksi klinik.

Command	Target	Value
7 ✓ click	xpath=//div/ul/li[3]/a	30000
8 ✓ click	linkText=Transaksi Klinik	
9 ✓ click	id=ctl00_ctl00_plhMPBase_plhMPMain_cboServiceUnit_I	
10 ✓ type	id=ctl00_ctl00_plhMPBase_plhMPMain_cboServiceUnit_I	\$(namaKlinik)
11 ✓ send keys	id=ctl00_ctl00_plhMPBase_plhMPMain_cboServiceUnit_I	\$(KEY_ENTER)
12 ✓ click	name=ctl00\$ctl00\$phMPBase\$phMPMain\$ctl10	
13 ✓ type	name=ctl00\$ctl00\$phMPBase\$phMPMain\$ctl10	\$(namaPasien)
14 ✓ send keys	name=ctl00\$ctl00\$phMPBase\$phMPMain\$ctl10	\$(KEY_ENTER)
15 ✓ click	css=td:nth-child(5) > div > div:nth-child(1)	

Gambar 3. 71 Skrip memanggil variabel pasien transaksi klinik

Pada Gambar 3. 71 di baris ke-10 dan baris ke-13 merupakan skrip yang digunakan untuk memanggil variabel “namaKlinik” dan “namaPasien”. Variabel “namaPasien” akan digunakan untuk mengisi kolom “Quick Filter” pada Gambar 3. 70.

Command	Target	Value
✓ click	id=ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPageNavigationPane_plhMPPatientPageNavigationPane_ctl05_rptHeader_ctl05_llCaption	
✓ click	id=lblServiceQuickPick	
✓ click	id=ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEntypopContent_ctl03_cbpView_grdView_ctl02_chkl...	
✓ click	css=#ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_btnMPEntypopSave > img	
✓ click	css=#ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPage_btnMPEntypopSave > img	
✓ click	css=#messageBoxConfirmation.w3-btn:nth-child(1)	

Gambar 3. 72 Skrip melakukan propose transaksi klinik

Pada Gambar 3. 72 di baris ke-20 merupakan skrip yang digunakan untuk mengklik button “Propose” sebelum melanjutkan ke menu ringkasan tagihan dan pembayaran.

PELAYANAN	OBAT & ALKES	RETUR OBAT & ALKES	BARANG UMUM	RETUR BARANG UMUM	TOTAL
50 treatment 3D0001 dr. H Sukamo, Sp. D	Kelas Tagihan	Kelas Tagihan	Harga Satuan	Jumlah	HARGA CITO Diskon Instansi Pasien Total
			7,510,000.00	1.00	7,500,000.00
TOTAL INSTANSI:					Rp. 0.00
TOTAL PASIEN:					Rp. 7,500,000.00

Informasi  
 Dibuat Oleh : Power User  
 Dibuat Pada : Wednesday, 29 November 2023 15:08:56  
 Terakhir Diubah Oleh : Power User  
 Terakhir Diubah Pada : Wednesday, 29 November 2023 15:09:13

Gambar 3. 73 Halaman berhasil propose transaksi klinik

Pada Gambar 3. 73 merupakan halaman yang menampilkan bahwa transaksi klinik sudah berhasil di *propose* yang ditandai dengan cap “PROCESSED” berwarna merah.

Command	Target	Value
24 ✓ click	xpath=//div/ul/li[4]/a	
25 ✓ click	linkText=Ringkasan Tagihan dan Pembayaran	
26 ✓ click	css=#ctf00_ctf00_plhMPBase_plhMPMain_cboServiceUnit_dxic	
27 ✓ click	id=ctf00_ctf00_plhMPBase_plhMPMain_cboServiceUnit_I	
28 ✓ type	id=ctf00_ctf00_plhMPBase_plhMPMain_cboServiceUnit_I	\$(namaKlinik)
29 ✓ send keys	id=ctf00_ctf00_plhMPBase_plhMPMain_cboServiceUnit_I	\$(KEY_ENTER)
30 ✓ click	name=ctf00\$ctf00\$plhMPBase\$plhMPMain\$ctf10	
31 ✓ send keys	name=ctf00\$ctf00\$plhMPBase\$plhMPMain\$ctf10	\$(KEY_ENTER)
32 ✓ type	name=ctf00\$ctf00\$plhMPBase\$plhMPMain\$ctf10	\$(namaPasien)

Gambar 3. 74 Skrip menu tagihan dan pembayaran

Pada Gambar 3. 74 di baris ke-24 merupakan skrip yang digunakan untuk klik elemen yang berada di dalam div, ul, dan li, dan kemudian mengandung tautan (a). Hal ini akan memilih elemen keempat dalam daftar yaitu, menu “Tagihan Pasien”.



Gambar 3. 75 Halaman utama filter pasien ringkasan tagihan dan pembayaran

Pada Gambar 3. 75 merupakan halaman yang menampilkan beberapa pasien yang sudah melakukan transaksi klinik. Pengguna dapat menggunakan fitur “Quick Filter” untuk melakukan filter pasien dengan mengisi nama pasien tertentu yang akan dilanjutkan ke halaman ringkasan tagihan dan pembayaran.



Gambar 3. 76 Layar pop-up konfirmasi lock transaksi

Pada Gambar 3. 76 merupakan layar *pop-up* untuk melakukan konfirmasi penguncian transaksi yang sudah dilakukan. *Lock* transaksi digunakan untuk mencegah perubahan yang tak terduga pada suatu transaksi yang telah dilakukan. Ada juga manfaat lain, seperti memberikan tingkat keamanan tambahan dengan mencegah manipulasi atau modifikasi pada tahap transaksi tertentu.



Gambar 3. 77 Halaman pilih transaksi untuk pembuatan tagihan

Pada Gambar 3. 77 merupakan halaman yang digunakan untuk memilih transaksi mana yang akan dilakukan untuk proses selanjutnya, yaitu pembuatan tagihan. Setelah pengguna melakukan ceklis pada transaksi yang dipilih, kemudian dilanjutkan dengan mengklik button “Proses” pada pojok kiri atas.

Command	Target	Value
32 ✓ type	name=ctl00\$ctl00\$plhMPBase\$plhMPMain\$ctl10	\$(namaPasien)
33 ✓ click	css=td:nth-child(6) > div > span	
34 ✓ click	id=ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPageNavigationPane_plhMPPatientPageNavigationPane_ctfToolbar_rptMenuChild_ctl03_jiCaption	
35 ✓ click	id=imgLock	
36 ✓ click	css=.btnOK	
37 ✓ pause	3000	
38 ✓ click	id=ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPage_cbpMPEntryContent_phEntry_cbpView_lwView_ctr0_chkIsSelected	
39 ✓ click	css=#ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPage_phCustomButtonToolbar_btnProcessGenerateBill > img	

Gambar 3. 78 Skrip lock transaksi dan pilih transkasksi

Pada Gambar 3.78 di baris ke-35 merupakan perintah yang digunakan untuk mengklik *lock* transaksi dengan id elemen “imgLock”. Pada baris ke-38 melakukan perintah untuk mengklik pilihan transaksi dan dilanjutkan dengan perintah *click* “btnProcessGenerateBill” pada baris ke-39.

Command	Target	Value
44 ✓ click	id=ctl00_ctl00_ctl00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_cboDiscountReason_1	
45 ✓ click	id=ctl00_ctl00_ctl00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_cboDiscountReason_DDD_L_LBI2T0	
46 ✓ click	id=ctl00_ctl00_ctl00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_chkPayerDiscountPercent	
47 ✓ click	id=ctl00_ctl00_ctl00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_btPayerEntryDiscountPercent	
48 ✓ type	id=ctl00_ctl00_ctl00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_btPayerEntryDiscountPercent	10
49 ✓ store value	id=ctl00_ctl00_ctl00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_btPayerEntryDiscountAmount	jumlahDiskon

Gambar 3. 79 Skrip menyimpan jumlah diskon ringkasan tagihan dan pembayaran

Pada Gambar 3. 79 merupakan skrip untuk menyimpan nilai diskon dengan value 10 dengan variabel yang diberi nama “jumlahDiskon”.

	Command	Target	Value
50	✓ <i>click</i>	id=btnSave	
51	✓ <i>click</i>	css=#ctl00_ctl00_ctlGeneral_pcRightPanelContent_HCB-1 > img	
52	✓ <i>store text</i>	xpath=/html/body/form/div[29]/div/div[6]/div[2]/div[2]/div[1]/fieldset/div/table/tbody/tr[3]/td[9]/div	jumlahDiskon2
53	✓ <i>execute script</i>	return \${jumlahDiskon2}.replace(/,/g,"")	diskon2Formatted
54	✓ <i>store text</i>	xpath=/html/body/form/div[29]/div/div[6]/div[2]/div[2]/div[1]/fieldset/div/table/tbody/tr[3]/td[10]/div	total
55	✓ <i>execute script</i>	return \${total}.replace(/,/g,"")	totalFormatted
56	✓ <i>store text</i>	xpath=/html/body/form/div[29]/div/div[6]/div[2]/div[2]/div[1]/fieldset/div/table/tbody/tr[3]/td[8]/div	tagihan
57	✓ <i>execute script</i>	return \${tagihan}.replace(/,/g,"")	tagihanFormatted
58	✓ <i>execute script</i>	return \${tagihanFormatted}-\${diskon2Formatted}	total2

Gambar 3. 80 Skrip melakukan format jumlah tagihan

Pada Gambar 3.80 merupakan skrip yang digunakan untuk melakukan perubahan format dengan menghilangkan tanda baca koma. Kemudian akan disimpan ke dalam sebuah variabel baru.

	Command	Target	Value
59	✓ <i>execute script</i>	return \${total2}+'.00'	total2Formatted
60	✓ <i>echo</i>	\${jumlahDiskon}	
61	✓ <i>echo</i>	\${jumlahDiskon2}	
62	✓ <i>if</i>	\${jumlahDiskon}==\${jumlahDiskon2}	
63	✓ <i>echo</i>	QC PASSED (Jumlah Diskon Sama)	
64	<i>else</i>		
65	<i>echo</i>	QC FAILED (Jumlah Diskon Tidak Sama)	
66	✓ <i>end</i>		

Gambar 3. 81 Skrip membandingkan jumlah diskon

Pada Gambar 3.81 di baris ke-59 merupakan skrip yang digunakan untuk menggabungkan variabel “total2” dengan string ‘.00’ dan mengembalikan hasilnya dan disimpan ke sebuah variabel baru dengan nama “total2Formatted”. Pada baris ke-62 menggunakan *if statement* untuk membandingkan jumlah diskon, jika kondisi terpenuhi maka QC PASSED.

Command	Target	Value
63 ✓ echo	QC PASSED (Jumlah Diskon Sama)	
64 else		
65 echo	QC FAILED (Jumlah Diskon Tidak Sama)	
66 ✓ end		
67 ✓ echo	`\${totalFormatted}`	
68 ✓ echo	`\${total2Formatted}`	
69 ✓ if	`\${totalFormatted}`==`\${total2Formatted}`	
70 ✓ echo	QC PASSED (Jumlah Total Sama)	
71 else		
72 echo	QC FAILED (Jumlah Total Tidak Sama)	
73 ✓ end		

Gambar 3. 82 Skrip membandingkan jumlah total

Pada Gambar 3. 82 merupakan skrip yang digunakan untuk membandingkan variabel “totalFormatted” sama dengan “total2Formatted” yang sebelumnya sudah diformat dengan menggabungkan string, dan akan mencetak “QC PASSED (Jumlah Total Sama)”.

### 3.2.6 Pembuatan command script Selenium IDE pada tab Outpatient Soap menu Asesmen Umum

A. Melakukan *testing* pada *fieldset* status pasien.

- Cek anamnesa

Gambar 3. 83 Halaman anamnesa

Pada Gambar 3. 83 merupakan halaman yang digunakan untuk mengisi anamnesa awal pasien yang bersifat mandatory, ditandai dengan adanya tanda (\*). Anamnesa merupakan hal yang penting dalam menyusun diagnosa, merencanakan pengobatan, dan memberikan pelayanan kesehatan yang tepat.

Command	Target	Value
20 ✓ <i>click</i>	css=#ct100_ct100_ct100_plhMPBase_plhMPPatientPage_plhCustomButtonToolbar_btnPatientStatusSave > img	
21 ✓ <i>store xpath count</i>	xpath=/html/body/form/div[10]/div[1]/div/div	popUpWarning
22 ✓ <i>echo</i>	\$(popUpWarning)	
23 ✓ <i>if</i>	\$(popUpWarning)==1	
24 ✓ <i>echo</i>	QC PASSED (Anamnesa tidak diisi)	
25 <i>else</i>		
26 <i>echo</i>	QC FAILED (Anamnesa tidak diisi)	
27 ✓ <i>end</i>		

Gambar 3. 84 Skrip pop up warning anmnesa

Pada Gambar 3. 84 merupakan skrip yang digunakan untuk memeriksa layar pop-up warning yang akan muncul ketika pengguna tidak mengisi anamnesa. Pemeriksaan dilakukan dengan perintah *store xpath count* untuk menghitung jumlah elemen pada pop-up.

Command	Target	Value
35 ✓ <i>click</i>	id=ct100_ct100_ct100_plhMPBase_plhMPPatientPage_plhList_txtChiefComplaint	
36 ✓ <i>type</i>	id=ct100_ct100_ct100_plhMPBase_plhMPPatientPage_plhList_txtChiefComplaint	Demam
37 ✓ <i>execute script</i>	return document.getElementById("ct100_ct100_ct100_plhMPBase_plhMPPatientPage_plhList_txtChiefComplaint").value.length > 0;	chiefComplaint
38 ✓ <i>if</i>	\$(chiefComplaint)==true	
39 ✓ <i>echo</i>	QC PASSED (Anemnesa berhasil diisi)	
40 <i>else</i>		
41 <i>echo</i>	QC FAILED (Anamnesa tidak berhasil diisi)	
42 ✓ <i>end</i>		

Gambar 3. 85 Skrip memeriksa anamnesa sudah diisi

Pada gambar 3. 85 merupakan skrip yang digunakan untuk memeriksa *pop-up* sudah terisi atau belum menggunakan metode “document.getElementById()” dan properti “.length” untuk mengukur

panjang atau jumlah karakter dalam string tersebut dan mengembalikan nilai true atau false.

- Riwayat Alergi

Allergen Name	Finding Source	Since	Severity	Reaction
paracetamol	Patient	29-Nov-2023	Unknown	mual

Gambar 3. 86 Halaman riwayat alergi

Pada Gambar 3. 86 merupakan halaman yang digunakan untuk melakukan pengisian riwayat alergi pasien, seperti tipe alergi, nama, dan reaksi.

Command	Target	Value
✓ click	css=.h4collapsed:nth-child(7)	
✓ click	id=ctl00_ctl00_ctl00_phlMPBase_phlMPPatientPage_plhList_cboAllergenType_I	
✓ click	id=ctl00_ctl00_ctl00_phlMPBase_phlMPPatientPage_plhList_cboAllergenType_DDD_L_LB110	
✓ click	id=ctl00_ctl00_ctl00_phlMPBase_phlMPPatientPage_plhList_btAllergenName	
✓ type	id=ctl00_ctl00_ctl00_phlMPBase_phlMPPatientPage_plhList_btAllergenName	paracetamol
✓ click	id=ctl00_ctl00_ctl00_phlMPBase_phlMPPatientPage_plhList_btReaction	
✓ type	id=ctl00_ctl00_ctl00_phlMPBase_phlMPPatientPage_plhList_btReaction	mual
✓ click	css=.btnApplyAllergy	

Gambar 3. 87 Skrip mengisi riwayat alergi

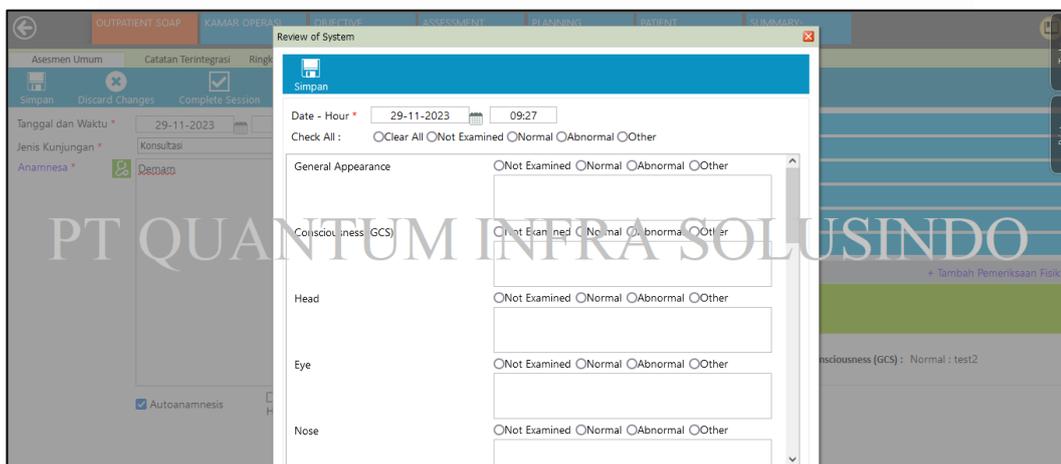
Pada Gambar 3. 87 merupakan skrip yang digunakan untuk mengisi riwayat alergi. Pada baris ke-30 dan baris ke-32 menggunakan perintah *type* untuk mengisi nilai nama dan reaksi alergi.

Command	Target	Value
35 ✓ <i>execute script</i>	<code>return document.evaluate("//html/body/form/div[29]/div[5]/div[2]/fieldset/table/body/tr[2]/div[4]/div/table/body/tr[2]/td/div[1]/div/div/table/body/tr[2]", document...</code>	applyAlergi
36 ✓ <i>if</i>	<code>\$(applyAlergi)!=true</code>	
37 ✓ <i>echo</i>	QC PASSED (Apply riwayat alergi berhasil)	
38 <i>else</i>		
39 <i>echo</i>	QC FAILED (Apply riwayat alergi tidak berhasil)	
40 ✓ <i>end</i>		

Gambar 3. 88 Skrip memeriksa status apply riwayat alergi

Pada Gambar 3. 88 di baris ke-35 merupakan skrip menggunakan metode “document.evaluate ()” untuk mengevaluasi ekspresi XPath tertentu pada dokumen HTML. Setelah evaluasi ekspresi XPath, memeriksa apakah hasilnya bukan null menggunakan pernyataan “!=”. Jika singleNodeValue tidak null, maka ekspresi XPath tersebut berhasil menemukan elemen yang sesuai, dan pernyataan tersebut mengembalikan true. Sebaliknya, jika singleNodeValue null, maka pernyataan tersebut mengembalikan false.

- Cek pemeriksaan fisik



Gambar 3. 89 Layar pop-up pengisian pemeriksaan fisik

Pada Gambar 3. 89 merupakan layar *pop-up* pengisian pemeriksaan fisik yang berupa *radio button*. Pengguna dapat memilih salah satu atau

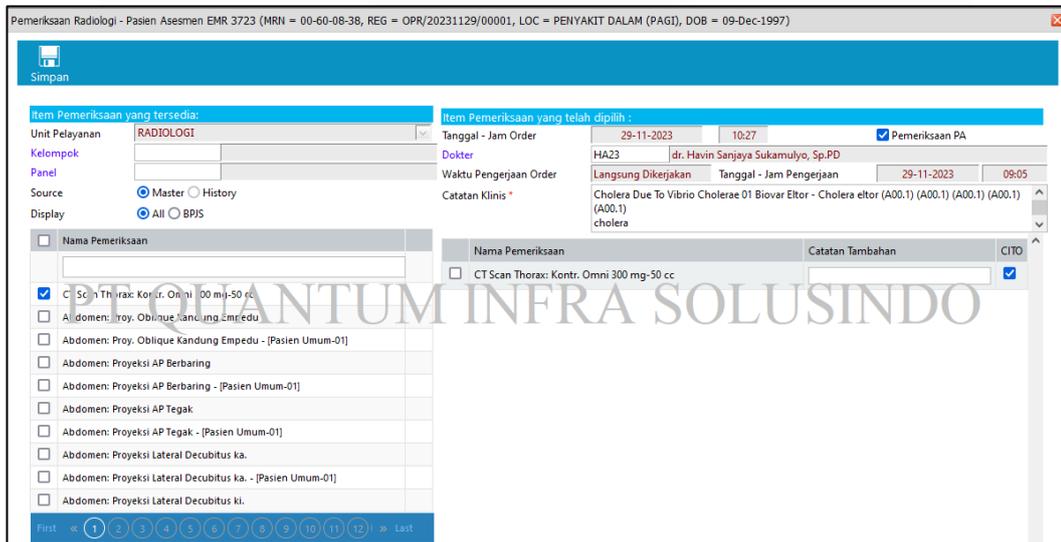
lebih dan juga dapat menggunakan fungsi check all pada bagian atas halaman.

Command	Target	Value
13 ✓ <i>click</i>	id=ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_rblCheckAll_1	
14 ✓ <i>execute script</i>	return document.getElementById('ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_...	notExamined1
15 ✓ <i>if</i>	\$(notExamined1)==true	
16 ✓ <i>echo</i>	QC PASSED (Radio button Not Examined baris ke-1 berhasil diklik)	
17 <i>else</i>		
18 <i>echo</i>	QC FAILED (Radio button Not Examined baris ke-1 tidak berhasil diklik)	
19 ✓ <i>end</i>		

Gambar 3. 90 Skrip pilih pemeriksaan fisik

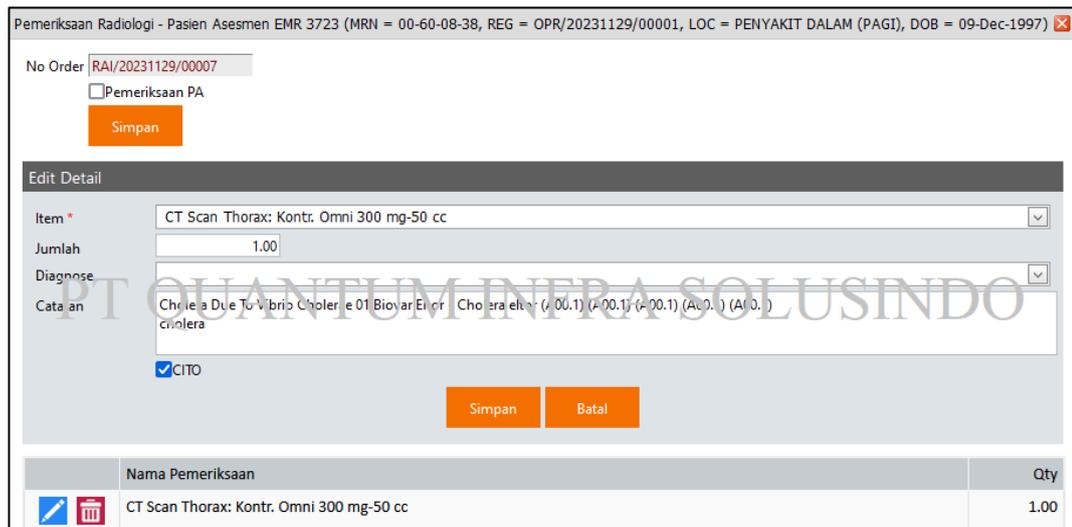
Pada Gambar 3. 90 di baris ke-14 merupakan skrip yang menggunakan metode “document.evaluate” untuk mencari elemen menggunakan ekspresi Xpath dan mengembalikan nilai *true* atau *false*.

- Cek order radiologi



Gambar 3. 91 Halaman cek pemeriksaan radiologi

Pada Gambar 3. 91 merupakan halaman yang digunakan untuk memilih jenis pemeriksaan yang dilakukan. Pengguna dapat melakukan *check* dan *uncheck* menambahkan dan menghapus pemeriksaan.



Gambar 3. 92 Halaman edit pemeriksaan radiologi

Pada Gambar 3. 92 merupakan halaman edit pemeriksaan radiologi yang digunakan untuk melakukan perubahan data yang sudah disimpan sebelumnya.

Command	Target	Value
38 ✓ click	id=ct00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_chkisPathologicalAnatomyTest	
39 ✓ click	id=ct00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_cbpPopup_grdView_ctl02_chkl...	
40 ✓ click	css=txtDetailRemarks	
41 ✓ type	css=txtDetailRemarks	Test catatan tambahan order list radiologi
42 ✓ store value	xpath=/html/body/form/div[7]/div[1]/table/tbody/tr[1]/td[1]/table/tbody/tr[2]/td/div/div[1]/div/div[2]/div[1]/fields/div/div/table/tbody/tr[2]/div/fields/div[1]/table...	txtRemarksRadList
43 ✓ execute script	return document.getElementById('ct00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl04_cbpMPEnterPopContent_ctl03_...	paRadList
44 ✓ click	css=chkCITO	
45 ✓ execute script	return document.querySelector('input[type="checkbox"][chkCITO]:checked')	citoRadList

Gambar 3. 93 Skrip menyimpan nilai ceklis PA dan CITO

Pada Gambar 3. 93 merupakan skrip yang digunakan untuk menyimpan nilai true ketika pengguna melakukan ceklis pada checkbox “Pemeriksaan PA” dan “CITO”

Command	Target	Value
50 ✓ <i>execute script</i>	return document.getElementById('ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_chkIsPathologicalAnatomyTest').c...	paRadList2
51 ✓ <i>if</i>	`\${paRadList2}`==`\${paRadList2}`	
52 <i>echo</i>	QC PASSED (Ceklis checkbox pemeriksaan PA radiologi tambah order berhasil)	
53 ✓ <i>else</i>		
54 ✓ <i>echo</i>	QC FAILED (Ceklis checkbox pemeriksaan PA radiologi tambah order tidak berhasil)	
55 ✓ <i>end</i>		

Gambar 3. 94 Skrip memeriksa ceklis pemeriksaan PA

Pada Gambar 3. 94 merupakan skrip yang digunakan untuk memeriksa nilai pada elemen *checkbox* “Pemeriksaan PA” pada edit pemeriksaan radiologi. Sebelumnya pengguna sudah melakukan ceklis pada *checkbox* “Pemeriksaan PA”, namun setelah di edit pemeriksaan radiologi, ceklis tersebut hilang sehingga menghasilkan nilai *false* pada baris ke-50. Pada kasus ini, skrip mencetak pesan “QC FAILED (Ceklis checkbox pemeriksaan PA radiologi tambah order tidak berhasil)”.

Command	Target	Value
72 ✓ <i>execute script</i>	return document.getElementById('ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_cbpRightPanelContent_ctl03_chkIsCITO').checked	citoRadList2
73 ✓ <i>if</i>	`\${citoRadList2}`==`\${citoRadList2}`	
74 ✓ <i>echo</i>	QC PASSED (Ceklis checkbox CITO radiologi tambah order berhasil)	
75 <i>else</i>		
76 <i>echo</i>	QC FAILED (Ceklis checkbox CITO radiologi tambah order tidak berhasil)	
77 ✓ <i>end</i>		
78 ✓ <i>click</i>	css=#ctl00_ctl00_ctl00_ctlGeneral_pcRightPanelContent_HCB-1 > img	

Gambar 3. 95 Skrip memeriksa ceklis CITO

Pada Gambar ke 3. 95 merupakan skrip yang digunakan untuk menyimpan nilai true elemen CITO pada menu edit pemeriksaan radiologi, kemudian membandingkan dengan nilai CITO pada saat pengguna awal pengguna melakukan tambah data. Jika kondisi tersebut terpenuhi, maka akan mencetak pesan “QC PASSED (Ceklis checkbox CITO radiologi tambah data berhasil)”.

- Cek catatan kunjungan berikutnya

Gambar 3. 96 Halaman jadwal kunjungan berikutnya

Pada Gambar 3. 96 merupakan halaman yang memberikan informasi untuk melihat kapan jadwal kunjungan berikutnya dapat dilakukan. Halaman ini berisi informasi tanggal dan jam disposisi, keadaan keluar, dan catatan perjanjian selanjutnya.

Command	Target	Value
25 ✓ click	css=h4collapsed:nth-child(1)	
26 ✓ click	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_cboPatientOutcome_1	
27 ✓ click	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_cboPatientOutcome_DDD_L_LBI0T0	
28 ✓ click	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_cboFollowupVisitType_1	
29 ✓ click	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_cboFollowupVisitType_DDD_L_LBI0T0	
30 ✓ click	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_txtFollowupVisitDate	
31 ✓ type	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_txtFollowupVisitDate	\${followUpVisit}
32 ✓ store value	id=ct100_ct100_ct100_phlMPBase_phlMPPatientPage_phlList_txtFollowupVisitDate	visitDate

Gambar 3. 97 Skrip memanggil variabel jadwal kujungan berikutnya sebelum disimpan

Pada Gambar 3. 97 di baris ke-31 merupakan skrip yang digunakan untuk memanggil variabel “followUpVisit” yang sebelumnya sudah disimpan dan melanjutkan untuk menyimpan nilai tanggal kunjungan berikutnya dengan variabel yang diberi nama “visitDate”.

	Command	Target	Value
37	✓ <i>click</i>	css=.h4collapsed:nth-child(1)	
38	✓ <i>pause</i>	3000	
39	✓ <i>store text</i>	id=ct00_ct00_ct00_plhMPBase_plhMPPatientPa ge_plhList_btAppointmentInfo	resultDate
40	✓ <i>echo</i>	\${resultDate}	
41	✓ <i>execute script</i>	return \${resultDate}.match(/(\d{2}-[A-Za-z]{3}-\d {4})/)[0]	resultDateFormatted
42	✓ <i>echo</i>	\${resultDateFormatted}	

Gambar 3. 98 Skrip menyimpan variabel jadwal kunjungan berikutnya setelah disimpan

Pada Gambar 3. 98 merupakan skrip yang digunakan untuk menyimpan variabel “resultDate” setelah melakukan simpan data. Pada baris ke-41 menggunakan metode “.match” untuk mencocokkan ekspresi reguler tertentu.

	Command	Target	Value
43	✓ <i>echo</i>	\${visitDate}	
44	✓ <i>execute script</i>	return \${visitDate}.split("-")	parts
45	✓ <i>echo</i>	\${parts[1]}	
46	✓ <i>store</i>	Jan-Feb-Mar-Apr-May-Jun-Jul-Aug-Sep-Oct-Nov-De c	monthNames
47	✓ <i>execute script</i>	return \${monthNames}.split("-")	monthFormatted
48	✓ <i>execute script</i>	return \${monthFormatted}\${parts[1]} - 1	matchedMonth
49	✓ <i>execute script</i>	return \${parts[0]} + '-' + \${matchedMonth} + '-' + \${pa rts[2]}	visitDateFormatted

Gambar 3. 99 Skrip array mengubah format tanggal

Pada Gambar 3. 99 merupakan skrip yang menggunakan array untuk mengubah format angka bulan menjadi format dengan tiga huruf bulan. Pada baris ke-43 melakukan pembagian string variabel “visitDate” berisi tiga elemen, yaitu tanggal, bulan, dan tahun dan disimpan ke variabel “parts”. Pada baris ke-46 mendefinisikan string yang berisi representasi tiga huruf dari bulan-bulan dalam satu string yang diberi nama “monthNames. Pada baris ke-47 mencoba membagi string “monthNames” menjadi array dengan pemisah "-", dan hasilnya disimpan dalam variabel “monthFormatted”. Pada baris ke-48 mengambil elemen array yang sesuai dengan bulan yang diambil dari array “parts”. Angka bulan (sebagai indeks) dikurangi 1 karena indeks array dimulai dari 0. Pada baris ke-49 menggabungkan elemen-elemen

dari array parts dengan representasi tiga huruf dari bulan yang sesuai “matchedMonth” untuk membentuk format tanggal baru.

	Command	Target	Value
49	✓ <i>execute script</i>	return \${parts[0]} + '-' + \${matchedMonth} + '-' + \${parts[2]}	visitDateFormatted
50	✓ <i>echo</i>	`\${visitDateFormatted}	
51	✓ <i>echo</i>	`\${resultDateFormatted}	
52	✓ <i>if</i>	`\${visitDateFormatted}`==`\${resultDateFormatted}	
53	✓ <i>echo</i>	QC PASSED (Tanggal perjanjian selanjutnya sesuai)	
54	<i>else</i>		
55	<i>echo</i>	QC FAILED (Tanggal perjanjian selanjutnya tidak sesuai)	
56	✓ <i>end</i>		
57	✓ <i>click</i>	css=#ct00_ctl00_ctl00_plhMPBase_plhMPPatientPage_plhCustomButtonToolbar_btnPatientStatusSave > img	

Gambar 3. 100 Skrip membandingkan tanggal perjanjian sebelum dan sesudah disimpan

Pada Gambar 3. 100 merupakan skrip yang digunakan untuk membandingkan variabel “visitDateFormatted” sama dengan “visitDateFormatted” dan mencetak pesan “QC PASSED (Tanggal perjanjian selanjutnya sesuai)”.

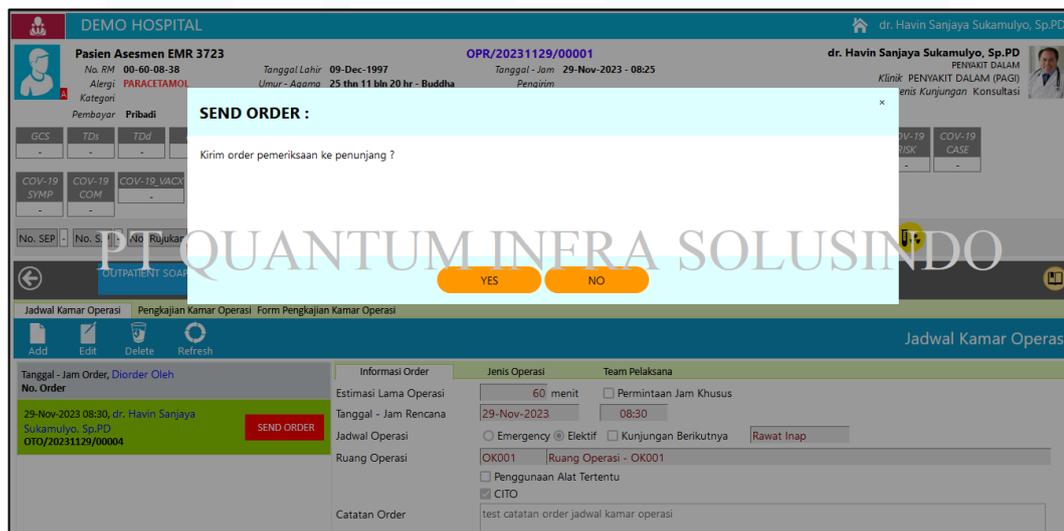
### 3.2.7 Pembuatan command script Selenium IDE pada tab Planning menu Online Prescription

A. Melakukan *testing* pada *button* yang terdapat pada *Add Online Prescription*, seperti *Quick Picks*, *Quick Picks Template*, dan *Compound*.

	Command	Target	Value
10	✓ <i>click</i>	css=btnOK	
11	✓ <i>execute script</i>	return document.evaluate("//html/body/form/div[29]/div[5]/div[4]/div[1]/div/div/table/tbody/tr[2]", document, null, XPathResult.FIRST_ORDERED_NODE_TYPE...	applyCompound
12	✓ <i>if</i>	`\${applyCompound}`==true	
13	✓ <i>echo</i>	QC PASSED (Apply compound berhasil)	
14	<i>else</i>		
15	<i>echo</i>	QC FAILED (Apply compound tidak berhasil)	
16	✓ <i>end</i>		

Gambar 3. 101 Skrip add online prescription

Pada Gambar 3. 101 merupakan skrip untuk memeriksa apakah tambah data online prescription berhasil atau tidak menggunakan metode “document.evaluate()” untuk mengevaluasi ekspresi XPath tertentu dalam halaman HTML, kemudian menggunakan pernyataan ketidaksetaraan “!=”. Jika singleNodeValue tidak null, maka ekspresi XPath tersebut berhasil menemukan elemen yang sesuai, dan pernyataan tersebut mengembalikan true. Sebaliknya, jika singleNodeValue null, maka pernyataan tersebut mengembalikan false.



Gambar 3. 102 Layar pop up send order online prescription

Pada Gambar 3. 102 merupakan halaman yang menampilkan layar *pop-up* untuk melakukan konfirmasi kirim order pemeriksaan ke penunjang medis setelah melakukan *add online prescription* pada Gambar 3. 101.

	Command	Target	Value
16	✓ end		
17	✓ click	css=#ctl00_ctl00_ctl00_plhMPBase_plhMPPatientPage_plhCustomButtonToolBar_btnSendOrder > img	
18	✓ store xpath count	xpath=/html/body/form/div[15]/div	popUpSendOrder
19	✓ if	\${popUpSendOrder}==1	
20	✓ echo	QC PASSED (Ada pop up send order ke farmasi)	
21	else		
22	echo	QC FAILED (Tidak ada pop up send order ke farmasi)	
23	✓ end		

Gambar 3. 103 Skrip memeriksa layar pop-up send order online prescription

Pada Gambar 3. 103 merupakan skrip yang digunakan untuk memeriksa layar *pop-up* konfirmasi kirim order ke penunjang medis. Pada baris ke-18 menggunakan perintah *store xpath count* untuk melakukan penghitungan jumlah elemen menggunakan XPath pada halaman web.

	Command	Target	Value
26	✓ click	id=ctl00_ctl00_ctl00_plhMPBase_rptMenuHeader_ctl09_jiCaption	
27	✓ store attribute	xpath=/html/body/form/div[29]/div[5]/table/body/tr/td[1]/div/div[1]/div/div/table/body/tr[2]/td[4]/table/body/tr/td[3]/div/input@value	buttonText
28	✓ if	\${buttonText}=="REOPEN"	
29	✓ echo	QC PASSED (Send order online prescription berhasil)	
30	else		
31	echo	QC FAILED (Send order online prescription tidak berhasil)	
32	✓ end		
33	✓ click	css=.divOpenRightPanelContent:nth-child(3) .divRightPanelBackground	

Gambar 3. 104 Skrip memeriksa status button

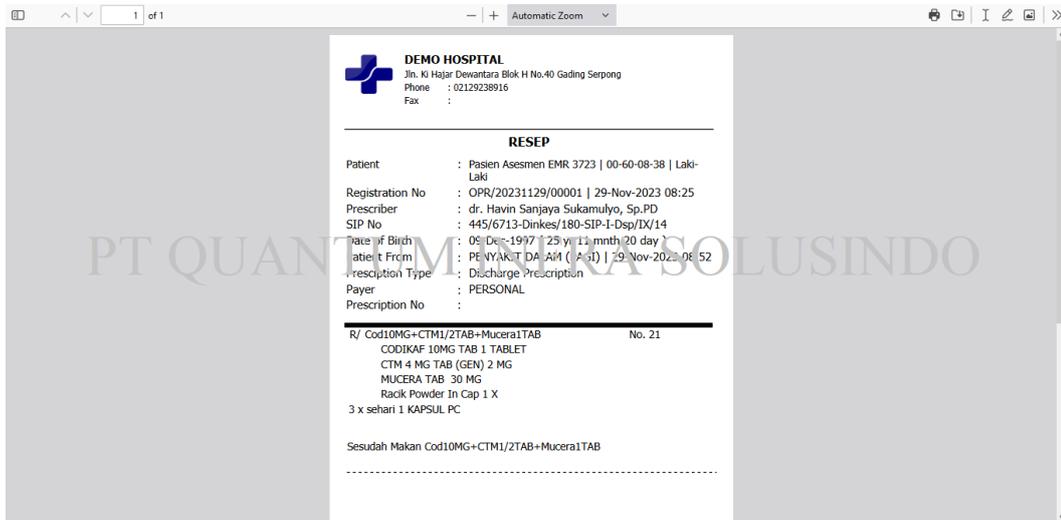
Pada Gambar 3. 104 merupakan skrip yang digunakan untuk memeriksa status button yang sebelumnya bernilai "OPEN" menjadi "REOPEN" setelah dilakukan *send order*. Pada baris ke-27 menggunakan perintah *store attribute* untuk mendapatkan nilai dari atribut "value" pada elemen input dan disimpan ke dalam variabel dengan nama "buttonText". Pada baris ke-28 menggunakan if statement untuk membandingkan nilai variabel "buttonText" sama dengan

“REOPEN” dan mencetak pesan “QC PASSED (Send order online prescription berhasil)”.

	Command	Target	Value
33	✓ click	css= divOpenRightPanelContent:nth-child(3) .divRightPanelBackground	
34	✓ click	name=rboPrint	
35	✓ click	id=imgRightPanelPrint	
36	✓ store window handle	print	
37	✓ select window	handle=\${print}	
38	✓ execute script	return document.readyState;	pageLoadStatus
39	✓ echo	\${pageLoadStatus}	
40	✓ if	\${pageLoadStatus}=="complete"	
41	✓ echo	QC PASSED (Muncul window handle print)	
42	else		
43	echo	QC FAILED (Tidak muncul window handle print)	
44	✓ end		

Gambar 3. 105 Skrip memeriksa status window

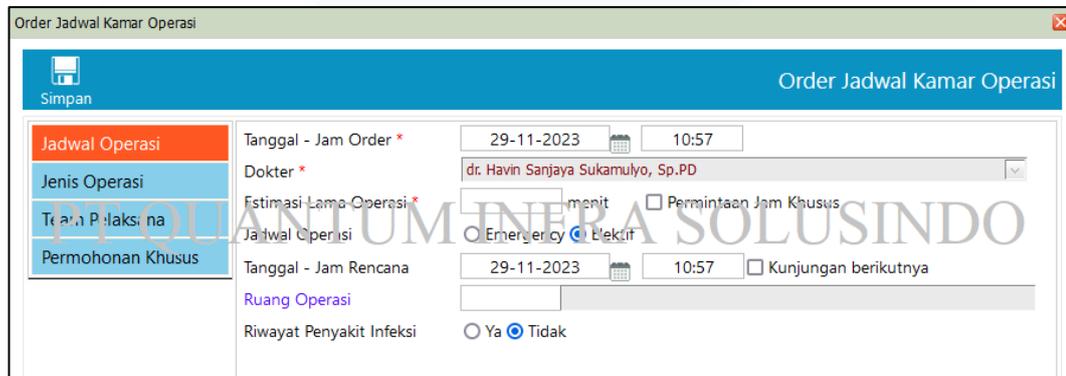
Pada Gambar 3. 105 di baris ke-36 merupakan skrip yang digunakan untuk menyimpan *window handle* dengan “print”. Pada baris ke-38 menggunakan metode “document.readyState” yang digunakan untuk memeriksa status kesiapan halaman web dan akan menghasilkan 3 nilai, seperti *loading*, yaitu halaman masih dalam proses pengunduhan, *interactive*, yaitu pengunduhan telah selesai dan dokumen sedang dalam proses pemuatan, dan *complete*, yaitu halaman telah sepenuhnya dimuat. Pada baris ke-40 menggunakan if statement dengan pernyataan sama dengan “complete” yang digunakan untuk memeriksa halaman telah sepenuhnya dimuat dan akan mencetak pesan “QC PASSED (Muncul window handle print)”. Pada Gambar 3. 106 dibawah ini merupakan halaman window handle print yang sepenuhnya berhasil dimuat. Pada halaman ini menampilkan rincian *online prescription* yang sudah dilakukan.



Gambar 3. 106 Halaman window handle print

### 3.2.8 Pembuatan command script Selenium IDE pada menu Kamar Operasi

A. Melakukan testing pada menu Jadwal Kamar Operasi.



Gambar 3. 107 Halaman jadwal kamar operasi

Pada gambar 3. 107 merupakan halaman pengisian jadwal kamar operasi, seperti dokter, estimasi lama operasi, jadwal operasi, dan ruang operasi.

	Command	Target	Value
137	✓ <i>click</i>	css=.btnPropose	
138	✓ <i>click</i>	css=#messageBoxConfirmation .w3-btn:nth-child (1)	
139	✓ <i>store attribute</i>	xpath=/html/body/form/div[29]/div[5]/table/tbody/tr/td [1]/div/div[1]/div/div/table/tbody/tr[2]/td[2]/table/tbody/t r/td[3]/div/input@value	sendOrderOperasi
140	✓ <i>echo</i>	\$(sendOrderOperasi)	
141	✓ <i>if</i>	\$(sendOrderOperasi)=="REOPEN"	
142	✓ <i>echo</i>	QC PASSED (Send order jadwal kamar operasi ber hasil)	
143	<i>else</i>		
144	<i>echo</i>	QC FAILED (Send order jadwal kamar operasi tida k berhasil)	
145	✓ <i>end</i>		

Gambar 3. 108 Skrip memeriksa status button send order kamar operasi

Pada Gambar 3. 108 merupakan skrip yang digunakan untuk memeriksa status button yang sebelumnya bernilai “OPEN” menjadi “REOPEN” setelah dilakukan send order menggunakan perintah *store attribute*.

#### B. Melakukan testing pada menu Pengkajian Kamar Operasi.

Gambar 3. 109 Halaman pengkajian kamar operasi

Pada Gambar 3. 109 merupakan halaman utama menu “Pengkajian Kamar Operasi”, pengguna atau tenaga medis diwajibkan mengisi kolom mandatory yang ditandai dengan tanda (\*). Pada kolom lama operasi akan otomatis terhitung *by system* dari hasil pengurangan jam mulai operasi dengan jam selesai operasi.

	Command	Target	Value
12	✓ store value	id=ctf00_ctf00_ctf00_plhMPBase_plhMPPatientPage_phList_btEndTime	endTimeSurgery
13	✓ store value	id=ctf00_ctf00_ctf00_plhMPBase_plhMPPatientPage_phList_btStartTime	startTimeSurgery
14	✓ execute script	return \${startTimeSurgery}.split(":")[0] * 60	startHours
15	✓ execute script	return \${startTimeSurgery}.split(":")[1]	startMinutes
16	✓ execute script	return \${endTimeSurgery}.split(":")[0] * 60	endHours
17	✓ execute script	return \${endTimeSurgery}.split(":")[1]	endMinutes
18	✓ execute script	return Number (\${endHours}) + Number (\${endMinutes})	endResult
19	✓ execute script	return Number (\${startHours}) + Number (\${startMinutes})	startResult

Gambar 3. 110 Skrip menghitung lama operasi

Pada Gambar 3. 110 di baris ke-12 dan baris ke-13 menggunakan perintah *store value* untuk menyimpan nilai jam selesai operasi dengan nama “endTimeSurgery” dan “startTimeSurgery” untuk menyimpan nilai jam mulai operasi. Pada baris ke-14 dan baris ke-16 menggunakan metode “split ()” mengambil elemen pertama dari array string “startTimeSurgery” sebelum titik (.), yaitu format jam yang kemudian mengonversinya menjadi nilai numerik dan mengalikannya dengan 60 untuk mengubah satuan jam ke menit. Pada baris ke-15 dan baris ke-17 mengambil elemen kedua dari array string “startTimeSurgery”, yaitu format menit. Pada baris ke-18 merupakan operasi hitung jam selesai operasi dengan mengambil nilai format jam yang sudah diubah ke dalam satuan menit dalam variabel “endResult” dan jam mulai operasi dalam satuan menit dalam variabel “startResult” pada baris ke-19.

	Command	Target	Value
20	✓ execute script	return \${endResult} - \${startResult}	resultDurations
21	✓ echo	\${resultDurations}	
22	✓ echo	\${durations}	
23	✓ if	\${resultDurations}==\${durations}	
24	✓ echo	QC PASSED (Durasi operasi sesuai)	
25	else		
26	echo	QC FAILED (Durasi operasi tidak sesuai)	
27	✓ end		

Gambar 3. 111 Skrip membandingkan durasi operasi

Pada Gambar 3. 111 di baris ke-20 merupakan skrip untuk melakukan operasi hitung pengurangan variabel “endResult” dengan “startResult” untuk menghitung durasi operasi dan disimpan ke dalam variabel baru dengan nama “resultDurations”. Pada baris ke-23 menggunakan *if statement* untuk membandingkan variabel dan jika kondisi terpenuhi akan mencetak pesan “QC PASSED (Durasi operasi sesuai)”.

### C. Melakukan testing pada menu Form Pengkajian Kamar Operasi.

Jenis Pengkajian	Tanggal	Jam	Dikaji Oleh		
Skrining Pasien Terminal	29-Nov-2023	08:50	dr. Havin Sanjaya Sukamulyo, Sp.PD	View	Copy
Asesmen Rawat Jalan Pasien Geriatri	29-Nov-2023	08:50	dr. Havin Sanjaya Sukamulyo, Sp.PD	View	Copy

Gambar 3. 112 Halaman utama form pengkajian operasi

Pada Gambar 3. 112 merupakan halaman utama form pengkajian operasi, halaman ini digunakan oleh tim perawatan kesehatan, terutama mereka yang terlibat dalam operasi atau tindakan bedah, untuk mengevaluasi dan mencatat informasi penting terkait kondisi pasien, persiapan operasi, dan lingkungan kamar operasi. Formulir ini berfungsi sebagai alat dokumentasi yang membantu memastikan keselamatan dan keberhasilan prosedur operasi.

Gambar 3. 113 Layar pop-up form pengkajian pasien

Pada Gambar 3. 113 merupakan layar *pop-up* yang berisi informasi pasien dan pengisian formulir skrining pasien yang berupa radio button dan dapat dipilih sesuai dengan kondisi pasien.

Command	Target	Value
50 ✓ <i>click</i>	linkText=View	
51 ✓ <i>execute script</i>	return document.querySelectorAll("checkbox").forEach(b=>b.removeAttribute("disabled"))	
52 ✓ <i>execute script</i>	return document.querySelectorAll("radio").forEach(b=>b.removeAttribute("disabled"))	
53 ✓ <i>execute script</i>	return document.querySelector("input[type='checkbox']").click();	viewChkIsInitialAssessment

Gambar 3. 114 Skrip view ceklis pengkajian pasien

Pada Gambar 3. 114 di baris ke-51 merupakan skrip untuk menghapus elemen checkbox di halaman yang memiliki atribut "disabled". Metode "document.querySelectorAll" untuk mencari semua elemen input dengan atribut *type* yang setara dengan "checkbox" dan metode "forEach" untuk melakukan iterasi melalui setiap elemen checkbox dan menghapus atribut

"disabled" dari masing-masing elemen menggunakan “.removeAttribute("disabled")”. Pada baris ke-53 untuk memilih elemen checkbox dalam dokumen menggunakan selector CSS dan memeriksa status checkbox tersebut menggunakan “.checked”. Jika *checkbox* tercentang, nilai akan *true*, jika tidak nilainya akan *false*.

Command	Target	Value
76 ✓ <i>click</i>	css=valid	
77 ✓ <i>execute script</i>	return document.querySelector("input[type='checkbox']")id="cti00_cti00_cti00_ctGeneral_pcRightPanelContent_cbpRightPanelContent_cti04_cbpMPEN...	copyChkInitialAssesment
78 ✓ <i>echo</i>	`\${copyChkInitialAssesment}`	
79 ✓ <i>echo</i>	`\${viewChkInitialAssesment}`	
80 ✓ <i>if</i>	`\${copyChkInitialAssesment}`==`\${viewChkInitialAssesment}`	
81 <i>echo</i>	QC PASSED (Copy checkbox bagian pengkajian awal pasien berhasil)	
82 ✓ <i>else</i>		
83 ✓ <i>echo</i>	QC FAILED (Copy checkbox bagian pengkajian awal pasien tidak berhasil)	
84 ✓ <i>end</i>		

Gambar 3. 115 Skrip membandingkan checkbox form pengkajian

Pada Gambar 3.115 merupakan skrip yang digunakan untuk membandingkan variabel “copyChkInitialAssesment” sama dengan “viewChkInitialAssesment” akan mencetak pesan “QC PASSED (Copy checkbox bagian pengkajian awal pasien berhasil)”. Akan tetapi pada kasus ini, mencetak pesan “QC FAILED (Copy checkbox bagian pengkajian awal pasien tidak berhasil)”, hal ini dikarenakan nilai value dari kedua variabel berbeda yang dapat dilihat menggunakan perintah *echo* pada baris ke-78 dan baris ke-79.

### 3.2.9 Pembuatan test skenario dan hasil dokumentasi testing

Pembuatan test skenario merupakan langkah kunci dalam proses pengujian perangkat lunak. Skenario ini dirancang untuk mencakup berbagai situasi dan kondisi yang mungkin terjadi selama penggunaan perangkat lunak. Proses ini dimulai dengan identifikasi fungsionalitas utama yang harus diuji, kemudian dibuatlah serangkaian langkah-langkah atau skenario yang mencakup situasi normal dan kondisi batas.

Test skenario membantu tim pengujian untuk memastikan bahwa perangkat lunak dapat beroperasi sesuai yang diinginkan dan dapat mengatasi situasi yang mungkin muncul selama penggunaan.

Hasil dokumentasi testing mencakup rekaman semua kegiatan pengujian yang telah dilakukan dan hasil yang diperoleh dari setiap skenario. Dokumentasi ini berfungsi sebagai catatan lengkap dari seluruh proses pengujian, termasuk skenario yang dijalankan, input yang digunakan, serta output yang dihasilkan. Selain itu, hasil dokumentasi juga mencakup catatan tentang setiap bug atau masalah yang ditemukan selama pengujian, bersama dengan informasi terkait seperti langkah-langkah untuk mereproduksi bug tersebut. Dokumentasi testing yang baik membantu tim pengembangan untuk memahami kinerja perangkat lunak, memperbaiki bug, dan meningkatkan kualitas keseluruhan dari produk yang dihasilkan. Pembuatan test skenario pada PT Quantum Infra Solusindo dapat dilihat pada Tabel. 3. 2 dibawah ini.



Tabel 3. 2 Test skenario modul EMR pada menu Asesmen Umum

MODUL	MENU	TEST STEPS	RESULT	NOTES
EMR	Asesmen Umum	- Store data pada test "Data_Pasien" harus diubah setiap kali melakukan pengecekan.	PASS!	
		- Melakukan pendaftaran pasien baru rawat jalan dan beralih akun menggunakan dokter havin untuk melakukan pemeriksaan pengkajian.		
	Anamnesa	- Anamnesa harus diisi, jika tidak diisi dan klik simpan maka muncul pop up warning dan ada tulisan "Anemnesa harus diisi.", maka gagal disimpan > QC PASSED	PASS!	
		- Jika anamnesa sudah diisi dan berhasil klik simpan serta tidak ada pop up > QC PASSED.		
	Riwayat Pasien	- Input riwayat penyakit dahulu, jika berhasil disimpan dan muncul data > QC PASSED.	PASS!	
		- Input riwayat penggunaan obat, jika berhasil disimpan dan muncul data > QC PASSED.		
		- Input riwayat alergi dengan memilih tipe alergi, input jenis/nama dan reaksi. Jika apply berhasil maka akan muncul display data dan isinya sesuai > QC PASSED.		
	Tanda Vital	- Mengecek link button "+ Tanda Vital", jika muncul pop up tambah tanda vital > QC PASSED.	PASS!	
	- Input data tambah tanda vital, kemudian klik SAVE. Jika muncul display data setelah disimpan dan data sesuai dengan yang diinput > QC PASSED.			

MODUL	MENU	TEST STEPS	RESULT	NOTES
	Pemeriksaan Fisik	- Mengecek link button "+ Tambah pemeriksaan fisik", jika muncul pop up tambah tanda vital > QC PASSED	PASS!	
		- Mengecek radio button select all (Clear All, Not Examined, Normal, Abnormal, Other). Jika berhasil select all > QC PASSED.		
		- Mengecek radio button tertentu beserta text pada textbox kemudian SAVE. Jika diedit kembali maka radio button yang dipilih dan text yang diinput sesuai > QC PASSED.		
	Body Diagram	- Mengecek "+Body Diagram", jika muncul pop up tambah body diagram > QC PASSED.	PASS!	Harus refresh agar muncul tombol next dan previous
		- Memberikan tanda "Arrow" kemudian input keterangan pada notes. Setelah itu klik APPLY lalu SAVE. Jika muncul penanda gambar setelah disimpan > QC PASSED.		
		- Melakukan tambah body diagram kembali dari body diagram group dengan foto yang berbeda, kemudian klik APPLY dan SAVE. Jika muncul penanda gambar setelah disimpan > QC PASSED.		
		- Melakukan pengecekan tombol next dan previous jika data yang tampil sesuai dengan notes yang sudah diberi keterangan sebelumnya > QC PASSED.		
Diagnosis	- Input menggunakan search parameter pada bagian textbox ICD X, kemudian klik ceklis. Jika berhasil apply akan muncul display data > QC PASSED.	PASS!		

MODUL	MENU	TEST STEPS	RESULT	NOTES
		- Input pada textbox diagnosa text, kemudia klik ceklis. Jika berhasil apply akaan muncul display data > QC PASSED.		
		- Klik icon edit pada data yang sudah berhasil diapply, jika isi data sesuai > QC PASSED.		
	Pemeriksaan Laboratorium	- Mengecek linkbutton Tambah Order (List) dan Order (Form), jika muncul pop up > QC PASSED.	PASS!	
		- Melakukan pemeriksaan check dan uncheck item Tambah Order dan pemeriksaan CITO setelah disimpan.		
		- Melakukan pemeriksaan item yang dipilih dan ceklis CITO dan pemeriksaan PA setelah disimpan.		
	Pemeriksaan Radiologi	- Mengecek linkbutton Tambah Order, Order (Tanpa Detail), dan Order Form, jika muncul pop up > QC PASSED.	FAIL!	- Ceklis "Pemeriksaan PA" bagian Tambah Order tidak berhasil setelah klik edit
		- Melakukan pemeriksaan check dan uncheck item Tambah Order dan pemeriksaan CITO setelah disimpan.		- Ceklis CITO bagian Order (Tanpa Detail) tidak berhasil setelah klik edit.
		- Melakukan pemeriksaan item pada Order Form yang dipilih dan ceklis CITO setelah disimpan.		
		- Melakukan pemeriksaan CITO dan catatan pemeriksaan pada Order (Tanpa Detail).		
	Pemeriksaan Penunjang Medis Lain-Lain	- Mengecek linkbutton Tambah Order, jika muncul pop up > QC PASSED.	PASS!	

MODUL	MENU	TEST STEPS	RESULT	NOTES
		- Melakukan pemeriksaan check dan uncheck item Tambah Order dan pemeriksaan CITO setelah disimpan.		
	Prosedur/Tindakan	- Input menggunakan search parameter pada bagian diagnosa/tindakan, kemudian klik ceklis. Jika berhasil diapply akan muncul display data > QC PASSED.	PASS!	
		- Input pada textbox prosedur/tindakan text, kemudian klik ceklis. Jika berhasil diapply akan muncul display data > QC PASSED.		
		- Klik icon edit pada data yang sudah berhasil diapply, jika isi data sesuai > QC PASSED.		
	Tindakan Terapi dan Instruksi Dokter	- Mengecek bagian textbox, jika sudah terisi > QC PASSED.	PASS!	
	Jadwal Kunjungan Berikutnya	- Input tanggal kunjungan berikutnya, klik SAVE. Jika tanggal pada textbox Catatan Perjanjian Selanjutnya sesuai dengan yang diinput > QC PASSED.	PASS!	Setelah klik save harus refresh agar muncul tampilan Catatan Perjanjian Selanjutnya

### **3.3 Kendala yang Ditemukan**

Selama proses kerja magang di PT Quantum Infra Solusindo dengan posisi sebagai *Application Quality Control Intern*, terdapat kendala yang ditemukan atau dialami selama kerja magang, yaitu :

1. Aplikasi SIMRS Medinfras memiliki alur yang sangat kompleks. Hal ini menyebabkan banyak interaksi dan validasi data kesehatan yang spesifik.
2. Kurangnya pengetahuan tentang *tools* Selenium IDE. Memerlukan waktu yang cukup banyak untuk menemukan skrip karena sebelumnya perusahaan melakukan *testing* secara manual.
3. Aplikasi SIMRS Medinfras memerlukan penyesuaian fitur dalam penggunaan *automation testing* menggunakan *tools* Selenium IDE. Kesulitan dalam mendeteksi elemen pada halaman web karena ada perubahan tata letak halaman atau atribut ID elemen HTML.

### **3.4 Solusi atas Kendala yang Ditemukan**

Selama proses kerja magang di PT Quantum Infra Solusindo, ditemukan beberapa kendala yang harus dilewati dalam kerja magang sebagai *Application Quality Control Intern*. Adapun solusi yang ditemukan dalam menghadapi kendala dan terus melaksanakan program kerja magang dengan efektif, yaitu :

1. Melakukan analisis yang lebih mendalam tentang alur aplikasi SIMRS Medinfras dengan melakukan pencatatan atau dokumentasi setiap langkah yang diperlukan dalam proses dan melakukan komunikasi dengan tim.
2. Belajar secara mandiri melalui sumber-sumber di website, serta melihat video penjelasan melalui YouTube dan berdiskusi langsung dengan pembimbing lapangan.
3. Melakukan identifikasi elemen berdasarkan lokasi atau struktur HTML yang lebih spesifik dan kurang rentan terhadap perubahan.