

ALAT PENGUJIAN APLIKASI
DENGAN FRAMEWORK SELENIUM DAN ESPRESSO



Oleh
Kelompok 3

Bimo Lucky Prasetyo	6706213001
Safina Sinandari Pratiwi	6706213030
Waldi Defari Kharie	6706210061

Telkom University
Bandung
2023

DAFTAR ISI

DAFTAR ISI.....	2
BAB I.....	3
PENDAHULUAN.....	3
A. Latar Belakang.....	3
B. Rumusan Masalah.....	4
C. Tujuan.....	4
BAB II.....	5
PEMBAHASAN.....	5
1. Selenium IDE.....	5
2. Framework Espresso.....	5
3. Keuntungan Menggunakan Selenium IDE dan Espresso.....	5
4. Langkah – Langkah Penggunaan Selenium IDE.....	7
5. Langkah-langkah penggunaan Espresso.....	10
6. Fitur-Fitur yang dimiliki Selenium IDE.....	13
7. Fitur – Fitur Framework Espresso.....	14
8. Pentingnya Pengujian Aplikasi.....	15
BAB III.....	16
PENUTUP.....	16
A. Kesimpulan.....	16
B. Saran.....	16

BAB I

PENDAHULUAN

A. Latar Belakang

Penggunaan aplikasi Android semakin meluas dan menjadi bagian penting dari kehidupan. Hal ini membuat persaingan di dunia pengembangan aplikasi semakin ketat, dimana pengguna mengharapkan aplikasi yang dapat memberikan pengalaman yang optimal, aman, dan nyaman digunakan. Oleh karena itu, kualitas dan kehandalan aplikasi mobile sangat penting untuk memastikan pengalaman pengguna yang optimal dan mempertahankan loyalitas pengguna terhadap aplikasi tersebut. Untuk mencapai tujuan tersebut, pengujian aplikasi mobile harus dilakukan secara menyeluruh dan terstruktur. Pengujian aplikasi Android, khususnya, memerlukan pengujian yang efektif untuk memastikan aplikasi berfungsi dengan baik dan memenuhi standar kualitas yang diharapkan. Pengujian aplikasi Android juga dapat membantu mengidentifikasi bug, kesalahan, dan kelemahan dalam aplikasi, sehingga memungkinkan pengembang untuk memperbaikinya sebelum aplikasi diluncurkan ke publik.

Pada saat zaman digital seperti ini memiliki keanekaragaman tools yang diperlukan untuk mendeteksi dan menelaah kekurangan dan kelebihan tiap aplikasi dimulai dari tampilan antarmuka aplikasi hingga ke dalam fitur-fitur ataupun code yang dibuat oleh developer. Seperti tools Selenium IDE dan Espresso. Dalam konteks pengembangan aplikasi Android, framework Espresso dan Selenium IDE dapat membantu para pengembang dalam menguji berbagai fitur dan fungsi aplikasi Android, termasuk pengujian UI, pengujian integrasi, dan pengujian fungsional.

Pada makalah ini, akan dijelaskannya tentang pengertian seputar tools yang akan digunakan, kelebihan dan proses penggunaan alat pengujian aplikasi Android dengan menggunakan Selenium IDE dan framework Espresso. Tujuan dibuatnya makalah ini yaitu untuk memberikan pemahaman dan informasi yang lebih baik tentang pengujian aplikasi Android dengan framework Espresso dan membantu para pengembang dalam meningkatkan kualitas aplikasi yang dibangun atau dikembangkannya.

B. Rumusan Masalah

Adapun rumusan masalah dalam makalah ini adalah sebagai berikut :

1. Apa itu tools Selenium IDE ?
2. Apa itu tools Espresso ?
3. Apa keuntungan penggunaannya Espresso dan Selenium IDE dalam pengujian aplikasi ?
4. Bagaimana langkah-langkah melakukan pengujian aplikasi dengan menggunakan Selenium IDE ?
5. Bagaimana langkah-langkah melakukan pengujian aplikasi dengan menggunakan framework Espresso?
6. Apa saja fitur-fitur yang dimiliki oleh Selenium IDE dalam melakukan pengujian aplikasi ?
7. Apa saja fitur-fitur yang dimiliki oleh framework Espresso dalam melakukan pengujian aplikasi ?
8. Apa pentingnya pengujian aplikasi dalam pengembangan aplikasi bagaimana penggunaan framework Espresso dan selenium dapat membantu dalam meningkatkan kualitas dan kehandalan aplikasi?

C. Tujuan

Berdasarkan rumusan masalah di atas, Adapun tujuan dari pembuatan makalah ini adalah sebagai berikut :

1. Memahami framework Espresso dan Selenium IDE dan keuntungan penggunaannya dalam pengujian aplikasi
2. Mengetahui langkah-langkah yang harus dilakukan dalam pengujian menggunakan framework Espresso dan Selenium IDE
3. Mengetahui fitur-fitur atau fungsi-fungsi yang dimiliki framework Espresso dan Selenium IDE dalam melakukan pengujian pada aplikasi
4. Memahami pentingnya pengujian aplikasi dalam pengembangan aplikasi dan memahami penggunaan framework Espresso dan selenium dalam membantu meningkatkan kualitas serta kehandalan aplikasi

BAB II

PEMBAHASAN

1. Selenium IDE

Selenium adalah sebuah alat auto testing yang digunakan untuk mengotomatisasi tes aplikasi web yang dilakukan pada browser. Pengujian selenium bisa dilakukan dengan menggunakan browser apapun, serta script tes bisa ditulis dalam banyak bahasa pemrograman contohnya seperti Java, C#, Python, Ruby PHP, dan lain sebagainya.

Auto testing tools dikhususkan untuk otomatisasi testing di web apps, sehingga dalam pengujian pada aplikasi desktop serta seluler tidak akan bisa dilakukan. Selenium sendiri bersifat open source, dimana tools tersebut bisa diunduh serta digunakan secara gratis. Karena sifatnya yang open source, sehingga para pengguna dapat membagi, memperluas, serta memodifikasi kode yang telah tersedia.

2. Framework Espresso

Framework Espresso adalah salah satu alat testing yang dirancang khusus untuk melakukan pengujian aplikasi Android. Framework ini dikembangkan oleh Google dan dirilis pertama kali pada tahun 2013. Espresso memungkinkan pengembang untuk membuat dan menjalankan tes otomatis pada aplikasi Android, sehingga dapat membantu dalam memastikan kualitas dan kehandalan aplikasi sebelum diluncurkan ke publik.

Espresso menggunakan pendekatan yang berbeda dengan alat pengujian yang lain, seperti Appium atau Robotium, karena Espresso dirancang untuk melakukan tes pada level yang lebih rendah pada aplikasi Android. Espresso adalah framework pengujian yang membantu developer menulis kasus pengujian otomatisasi untuk pengujian antarmuka pengguna (UI), sehingga dapat memastikan bahwa interaksi antara pengguna dan aplikasi berjalan dengan baik. Ini telah dikembangkan oleh Google dan bertujuan untuk menyediakan kerangka kerja yang sederhana namun kuat. Espresso adalah salah satu Kerangka Pengujian Aplikasi Android yang populer, yang banyak digunakan oleh QA. Ini memungkinkan pengujian *black box*, pada saat yang sama, memungkinkan QA untuk menguji fragmen dan komponen individual selama siklus pengembangan.

3. Keuntungan Menggunakan Selenium IDE dan Espresso

a. Selenium IDE

- Perintah Selenese seperti `pen`, `clickAndWait`, `assert`, `verify`, dan lain sebagainya.
- Pencari seperti `id`, `nama`, `xpath`, `pemilih css`, dan lain sebagainya.
- Menjalankan kode JavaScript khusus untuk menggunakan `runScript`
- Mengekspor kasus uji dalam berbagai macam format.
- Untuk membuat tes dengan sedikit maupun tanpa pengetahuan sebelumnya dalam pemrograman.

- Untuk membuat kotak uji sederhana serta ruang uji yang bisa anda ekspor nanti ke RC atau WebDriver.
- Untuk menguji aplikasi web hanya terhadap Firefox serta juga Chrome.

b. Espresso

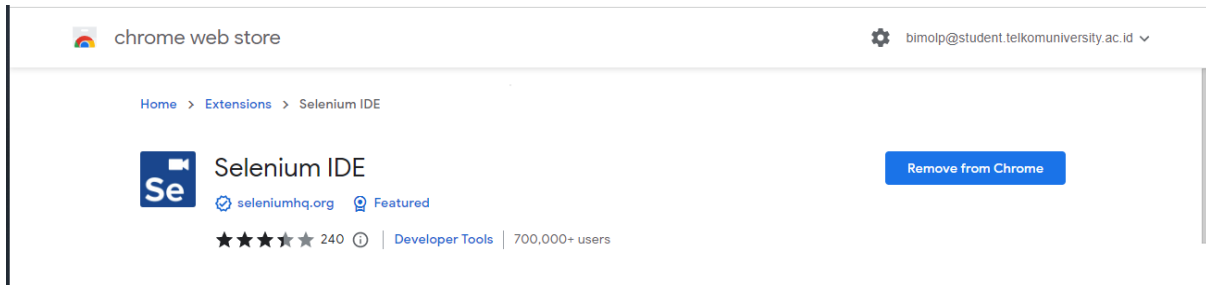
Keuntungan menggunakan framework Espresso untuk pengujian aplikasi Android adalah efisiensi waktu dan biaya. Espresso dapat membantu para pengembang dalam melakukan pengujian secara otomatis, sehingga dapat menghemat waktu dan biaya pengujian yang diperlukan. Selain itu, Espresso juga dapat membantu para pengembang dalam mengidentifikasi bug dan kesalahan dengan cepat, sehingga dapat memperbaikinya sebelum aplikasi diluncurkan ke publik.

- Mudah digunakan: Espresso sangat mudah digunakan dan tidak memerlukan banyak konfigurasi. Dengan Espresso, pengembang dapat melakukan pengujian pada aplikasi dengan mudah dan cepat.
- Stabil dan akurat: Espresso memberikan hasil pengujian yang akurat dan stabil. Framework ini dapat menangani situasi yang kompleks dengan mudah, seperti menemukan elemen di layar, menunggu untuk aksi user, dan memeriksa hasil.
- Mempercepat waktu pengembangan: Dengan pengujian otomatis menggunakan Espresso, pengembang dapat mempercepat waktu pengembangan dan memastikan aplikasi bekerja dengan benar sejak awal. Hal ini membantu untuk mengurangi jumlah kesalahan dan kerusakan di kemudian hari.
- Integrasi dengan CI/CD: Espresso dapat dengan mudah diintegrasikan dengan alat CI/CD, seperti Jenkins, Travis CI, atau CircleCI, sehingga memungkinkan pengembang untuk melakukan pengujian secara otomatis pada setiap perubahan kode.
- Dukungan terhadap testing pada aplikasi yang kompleks: Espresso memiliki kemampuan untuk menguji aplikasi yang kompleks seperti aplikasi dengan banyak aktivitas, fragment, atau tampilan yang berbeda. Framework ini memungkinkan pengujian yang lebih terperinci dan memudahkan dalam menemukan bug pada aplikasi.
- Dapat digunakan dalam proses pengujian yang berulang: Espresso memungkinkan pengembang untuk melakukan pengujian yang berulang pada setiap perubahan kode, dengan mudah dan cepat. Hal ini sangat penting dalam pengembangan aplikasi yang memerlukan pembaruan secara teratur.

4. Langkah – Langkah Penggunaan Selenium IDE

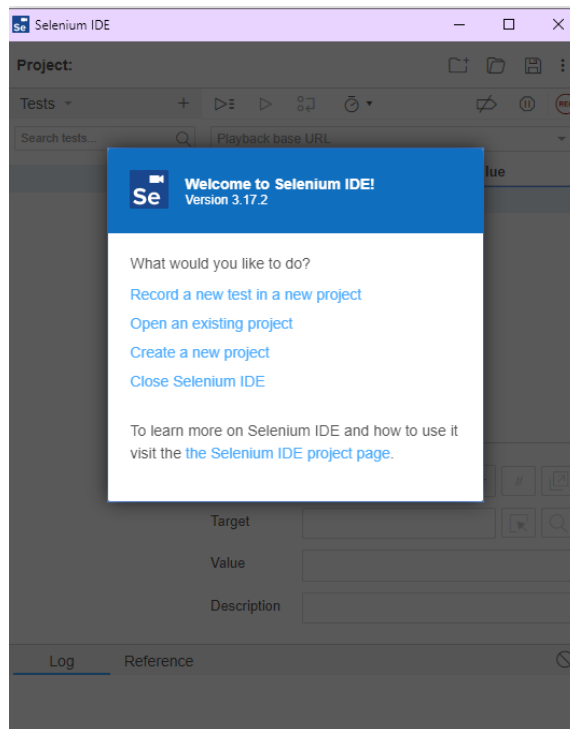
Adapun langkah – langkah dalam menggunakan Selenium IDE sebagai *testing tools* aplikasi adalah sebagai berikut :

1. Pertama tama kita install extension Selenium IDE pada chrome:

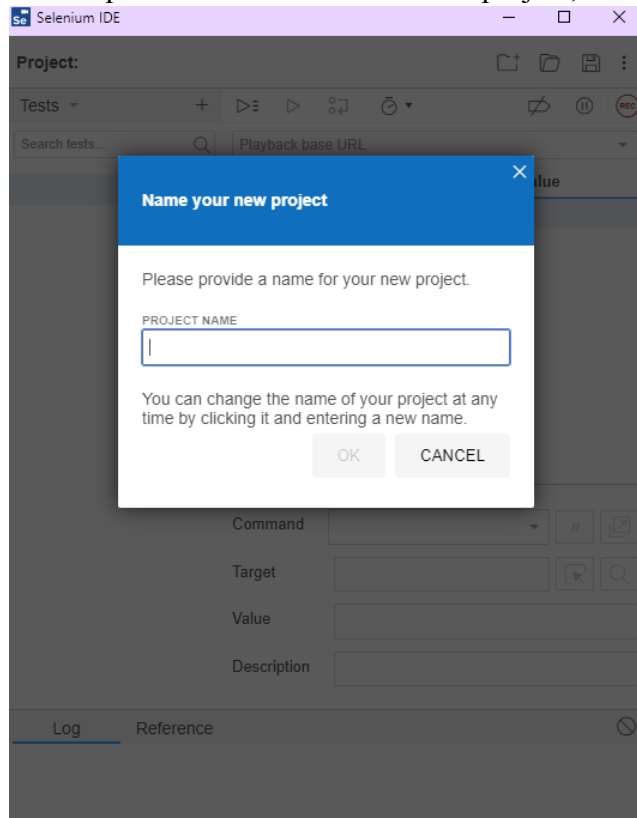


Gambar5. Build gradle

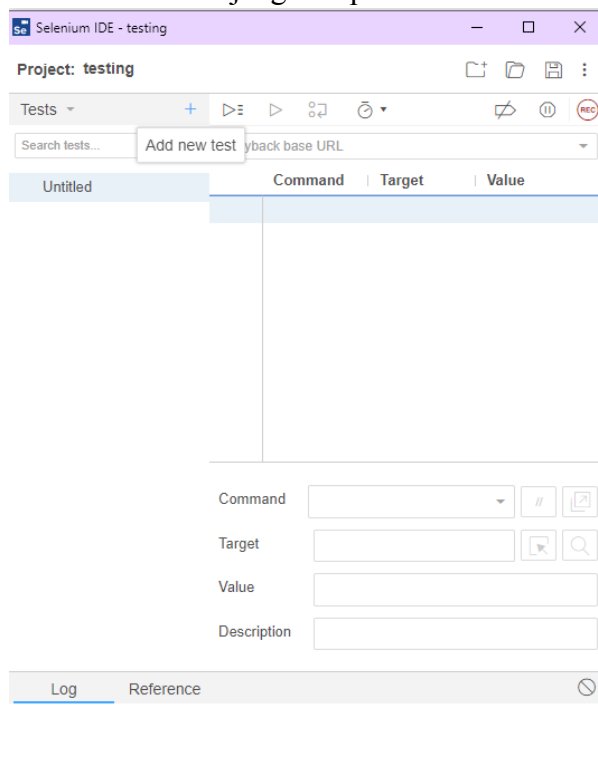
2. Lalu kita buka extension Selenium IDE tersebut jika sudah terinstall



3. Lalu pilih Create a new project, lalu masukan nama projectnya

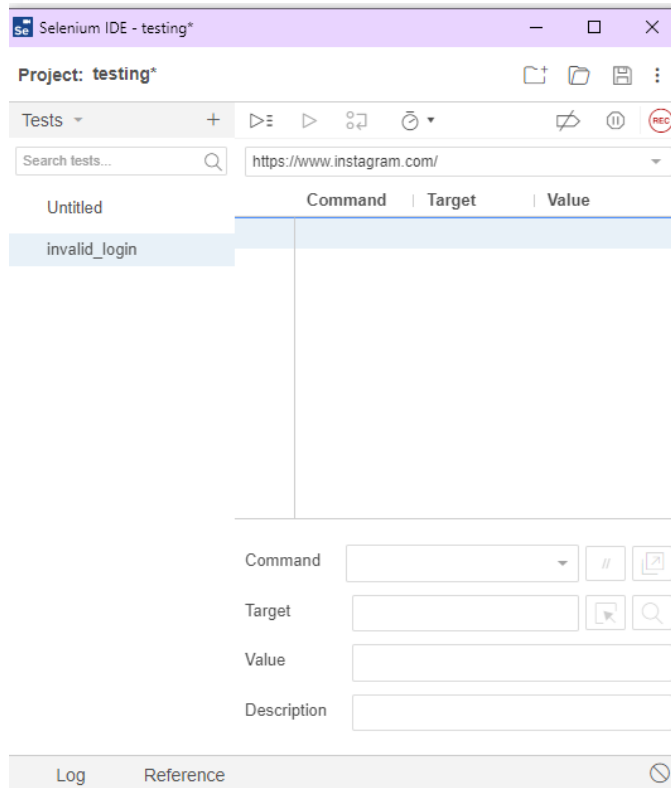


4. Setelah itu kita klik icon tambah (+) untuk membuat test baru dan masukan nama test baru tersebut jangan lupa untuk save file test yang telah dibuat

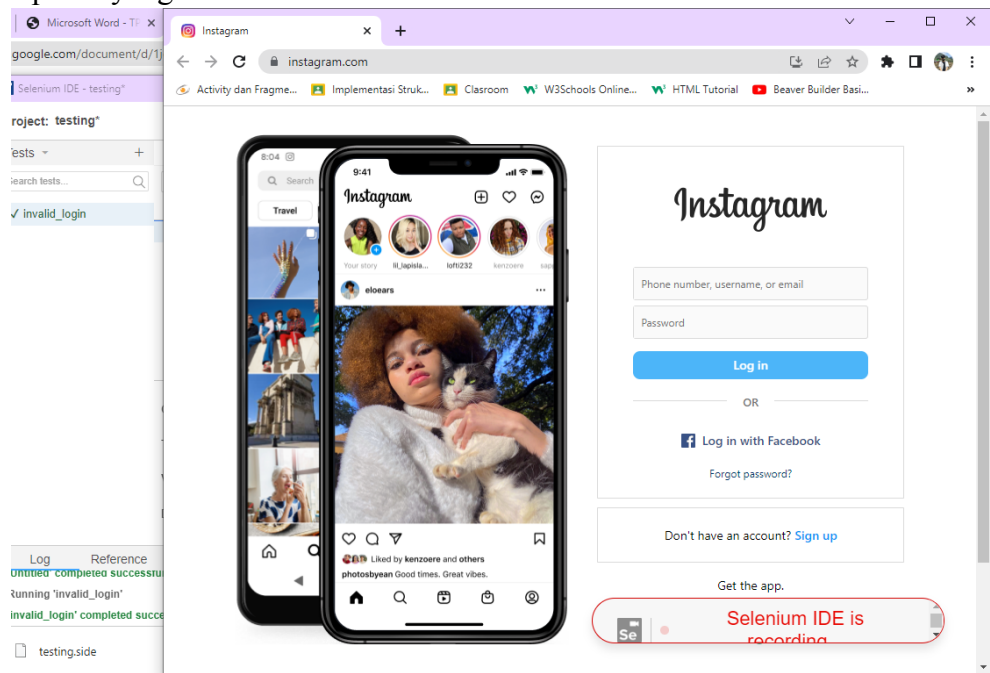


Gambar6. Kode test

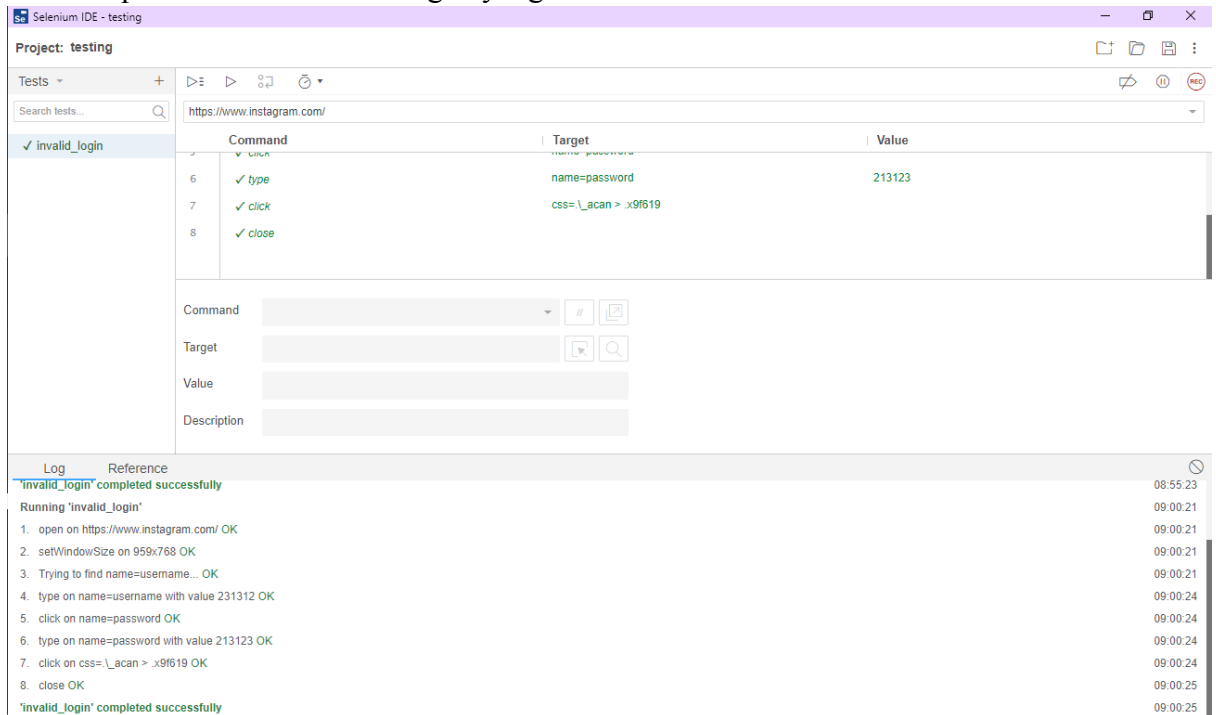
5. Jangan lupa untuk memasukkan link website yang akan kita test, sebagai contoh masukan link website instagram “<https://www.instagram.com/>” lalu coba play untuk mengecek apakah link tersebut valid. disini dilakukan pengecekan login jika invalid



6. klik rec lalu akan dialihkan ke tampilan login instagram lalu coba menginputkan inputan yang tidak valid



7. Setelah selesai memasukkan inputan yang invalid lalu matikan recordnya.jangan lupa save kembali dengan nama yang sama.Setelah di save kita run kembali untuk melihat apakah record sesuai dengan yang telah di lakukan

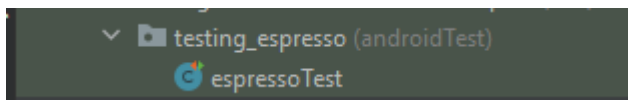


jika sudah berhasil maka tulisan akan berwarna hijau dan bertuliskan succes melakukan testing.Lakukan hal yang sama untuk login dengan valid

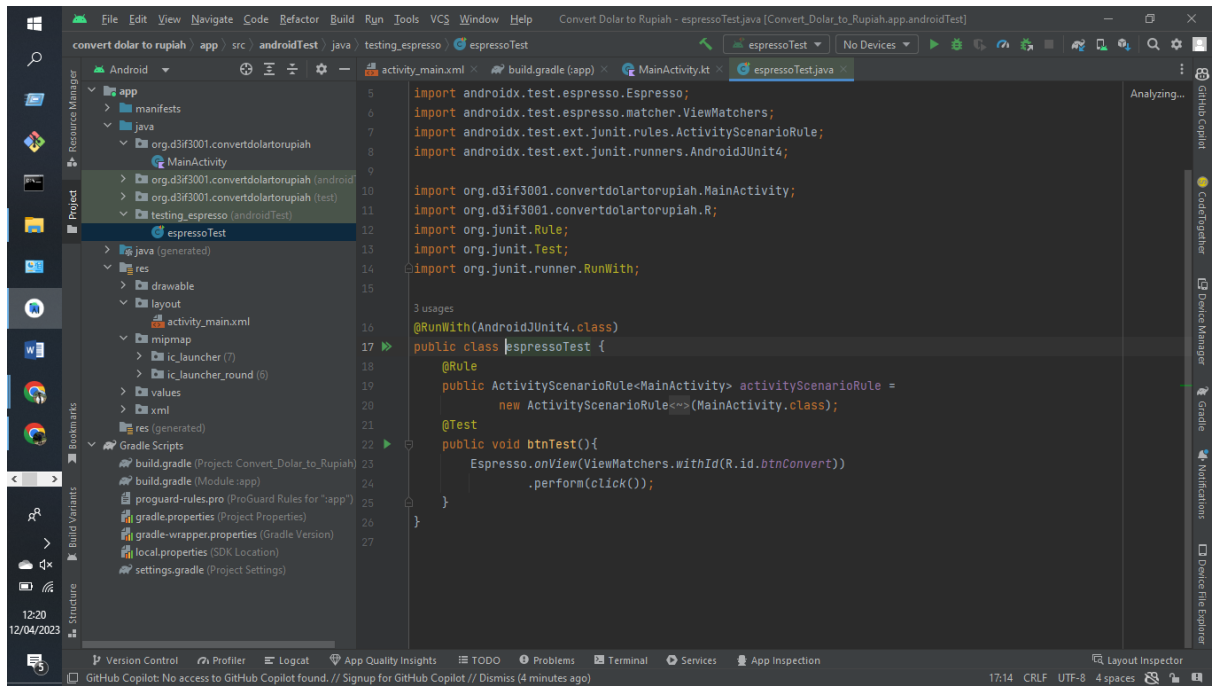
5. Langkah-langkah penggunaan Espresso

Adapun langkah-langkah yang di perlukan

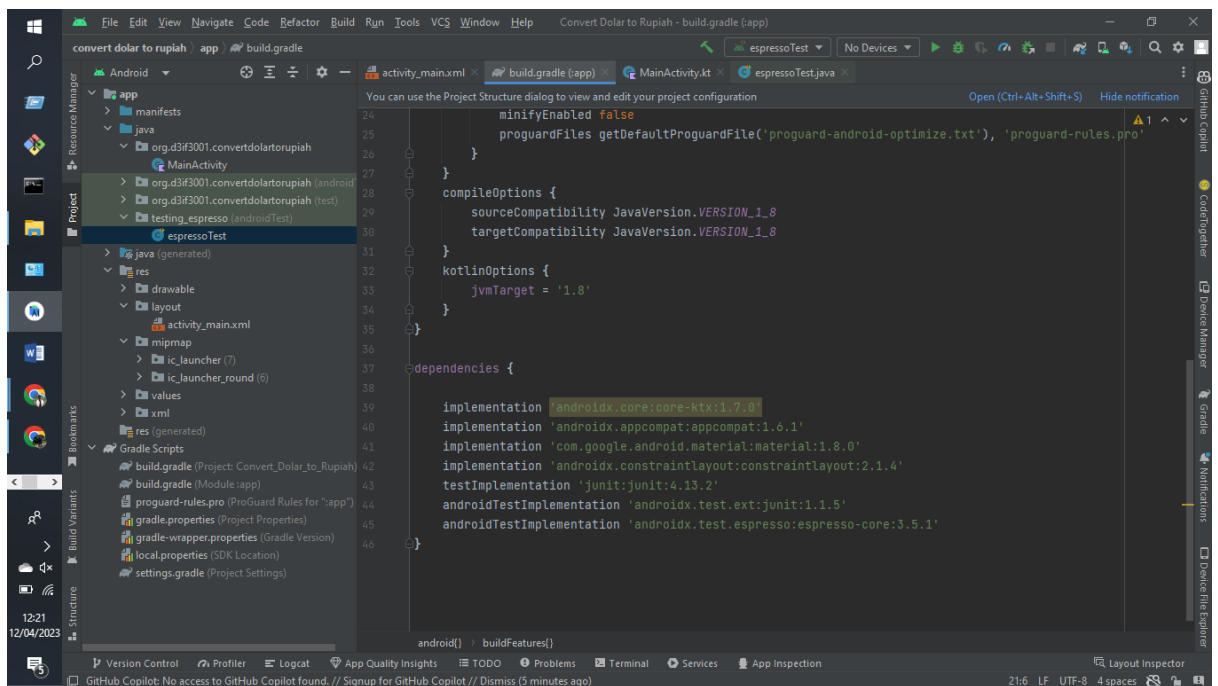
- a. Buatlah package baru beserta filenya



- b. inputkan kode berikut agar menghubungkan file yang akan menjadi target untuk testing dan button yang akan di tes



- c. setelah itu cek pada gradle.app pada bagian dependencies apakah espresso telah terimport otomatis



- d. Pada class espressoTest run classnya saja atau icon segitiga double

```
16 @RunWith(AndroidJUnit4.class)
17 public class EspressoTest {
18     @Rule
19     public ActivityScenarioRule<MainActivity> activityScenarioRule =
20         new ActivityScenarioRule<>>(MainActivity.class);
21     @Test
22     public void btnTest(){
23         Espresso.onView(ViewMatchers.withId(R.id.btnConvert))
24             .perform(click());
25     }
26 }
27
```

- e. Terakhir kita tunggu progressnya apakah test nya berhasil atau gagal

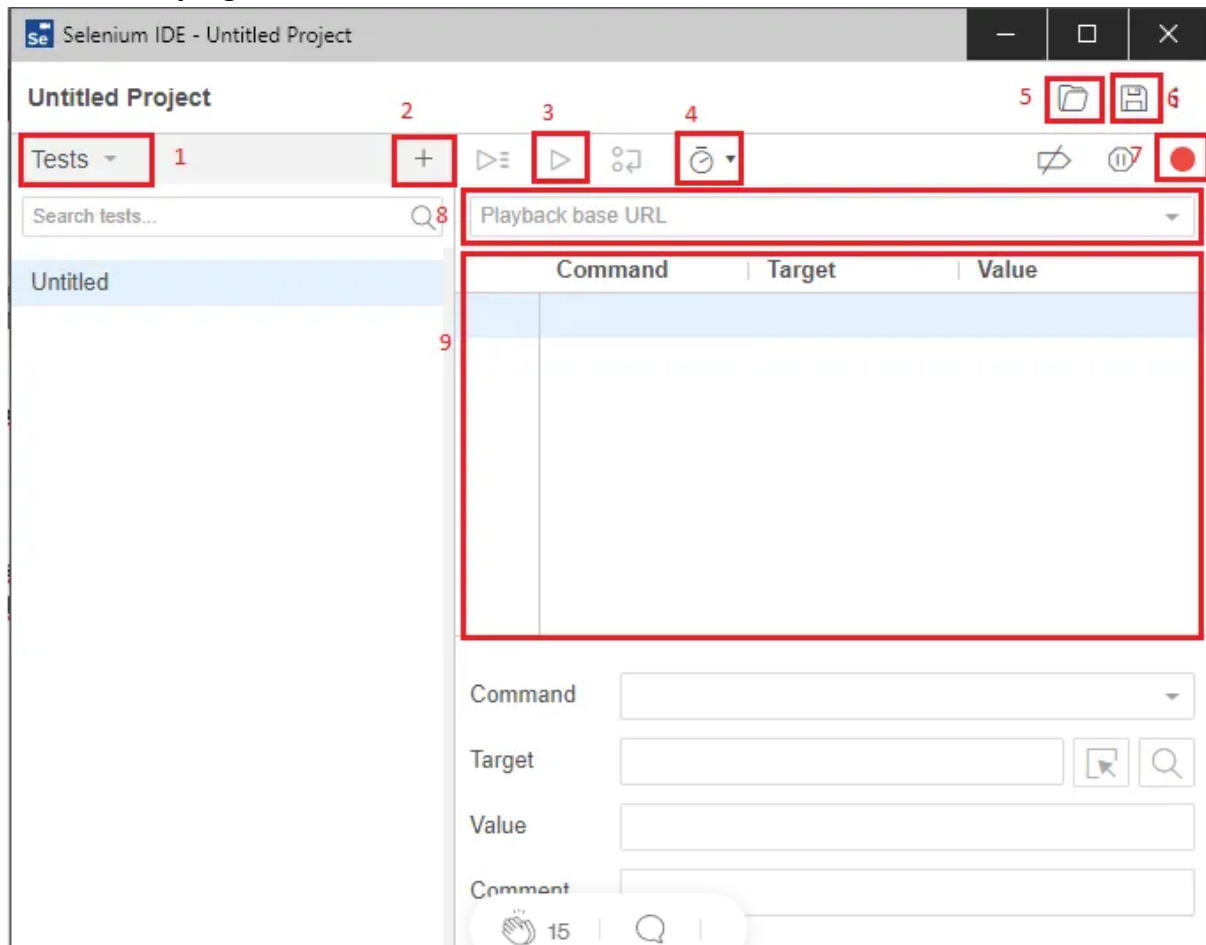
The screenshot shows the Android Studio IDE with the EspressoTest class open. The test results are displayed at the bottom, indicating that the test passed successfully.

Test Results

Test	Duration	Result
Test Results	1s	1/1
espressoTest	1s	1/1
btnTest	1s	✓

Gradle build finished in 10 m 14 s 509 ms

6. Fitur-Fitur yang dimiliki Selenium IDE



- **Test**
Test sendiri ada 3 macam, jika kalian klik maka akan muncul 3 opsi yaitu test, test suites, sama executing. Test suites itu kumpulan dari banyak test, sedangkan executing adalah tes yang sedang dieksekusi. Dan kita akan memulai memulainya dengan membuat Test.
- **Add Test**
Add Test untuk menambah Test Plan
- **Play**
Untuk memulai test yang udah direkam, kalau yang sampingnya play itu untuk memulai test pada test suites
- **Speed**
Mengatur kecepatan test
- **Open**
Membuka file test plan yang sudah dibuat
- **Save**
Menyimpan test
- **Record**
Untuk memulai rekaman test
- **Url**
Kolom untuk memasukkan url yang akan ditest

- Command
Melihat hasil test

7. Fitur – Fitur Framework Espresso

Berikut ini merupakan fitur – fitur atau fungsi – fungsi yang dimiliki framework Espresso :

1. UI testing

Framework Espresso berfokus pada pengujian interaksi antara pengguna dan aplikasi, sehingga memungkinkan pengembang untuk menguji fungsi-fungsi aplikasi pada level tampilan (UI) dengan mudah.

2. Espresso Test Recorder

Merupakan fitur yang memungkinkan pengembang untuk merekam interaksi pengguna dengan aplikasi dan menghasilkan kode pengujian Espresso yang sesuai.

3. Assertion library

Espresso menyediakan library assertion yang kuat dan mudah digunakan, sehingga memungkinkan pengembang untuk mengecek hasil pengujian secara otomatis.

4. AndroidX support

Framework Espresso mendukung AndroidX, yaitu library yang menyediakan komponen Android terbaru, sehingga memudahkan pengembang untuk mengintegrasikan Espresso dengan proyek Android mereka.

5. Simplicity

Espresso memiliki API yang mudah dipahami dan digunakan, sehingga memungkinkan pengembang untuk membuat pengujian dengan cepat dan mudah.

6. Parallel testing

Espresso memungkinkan pengembang untuk menjalankan tes secara paralel pada beberapa perangkat atau emulator, sehingga dapat meningkatkan efisiensi dan kecepatan pengujian.

7. Customization

Espresso memungkinkan pengembang untuk menyesuaikan pengujian sesuai dengan kebutuhan mereka, seperti menentukan waktu tunggu (timeout) atau menambahkan handling error.

8. Integration with other testing frameworks

Espresso dapat diintegrasikan dengan alat pengujian lainnya, seperti JUnit dan Mockito, sehingga memungkinkan pengembang untuk membuat pengujian yang lebih kompleks dan komprehensif.

9. Test coverage reporting

Framework Espresso dapat menghasilkan laporan cakupan pengujian (test coverage), sehingga memungkinkan pengembang untuk memantau kualitas pengujian dan menemukan area yang memerlukan perbaikan.

10. Continuous Integration (CI) support

Espresso mendukung integrasi dengan alat CI seperti Jenkins dan Travis, sehingga memungkinkan pengembang untuk menjalankan pengujian secara otomatis pada setiap perubahan kode.

8. Pentingnya Pengujian Aplikasi

Pengujian aplikasi merupakan proses yang sangat penting dalam pengembangan aplikasi karena dapat membantu dalam meningkatkan kualitas aplikasi. Dampak yang dihasilkan melakukan pengujian aplikasi secara teratur, developer dapat menemukan dan memperbaiki bug atau kesalahan dalam aplikasi sebelum dirilis ke publik. Hal ini dapat meningkatkan kepuasan user dan membantu dalam membangun reputasi aplikasi yang baik.

Framework Espresso dan Selenium IDE dapat membantu dalam meningkatkan kualitas setiap aplikasi yang digunakan dan yang di test

BAB III

PENUTUP

A. Kesimpulan

Dalam pengembangan aplikasi, pengujian aplikasi merupakan aspek yang sangat penting untuk memastikan bahwa aplikasi memiliki kualitas dan kehandalan yang baik sebelum dirilis ke publik. Framework Espresso dan selenium adalah alat pengujian yang sangat berguna untuk melakukan pengujian aplikasi Android dengan cepat dan mudah. Framework ini memiliki fitur-fitur yang dapat membantu pengembang dalam menemukan bug, meningkatkan kualitas kode, memastikan konsistensi aplikasi, dan meningkatkan efisiensi pengujian.

Dalam pembuatan makalah ini, kami telah membahas tentang latar belakang, tujuan, rumusan masalah, fitur-fitur, serta pentingnya pengujian aplikasi dan penggunaan framework Espresso dan selenium dalam meningkatkan kualitas dan kehandalan aplikasi. Supaya dalam Peluncuran aplikasi akan tetap konsisten dan tidak ada kendala. Dari hasil pembahasan tersebut, dapat disimpulkan bahwa pengujian aplikasi menggunakan framework Espresso dan selenium sangat penting dan dapat membantu pengembang dalam meningkatkan kualitas dan meningkatkan efisiensi kehandalan aplikasi.

B. Saran

Berikut adalah beberapa saran yang dapat diberikan terkait dengan pengujian aplikasi menggunakan framework Espresso:

1. Lakukan pengujian aplikasi secara teratur

Pengujian aplikasi harus dilakukan secara teratur untuk memastikan bahwa aplikasi selalu memiliki kualitas dan kehandalan yang baik.

2. Pelajari fitur-fitur dari framework Espresso

Penting bagi pengembang untuk mempelajari fitur-fitur dari framework Espresso agar dapat menggunakannya secara optimal.

3. Gunakan Espresso Test Recorder

Espresso Test Recorder dapat membantu pengembang dalam merekam interaksi pengguna dengan aplikasi dan memastikan bahwa semua fungsi aplikasi berjalan dengan benar.

4. Gunakan Assertion library

Assertion library dapat membantu pengembang dalam menentukan kondisi yang diharapkan pada setiap tes dan memastikan bahwa kode aplikasi berjalan dengan benar.

5. Gunakan AndroidX support

AndroidX support dapat memastikan bahwa aplikasi berjalan dengan benar di berbagai platform.

Berikut beberapa saran terkait pengujian aplikasi menggunakan framework Selenium:

1. **Buatlah rencana pengujian yang baik:** Sebelum memulai pengujian, buatlah rencana yang jelas dan terperinci tentang apa yang akan diuji, cara menguji, dan siapa yang bertanggung jawab untuk pengujian tersebut. Rencana yang baik akan membantu Anda untuk menyelesaikan pengujian lebih efektif dan efisien.
2. **Gunakan Page Object Model (POM):** POM adalah pola desain yang sangat berguna dalam pengujian aplikasi web menggunakan Selenium. Dengan POM, Anda dapat memisahkan kode pengujian dari kode HTML, membuat kode lebih mudah dibaca, dan memudahkan pemeliharaan kode.
3. **Gunakan teknik asertif yang baik:** Asertif adalah cara untuk memastikan bahwa aplikasi berfungsi dengan benar. Selenium menyediakan banyak metode asertif yang berbeda, seperti `assertEquals()`, `assertTrue()`, dan `assertFalse()`. Pastikan Anda menggunakan teknik asertif yang tepat untuk memastikan aplikasi bekerja dengan benar.
4. **Gunakan data uji yang berbeda:** Pastikan untuk menguji aplikasi dengan data uji yang berbeda-beda. Hal ini dapat membantu Anda menemukan bug dan masalah yang mungkin tidak terlihat dengan data uji yang sama.
5. **Gunakan log yang baik:** Selalu gunakan log untuk mencatat aktivitas pengujian dan hasil yang diperoleh. Ini akan sangat membantu jika Anda harus mengulangi pengujian atau memperbaiki masalah yang ditemukan.
6. **Berikan perhatian pada performa pengujian:** Jangan lupa untuk mengukur performa pengujian Anda. Selenium dapat memakan waktu yang cukup lama jika tidak dikonfigurasi dengan benar. Pastikan untuk mengoptimalkan pengujian Anda agar dapat berjalan lebih cepat dan efisien.
7. **Uji selalu pada berbagai platform dan browser:** Pastikan untuk menguji aplikasi Anda pada berbagai platform dan browser, karena aplikasi dapat berperilaku berbeda di setiap platform dan browser. Pastikan bahwa aplikasi bekerja dengan benar pada semua platform dan browser yang Anda uji.

Dengan menerapkan saran-saran ini, Anda dapat mengoptimalkan pengujian aplikasi Anda dan memastikan bahwa aplikasi Anda berfungsi dengan benar dan sesuai dengan kebutuhan pengguna. Dan diharapkan dapat membantu pengembang dalam melakukan pengujian aplikasi menggunakan framework Espresso dan selenium dengan lebih efektif dan efisien.