**Laporan Akhir Pengujian Perangkat Lunak( Pemesanan Obat, my.unpam.ac.id)**



Nama : Tubagus Ganendra Krisna Priya

NM : 2239099057

Kelas : PKPL B 2022

Disusun Untuk memenuhi tugas UAS (Pengganti Praktikum) Mata Kuliah PKPL

Link Github : <https://github.com/Ganen04/Project-PKPL>

Link Youtube : <https://youtu.be/y8qgo2XwqCc>

Fakultas Teknologi Industri

Universitas Ahdmad Dahlan Yogyakarta

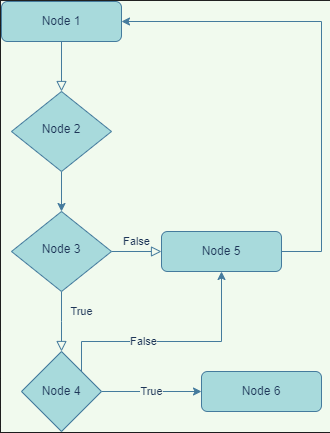
* (Diagram Path Dari Source)

|  |  |
| --- | --- |
| Kode Program | Node |
| <?php  //ambil data dari form login  $btn=$\_POST['login'];  $user=$\_POST['user'];  $pwd=$\_POST['pwd'];  $pwd\_enkripsi= md5($pwd); | 1 |
| <?php  // mysqli\_connect("hostname","user database","password","nama")  $koneksi= mysqli\_connect("localhost","root","","db\_pemesanan") or die("Tidak bisa terhubungan ke database"); | 2 |
| include 'config/koneksi.php';  $sql="SELECT \* FROM user WHERE username='$user' | 3 |
| include 'config/koneksi.php';  $sql="SELECT \* FROM user WHERE paswd='$pwd\_enkripsi'"; | 4 |
| if($jumlahdata > 0){  $data= mysqli\_fetch\_array($query); //ambil data dan konversi menjadi array  session\_start(); //aktifkan session wajib  $\_SESSION['username']=$user;  $\_SESSION['idsesi']=session\_id();  $\_SESSION['level']=$data['level'];  $\_SESSION['nama']=$data['nama'];  $\_SESSION['ket']=$data['ket'];  $\_SESSION['email']=$data['email'];  //pindahkan ke halaman index  header("location:index\_admin.php", true); | 6 |
| }else{  echo "<script> window.location.assign('index.php?error=yes');</script>"; | 5 |

Tabel Listing Source Route Form Login

Penjelasan Node :

* Node 1 : Input data Username dan Password dari form login
* Node 2 : Submit input pada form login dan menghubungkan ke Database
* Node 3 : Mengecek Username
* Node 4 : Mengecek Password
* Node 5 : Error (Kesalahan Username atau Password)
* Node 6 : Proses login menghubungkan ke index\_admin.php



R1 : 1 – 2 – 3 – 4 – 6

R2 : 1 – 2 – 3 – 5 – 1 – 2 – 3 – 4 – 6

R3 : 1 – 2 – 3 – 4 – 5 – 1 – 2 – 3 – 4 – 6

Diagram Path Form Login Aplikasi Pemesanan Obat

Berdasarkan Diagram Path diatas diketahui bahwa jumlah edge (E) = 7 yang merupakan garis yang menghubungkan node, jumlah node (N) = 6, jumlah predicate (P) = 2 yang merupakan node bercabang, dan jumlah region (R) = 3 yang merupakan satu area dalam flow graph.

* ( Cyclomatic Complexity)

|  |  |  |
| --- | --- | --- |
| *V(G)* = E – N + 2 | *V(G)* = P + 1 | *V(G)* = R |
| *V(G)* = 7 – 6 + 2 | *V(G)* = 2 + 1 | *V(G)* = 3 |
| *V(G*) = 3 | *V(G)* = 3 |  |

Cyclomatic Complexity Route From Login

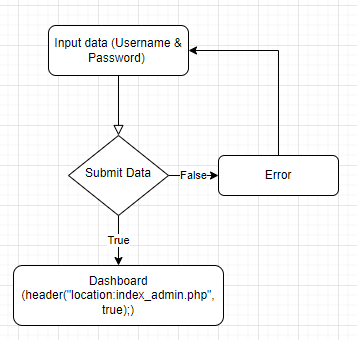
* (Tabel Jalur Bebas)

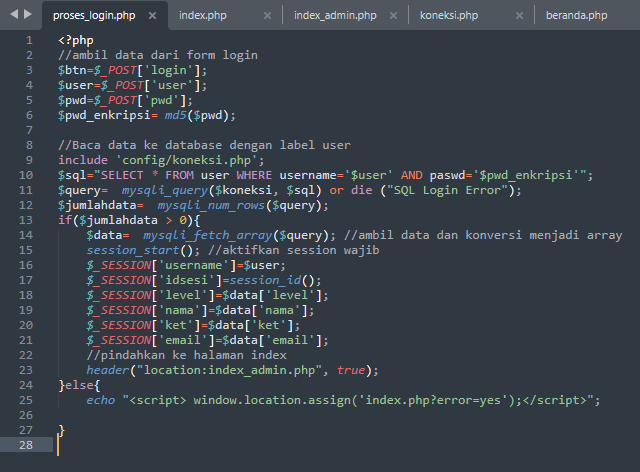
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | Hasil |
| 1 |  | 1 |  |  |  |  | 0 |
| 2 |  |  | 1 |  |  |  | 0 |
| 3 |  |  |  | 1 | 1 |  | 1 |
| 4 |  |  |  |  | 1 | 1 | 1 |
| 5 | 1 |  |  |  |  |  | 0 |
| 6 |  |  |  |  |  |  | 0 |

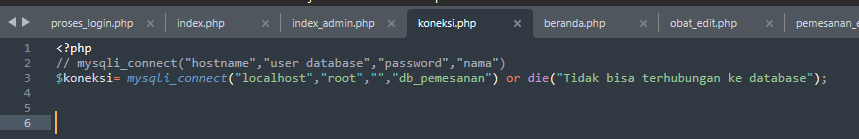
* Poin 6 Skenario Uji (BVA)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Region | Keterangan | Expected Result | Actual result | Pass/Fail |
| S01 | 1 – 2 – 3 – 4 – 6 | 1 = True  2 = True  3 = True  4 = True | T | T | Pass |
| S02 | 1 – 2 – 3 – 5 – 1 – 2 – 3 – 4 – 6 | 1 = True  2 = True  3 = False  5 = True  1 = True  2 = True  3 = True  4 = True | T | T | Pass |
| S03 | 1 – 2 – 3 – 4 – 5 – 1 – 2 – 3 – 4 – 6 | 1 = True  2 = True  3 = True  4 = False  5 = True  1 = True  2 = True  3 = True  4 = True | T | T | Pass |

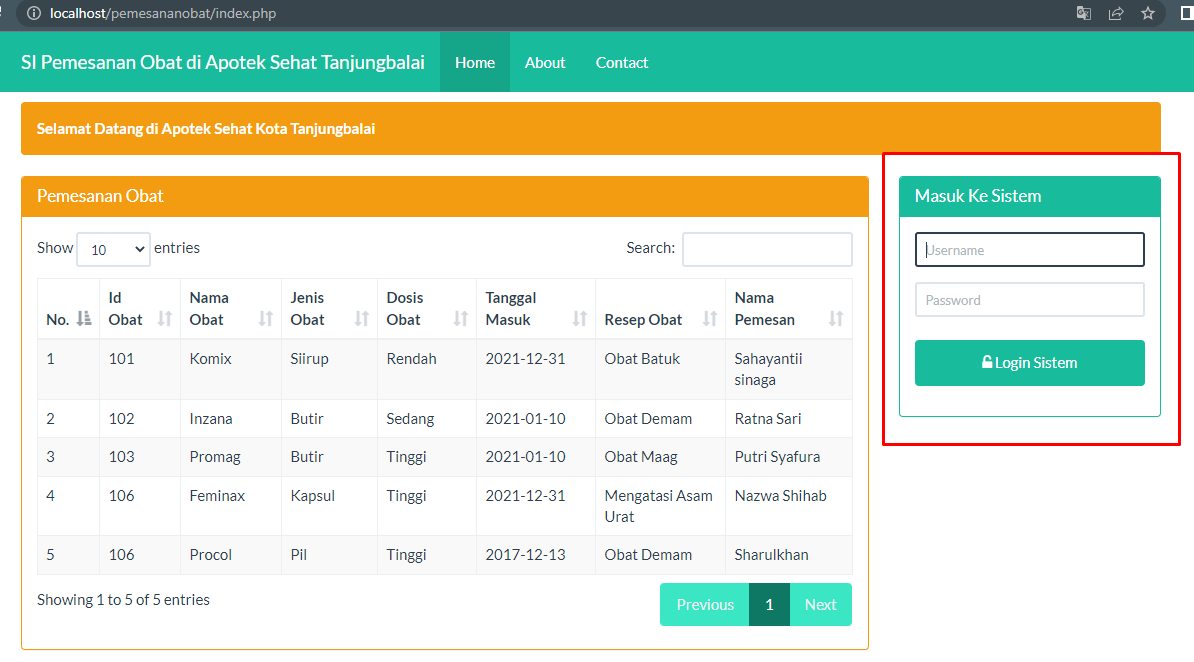
* Hasil Uji White Box







* Hasil Uji Black Box

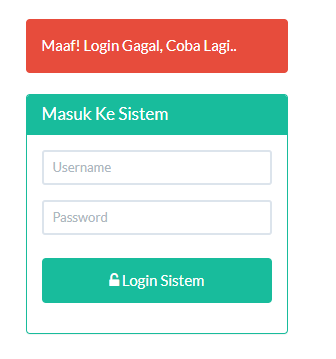


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Inputs | C1 | C2 | C3 | C4 | C5 | C6 |
| Username | Yes | Yes | Yes | No | Yes | No |
| Password | Yes | No | Yes | Yes | No | No |
| Login Sistem | Yes | Yes | No | Yes | No | No |
| Result |  | Error | Error | Error | Error | Error |

|  |  |  |  |
| --- | --- | --- | --- |
| No Test | Category | Test Scenario | Expected Result |
| 01 | Form Login | User Menginput Username dan Password yang tersimpan pada server dan mensubmit untuk validasi Username dan Password | Pass Menuju dashboard  (index\_admin.php) |



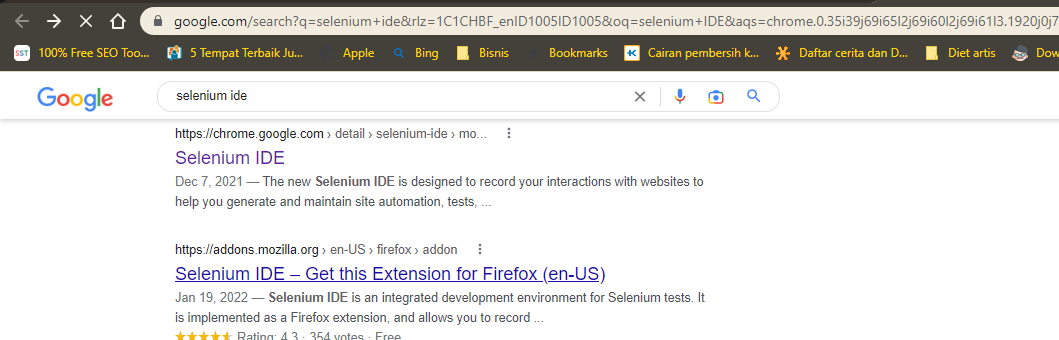
Actual result (True)



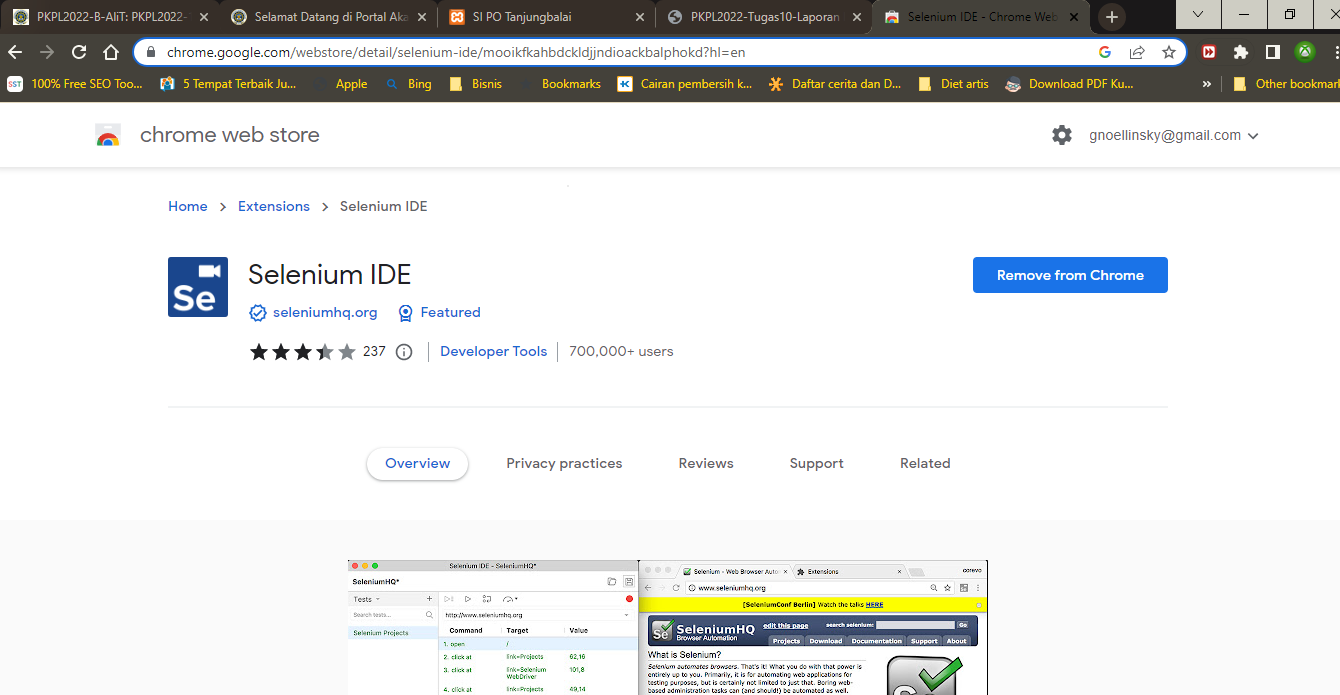
Actual Result (False)

* **Pengujian Otomatis Aplikasi Pemesanan Obat dengan Selenium IDE**

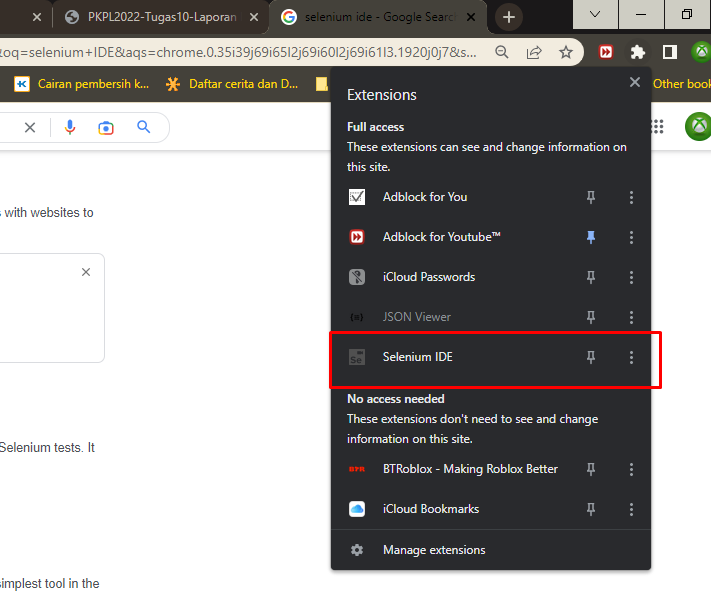
Langkah pertama yaitu buka Chrome, kemudian ketikan selenium IDE pada kolom pencarian.



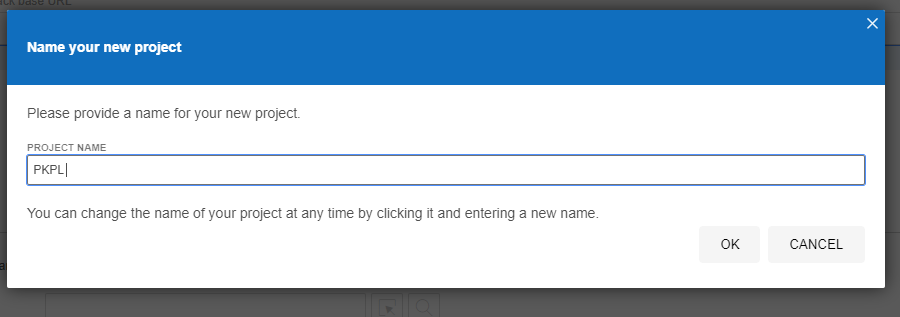
Kemudian klik pada chrome web store, dan install. Setelah itu klik add to chrome extension.



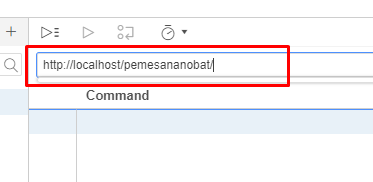
Setelah terinstal, klik menu extensions dan pilih Selenium IDE



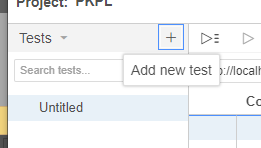
Setekah itu, tab Selenium IDE akan terbuka. Untuk memulai proyek ,pertama-tama klik create new project, kemudian berikan nama untuk proyek yang akan dikerjakan.



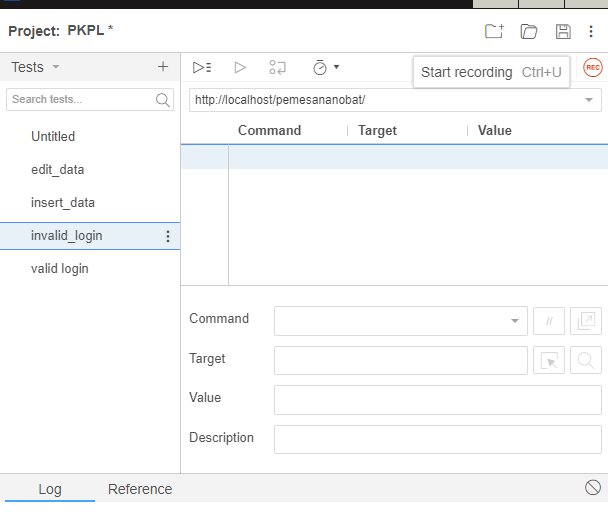
Masukan link URL untuk aplikasi yang akan diuji coba.



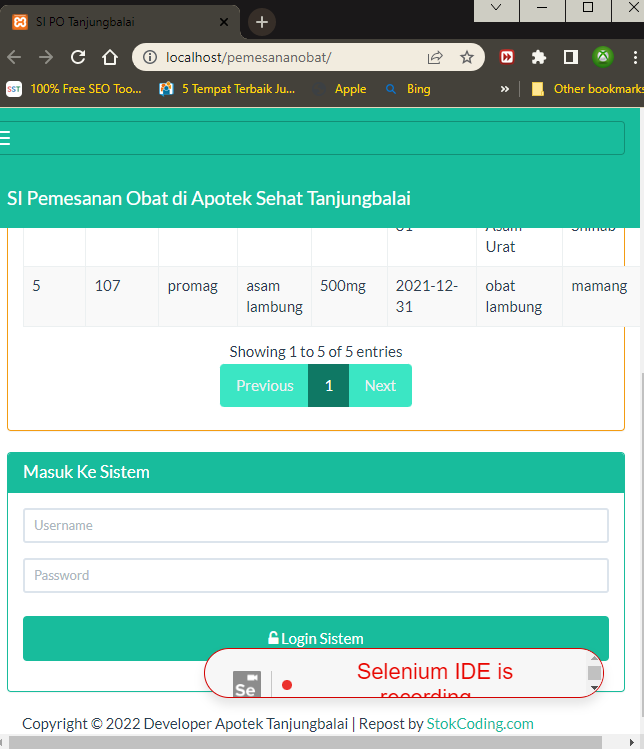
Klik ikon “+” untuk membuat test baru dan berikan nama untuk pengujian yang akan dilakukan.



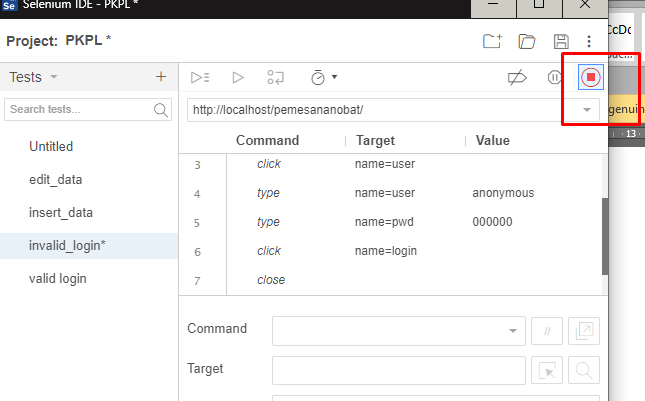
Untuk memulai pengujian, klik menu Record pada nama pengujian yang dipilih.



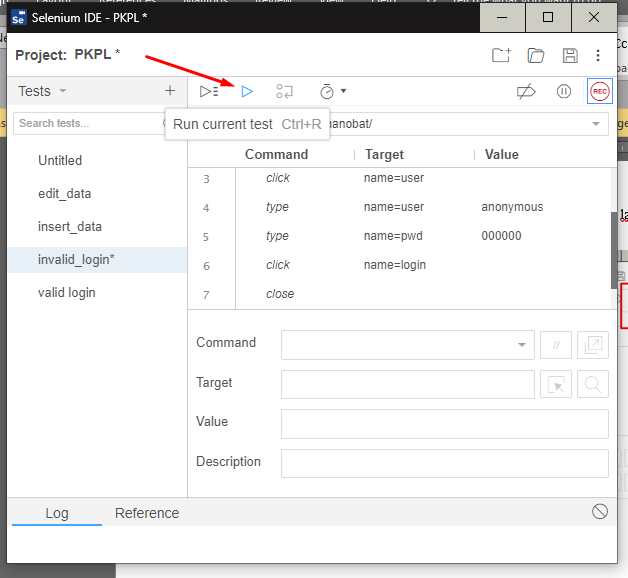
Kemudian nantinya selenium IDE akan mebuka tab baru yang menuju kepada web yang akan diuji. Setelah itu lakukan pengujian sesuai dengan fitur yang akan di uji cobakan.



Jika sudah selesai, kik tombol close, dan hentikan recording. Setelah itu, langkah pada pengujian akan ditampilkan.



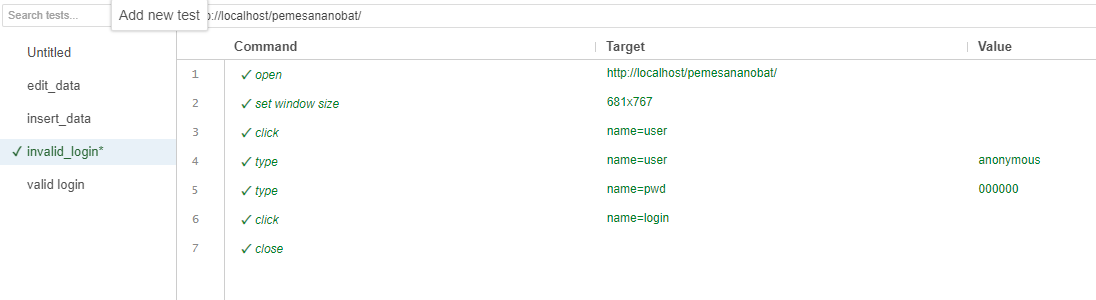
Untuk melakukan uji otomatis, klik tombol run test.



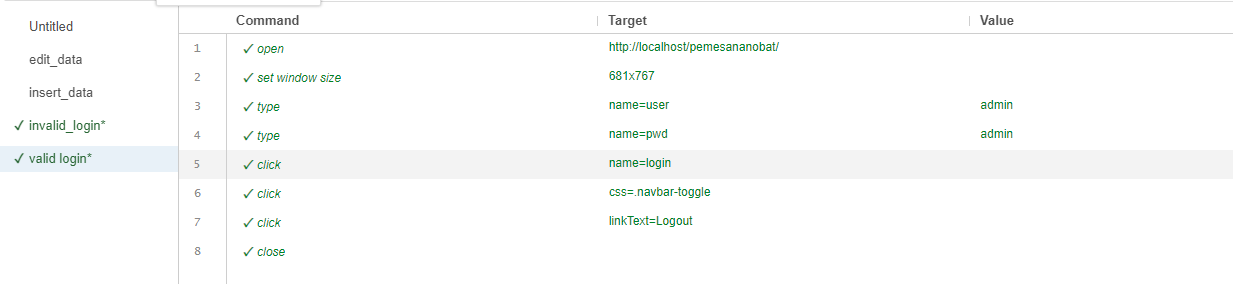
Kemudian Selenium IDE akan menjalankan test secara otomatis dengan membuka tab baru yang merupakan website yang akan di ujicoba. Setelah uji coba selesai hasil pengujian akan ditampilkan.

Berikut adalah hasil pengujian yang telah dilakukan kepada aplikasi Pemesanan Obat. :

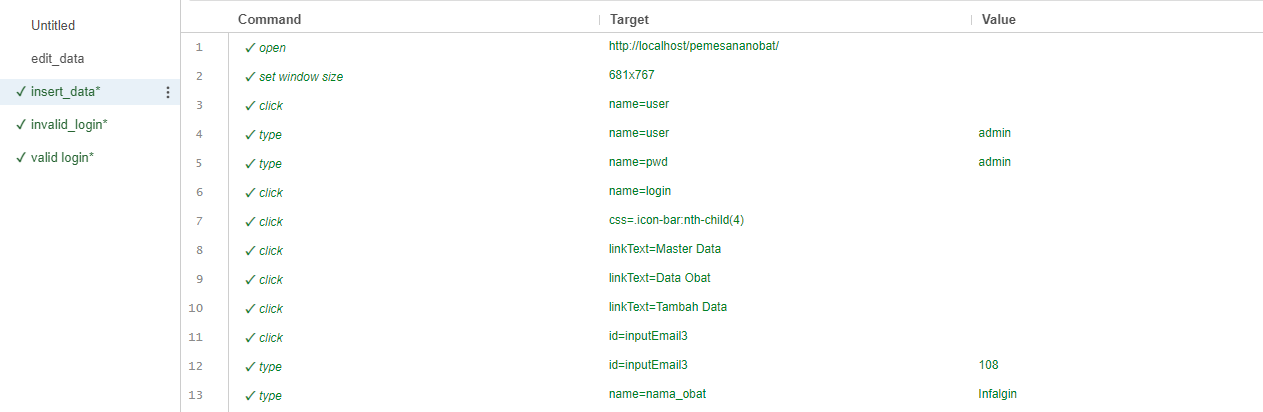
* **Invalid Login**

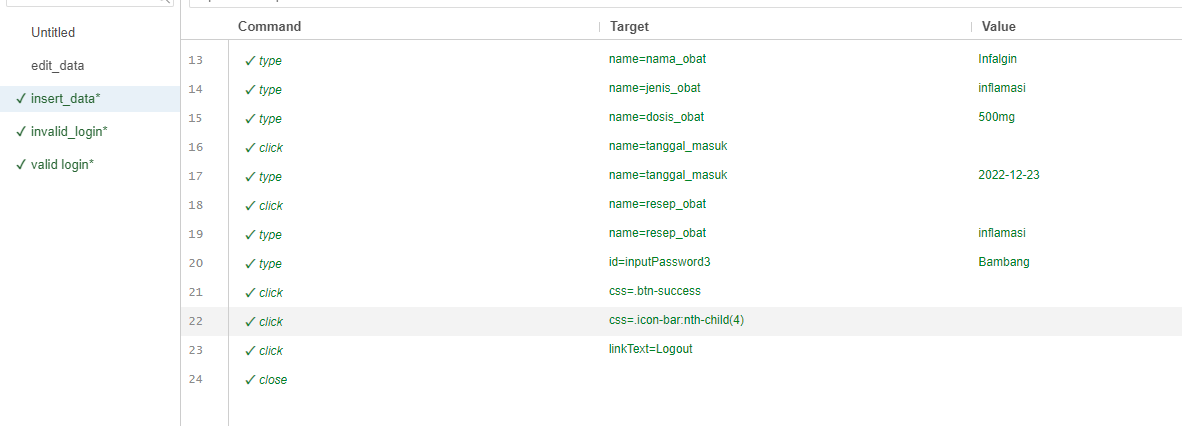


* **Valid Login**

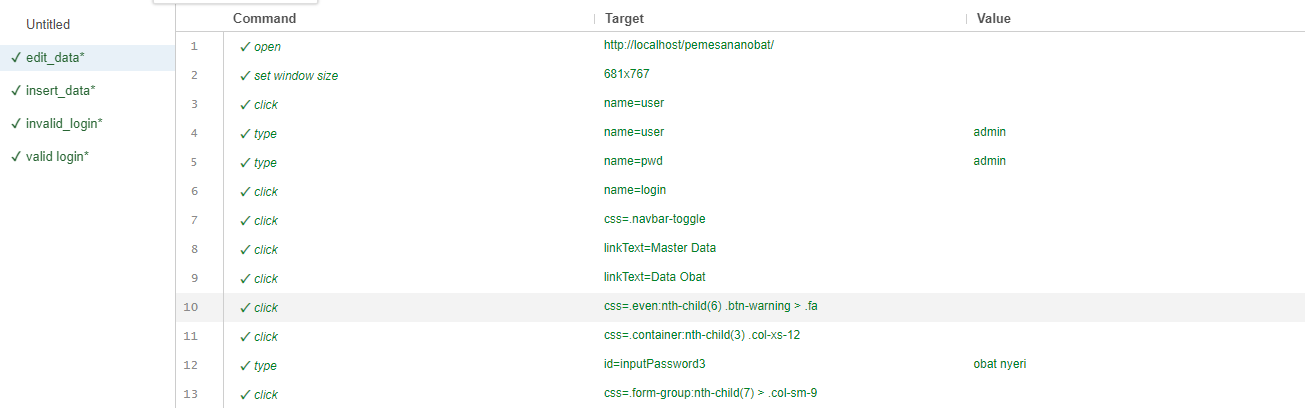


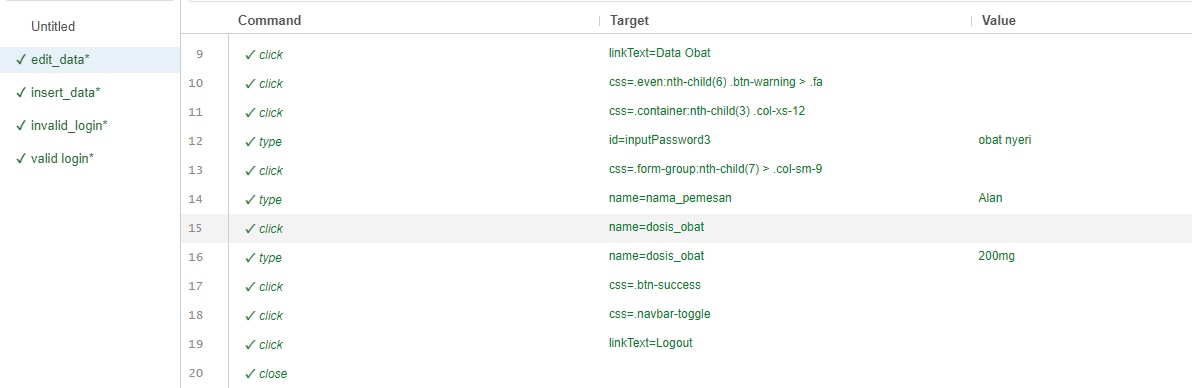
* **Insert Data**





* **Edit Data**

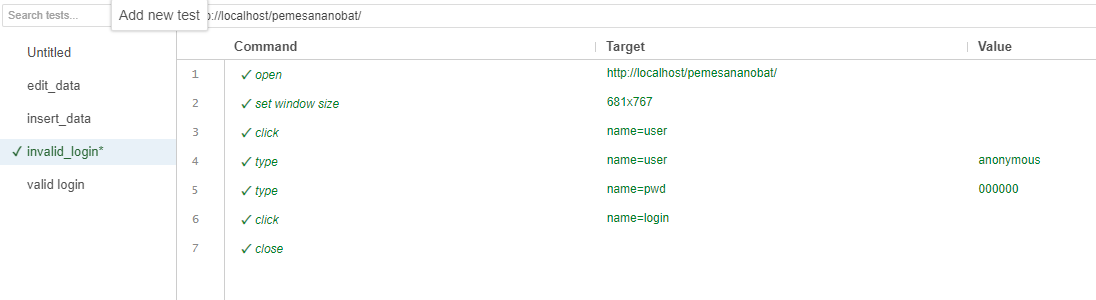




* **Laporan Pengujian Otomatis Aplikasi Pemesanan Obat dengan Selenium IDE**

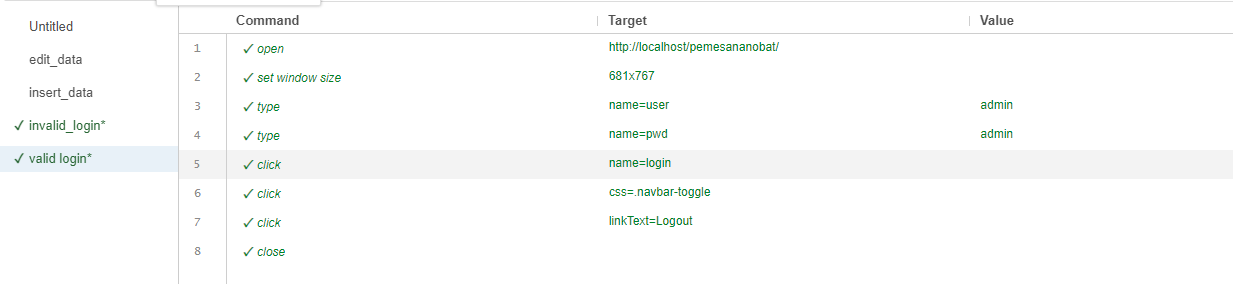
Ada 4 fitur yang diujikan dalam pengujian otomatis, dengan hasil pengujian yang telah dilakukan kepada aplikasi Pemesanan Obat. :

* **Invalid Login**



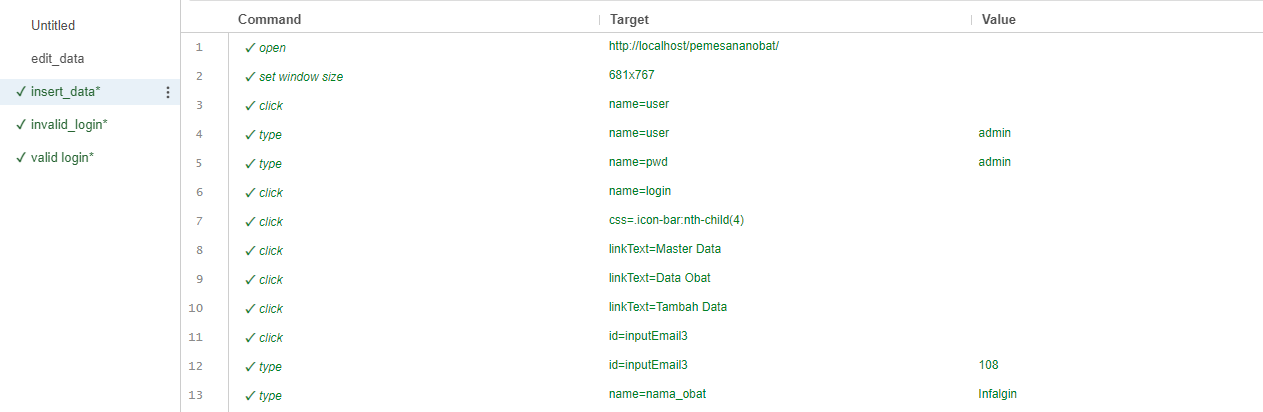
Pada pengujian invalid login, terdapat 7 langkah pengujian. Tujuan dari pengujian ini adalah untuk mengetahui apabila ada kesalahan dari username dan password yang diinputkan pada aplikasi apakah user tetap dapat mengakses aplikasi atau tidak dengan hasil uji yang diharapkan adalah *user tidak akan dapat mengakses aplikasi*. Pada langkah pengujian ke-4 dan ke-5 adalah langkah ketika memasukan username dan password yang salah dan hasilnya adalah sukses dan tidak ada kejanggalan pada system.

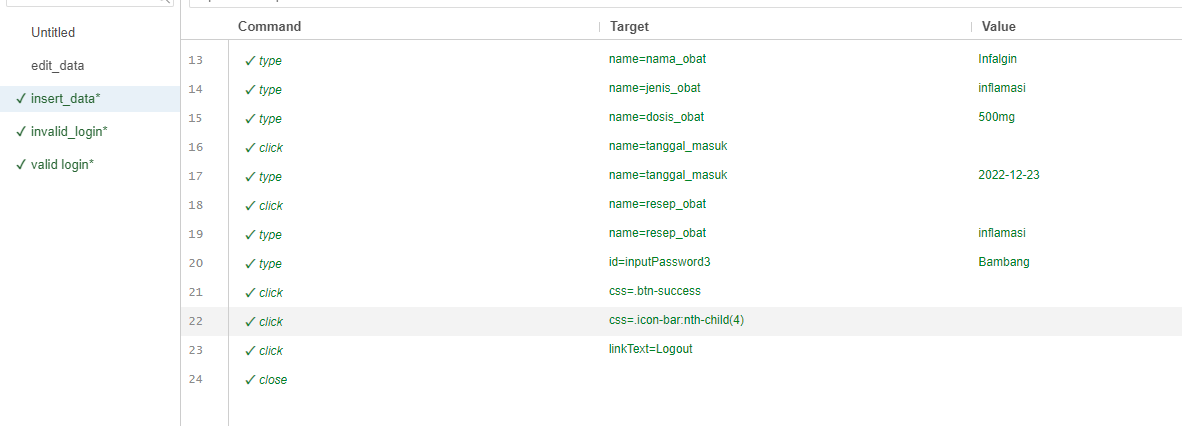
* **Valid Login**



Kemudian pengujian yang kedua adalah pengujian valid login yang memiliki 8 langkah pengujian. Tujuan dari pengujian ini adalah untuk mengetahui bila username dan password yang diinputkan pada aplikasi adalah username dan password yang terdaftar, apakah user akan dapat mengakses data pada aplikasi atau tidak dengan hasil yang diharapkan adalah *user dapat mengakses data pada aplikasi*. Pada langkah pengujian ke-3 dan ke-4 adalah langkah ketika memasukan username dan password yang terdaftar dan hasilnya adalah user dapat mengakses data dan tidak ada kejanggalan pada system. Kemudian pada langkah ke-7 adalah pengujian logout dan user berhasil logout.

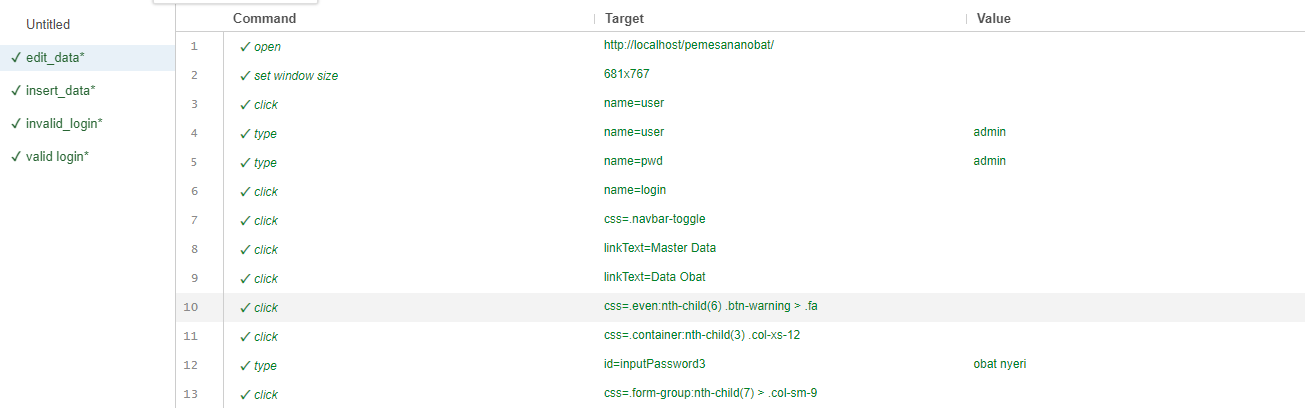
* **Insert Data**

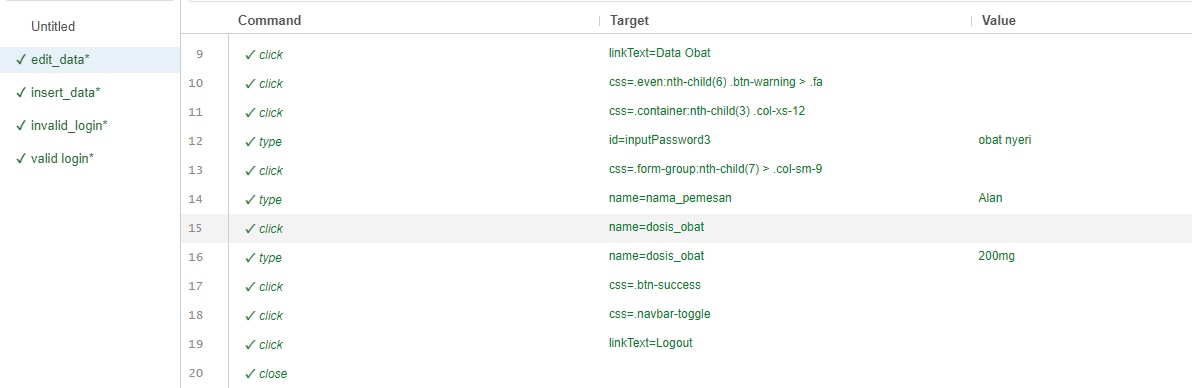




Pada pengujian ke-3, yang menjadi acuan uji coba adalah fitur menambah data pada aplikasi pemesanan obat. Tujuan dari uji coba insert\_data adalah untuk mengetahui apakah fitur insert data yang ada pada aplikasi berjalan dengan baik dengan hasil yang diharapkan adalah *Data dapat ditambahkan pada aplikasi*. Pengujian ke-12 sampai ke-20 adalah langkah pengujian untuk insert data, dengan hasil pengujiannya adalah sukses dan data berhasil ditambahkan.

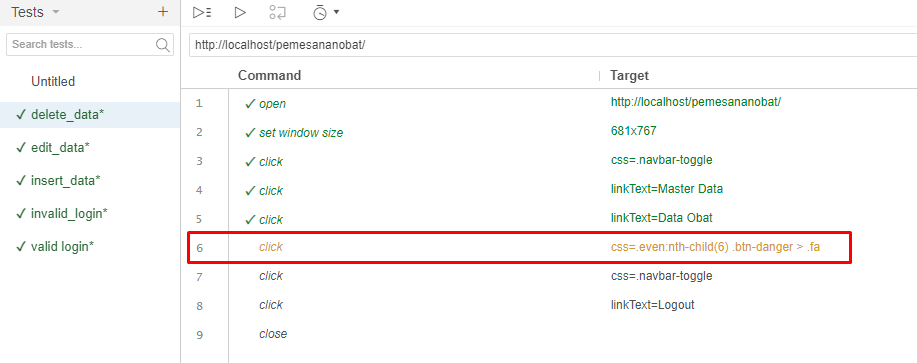
* **Edit Data**





Pada pengujian ke-4, yang menjadi acuan uji coba adalah fitur mengedit data pada aplikasi pemesanan obat. Tujuan dari uji coba edit\_data adalah untuk mengetahui apakah fitur editdata yang ada pada aplikasi berjalan dengan baik dengan hasil yang diharapkan adalah *Data dapat diedit atau diupdate pada aplikasi*. Langkah pengujian ke-12 sampai ke-16 adalah langkah pengujian untuk edit data, dengan hasil pengujiannya adalah sukses dan data berhasil diubah.

* Kejanggalan pada pengujian
* Delete data



Pada pengujian delete data, uji coba memiliki 9 langkah. Sebelum melakukan pengujian otomatis, langkah-langkah pengujian akan ditentukan terlebih dahulu setelah itu baru pengujian otomatis dapat dilakukan. Pada langkah ke-6 data yang akan dihapus tidak dapat ditemukan karena data sudah terlebih dahulu terhapus pada penentuan langkah-langkah pengujian otomatis. Hal ini membuat pengujian otomatis tidak dapat dilanjutkan, dan langkah pengujian terjebak pada langkah ke-6. Menurut pendapat saya, pengujian otomatis dengan Selenium IDE tidak cocok untuk menguji fitur menghapus data.

* **Pengujian Non-Fungsional Stress dengan JMeter pada aplikasi my.unpam.ac.id**

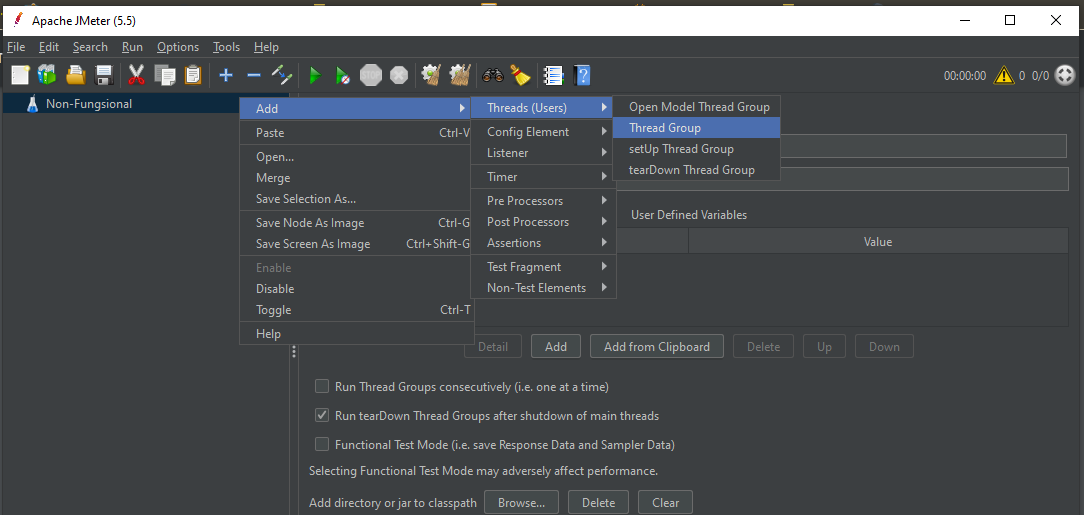
Metode pengujian yang akan dilakukan adalah pengujian Non-Fungsional Stress test dengan menggunakan aplikasi Apache JMeter. Tujuan pengujian Stress Test adalah untuk memeriksa kemampuan dari aplikasi yang akan ditest dalam menerima traffic dari luar, dengan tujuan menguji aplikasi/website apakah akan terjadi down atau tidak saat banyak user yang mengakses aplikasi, website tersebut.

* **Rangkaian Pengujian**

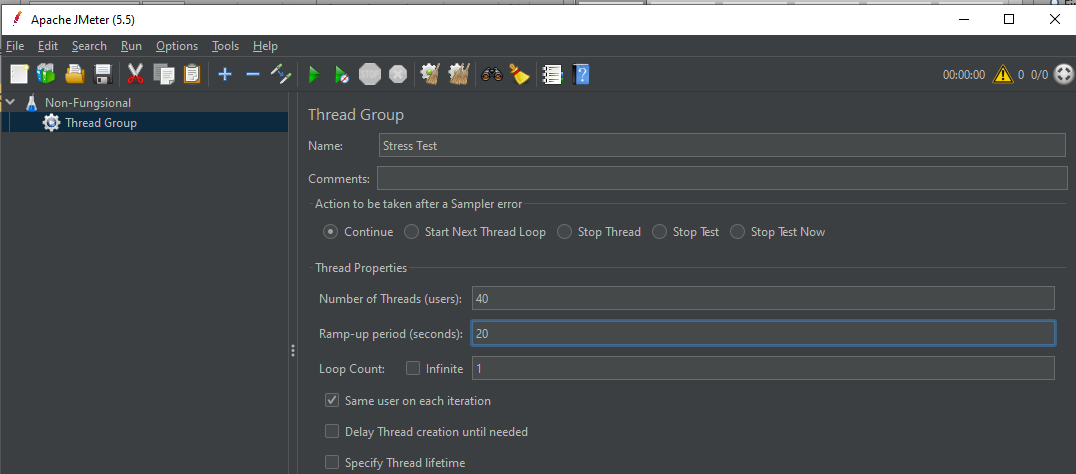
Pengujian yang akan dilakukan adalah :

* Threads Test :
* Homepage
* Login
* Particular Element Test
* **Tahapan Pengujian**

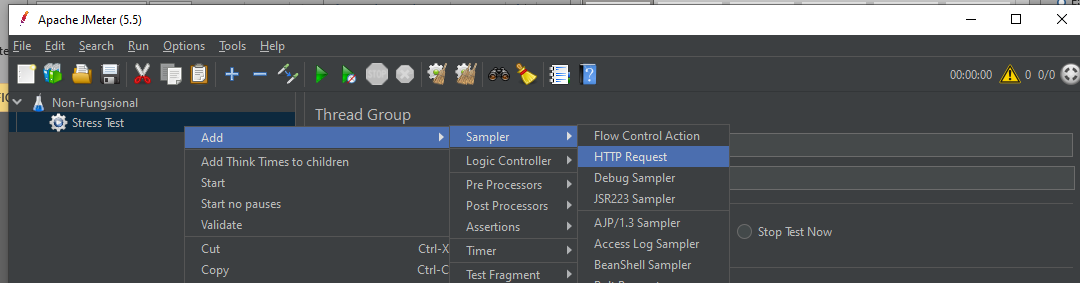
Langkah pertama, buka aplikasi Apache JMeter, kemudian klik Kanan dan pilih Add => Threads(User) => Thread Group.



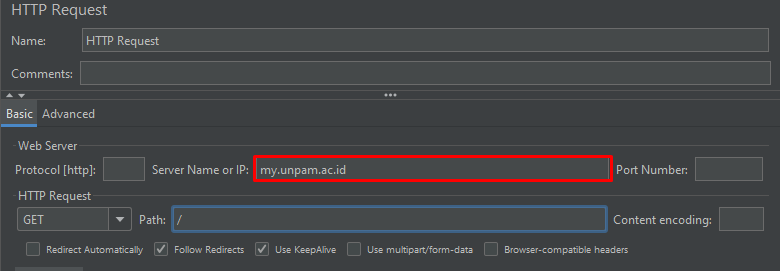
Pada Menu Threads nama pengujian saya isi dengan “Stress Test”. Kemudian untuk Number of User saya isi sebanyak 40, lama periode sebanyak 20 dan loop sebanyak 1.



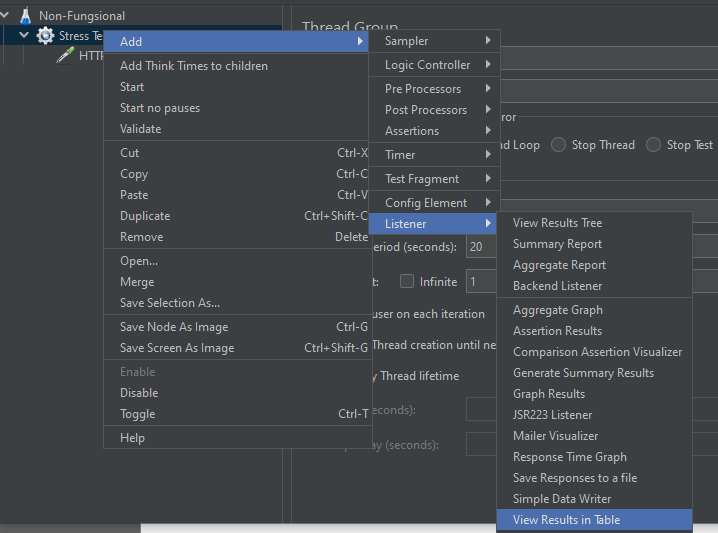
Kemudian, karena aplikasi yang akan diujikan berbasis web maka klik kanan pada Stress Test, klik Add => Sampler=> HTTP Request.



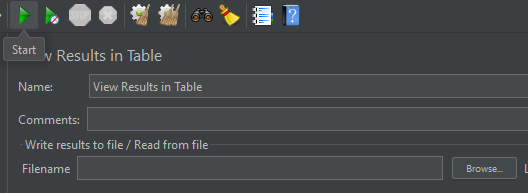
Pada HTTP Request , masukan server address dari aplikasi yang akan diujikan. Pada bagian Path karena pengujian pertama yang akan dilakukan adalah Homepage test maka isi dengan “/”



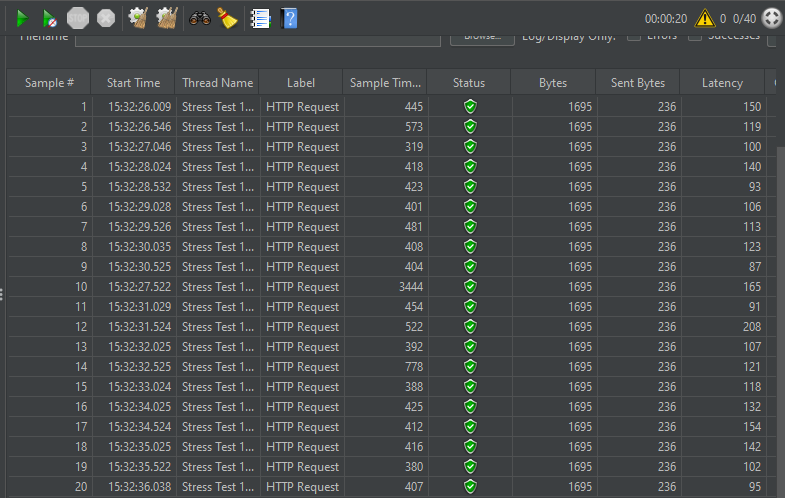
Untuk melihat hasil pengujian yang akan dilakukan, klik kanan pada Stress Test kemudian pilih Add=>Listener=>View Result in Table. Maka nanti Hasil pengujian akan ditampilkan dalam bentuk table.

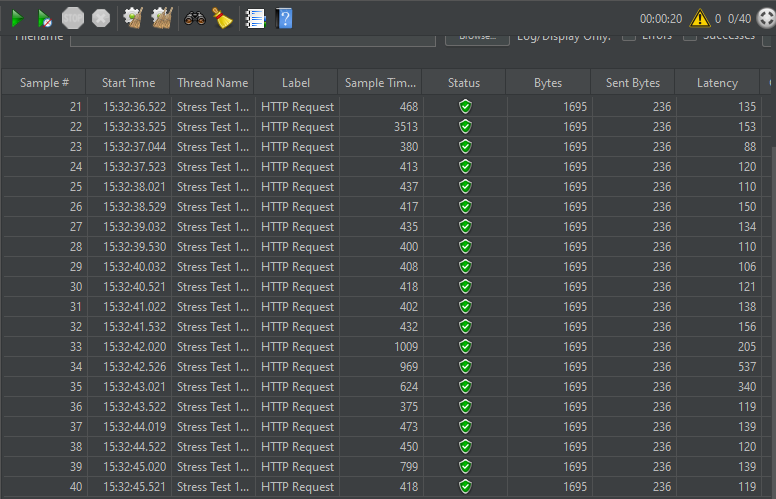


Untuk melakukan pengujian klik ikon start, namun sebelum melakukan pengujian alangkah baiknya menyimpan hasil pengujian terlebih dahulu.

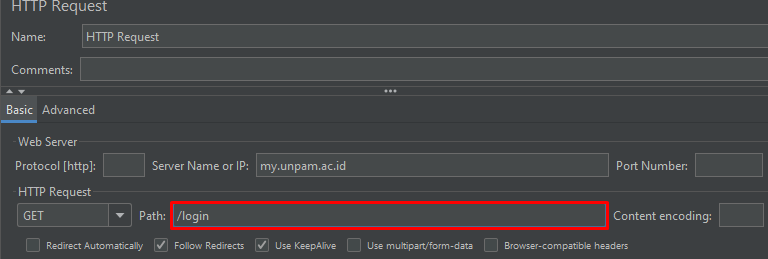


Berikut adalah hasil pengujian pada Homepage

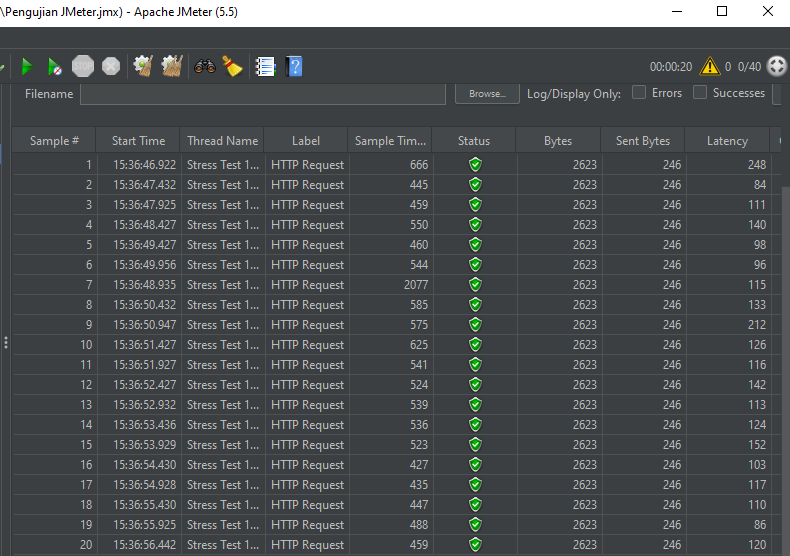


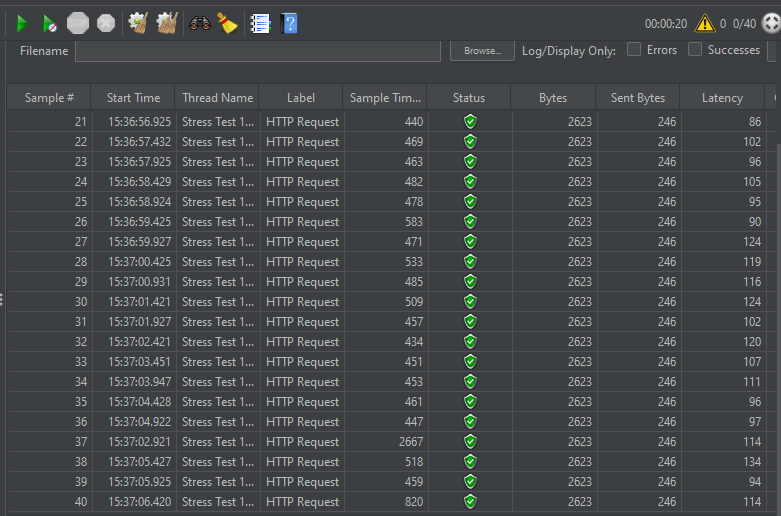


Sebanyak 40 user virtual yang diujikan dengan lama periode 20 detik dan 1 pengulangan hasilnya adalah semua user berhasil dan tidak ada yang gagal.



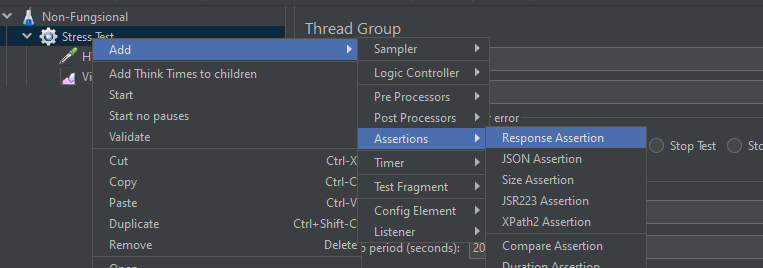
Selanjutnya adalah pengujian pada Login page dengan pengaturan yang sama dengan homepage yakni sebanyak 40 user virtual dengan panjang periode 20 detik. Hasilnya :



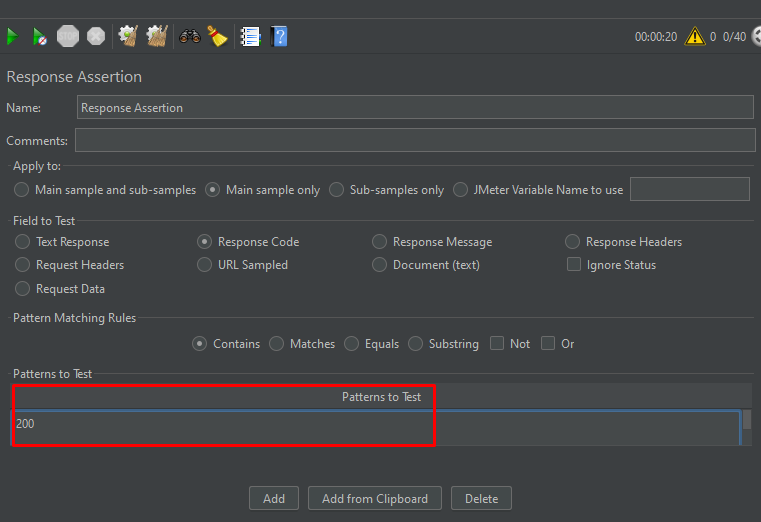


Particular Element Test

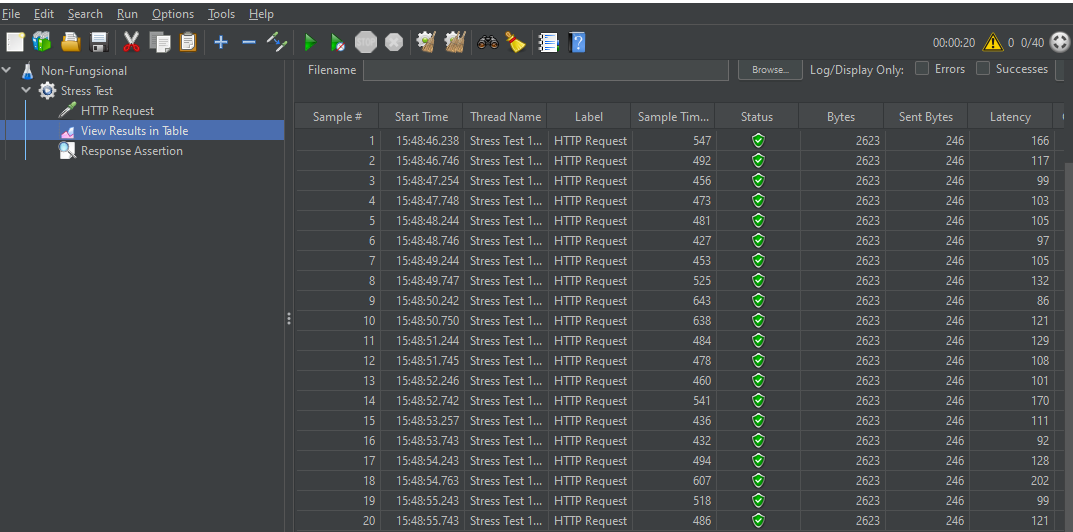
Untuk melakukan pengujian Particular Element, klik kanan pada Stress Test, kemudian klik Add=>Assertions=>Response Assertion.



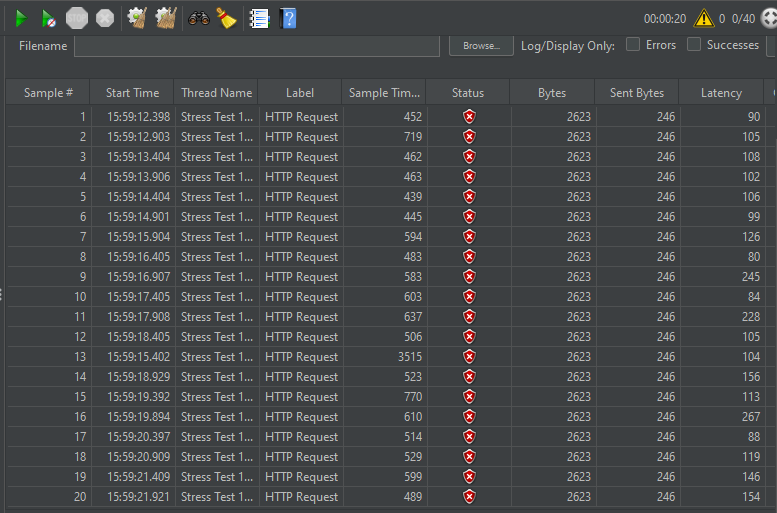
Lalu pada pengaturan Response Assertion lakukan seperti diawah ini. Untuk Pattern to Test sebelumnya klik Add, dan masukan 200 yang merupakan kode default dari “HTTP”.

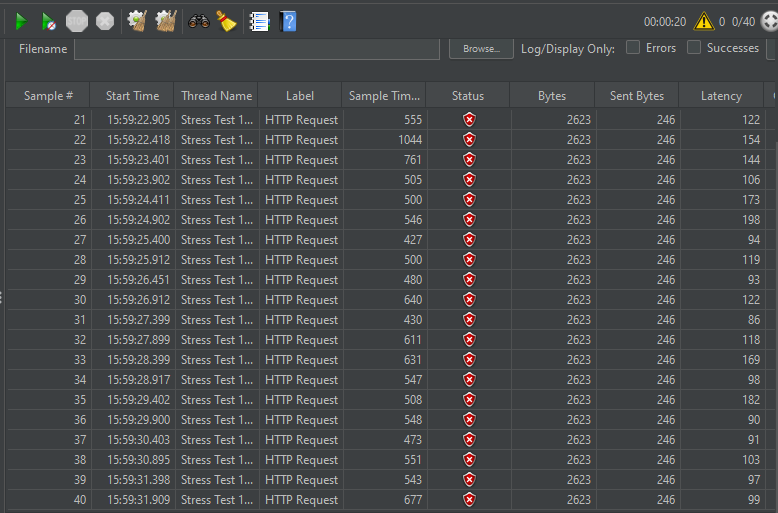


Untuk melihat hasil pengujian klik menu “View Result in Table”.

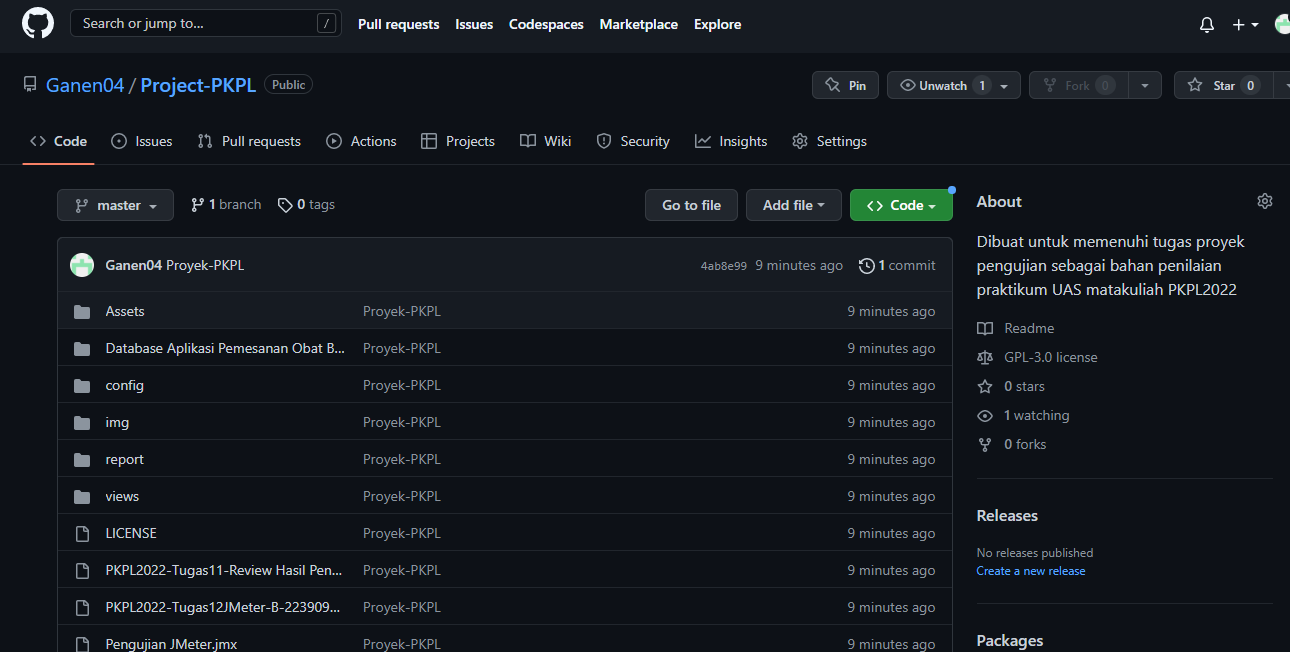


Berikut adalah hasil pengujian Particular Element dengan seluruh User virtual berhasil tanpa mengalami kegagalan. Namun apabila Pattern to Test diganti dengan kode 201 maka hasilnya akan gagal seperti berikut :



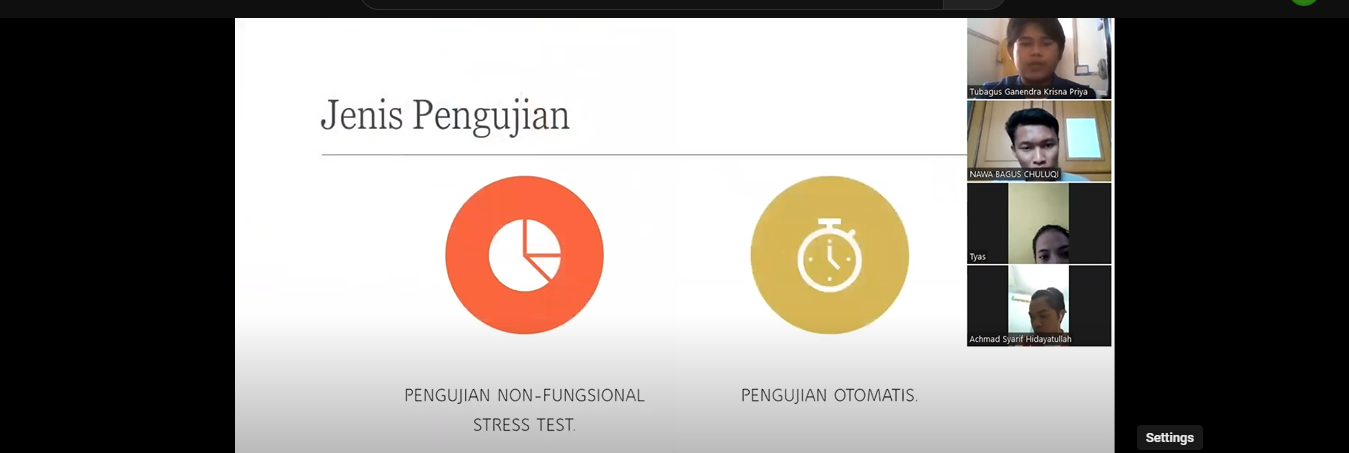


* **File yang telah di upload pada Github**



File aplikasi dan laporan proyek telah diupload pada repository yang diberi nama “Proyek-PKPL” dengan link <https://github.com/Ganen04/Project-PKPL>

* **Presentasi yang telah diunggah ke Youtube**



Pada presentasi yang sudah saya buat, saya mempresentasikan mengenai pengujian otomatis dan pengujian Non-Fungsional Stress Test pada aplikasi pemesanan obat dan aplikasi pengaksesan data mahasiswa yaitu my.unpam.ac.id. Durasi Video 4 menit 46 detik yang saya lakukan dengan hadirnya beberapa audience dalam presentasi yang saya lakukan. Berikut link Video Youtube dari presentasi yang telah saya lakukan <https://youtu.be/y8qgo2XwqCc>

* **Analisa Proyek**

Penggunaan metode pengujian dapat dilakukan dan dipilih dengan menyesuaikan aplikasi yang akan kita uji dan juga spesifikasi yang ingin dicapai. Pada proses pemilihan metode dapat dipilih pada saat perancangan metode pengujian. Metode yang tidak tepat akan menyebabkan hambatan pada waktu pengujian. Kemudian perancangan tahapan pengujian yang tidak tepat juga dapat menghambat durasi pengujian aplikasi.