

**LAPORAN TESTING WEB TWITER
MENGUNAKAN SELERON**

Dosen : Iwan Lesmana, S.Kom., M.kom.



Disusun oleh:

NAMA : Handy Nanda Fachrizal

NIM : 20220810002

KELAS : TINFC-2022-03

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS KUNINGAN**

2023

Daftar Isi

Pendahuluan.....	3
Deskripsi Aplikasi	3
Pemilihan Tool Automation	3
Skenario Pengujian.....	4
Implementasi Test (Cuplikan Script)	4
Hasil Pengujian	5
Permasalahan & Solusi.....	6
Kesimpulan.....	6
Link Video YouTube	6

Pendahuluan

Pengujian perangkat lunak merupakan tahapan penting dalam proses pengembangan sistem untuk memastikan bahwa aplikasi bekerja sesuai dengan fungsionalitas yang diharapkan dan memiliki tingkat keandalan yang tinggi. Salah satu pendekatan yang digunakan dalam proses ini adalah pengujian otomatis (automation testing), yaitu metode pengujian yang dijalankan secara otomatis dengan memanfaatkan alat bantu tertentu.

Dalam laporan ini, dilakukan pengujian otomatis terhadap beberapa fitur utama pada aplikasi web Twitter (X.com) menggunakan tools Selenium WebDriver. Selenium dipilih karena kemampuannya dalam mengotomatisasi interaksi dengan elemen-elemen antarmuka pengguna (UI) pada browser secara real-time. Fitur-fitur yang diuji mencakup proses login, pencarian konten, dan logout.

Tujuan dari pengujian ini adalah untuk memastikan bahwa fitur-fitur tersebut dapat dijalankan dengan baik dan sesuai dengan ekspektasi pengguna. Selain itu, melalui automation testing, pengujian dapat dilakukan dengan lebih cepat, konsisten, dan efisien dibandingkan pengujian manual.

Deskripsi Aplikasi

Aplikasi yang diuji adalah Twitter/X versi desktop (["https://x.com/login"](https://x.com/login)) Twitter/X merupakan platform media sosial yang memungkinkan pengguna untuk membagikan teks, gambar, video, serta melakukan pencarian dan interaksi sosial. Pengujian ini difokuskan pada:

- Login dengan kredensial benar dan salah
- Autentikasi tambahan (username)
- Pencarian konten ("android developer")
- Logout dari aplikasi

Pemilihan Tool Automation

Tool yang digunakan adalah Selenium WebDriver dengan bahasa pemrograman Python. Alasan pemilihannya:

- Gratis dan open-source
- Mendukung berbagai browser (Chrome, Firefox, dll)
- Cocok untuk pengujian berbasis antarmuka (GUI)
- Kompatibel dengan banyak bahasa pemrograman
- Dapat mengontrol interaksi web secara dinamis dan kompleks

Skenario Pengujian

1. Skenario 1: Login gagal dengan password yang salah
2. Skenario 2: Login berhasil dengan email dan password benar (termasuk autentikasi username)
3. Skenario 2b: Melakukan pencarian "android developer" di kolom search
4. Skenario 3: Logout dari akun Twitter

Implementasi Test (Cuplikan Script)

```
username_input = WebDriverWait(driver, 20).until(
    EC.presence_of_element_located((By.XPATH, "//input[@name='text']"))
)
print(f"INFO: Memasukkan username: {username}")
username_input.send_keys(username)

next_button = WebDriverWait(driver, 10).until(
    EC.element_to_be_clickable((By.XPATH, "//span[contains(text(), 'Next')]"))
)
next_button.click()
print("INFO: Mengklik tombol 'Next'.")

password_input = WebDriverWait(driver, 20).until(
    EC.presence_of_element_located((By.XPATH, "//input[@name='password']"))
)
print("INFO: Memasukkan password...")
password_input.send_keys(password)

login_button = WebDriverWait(driver, 10).until(
    EC.element_to_be_clickable((By.XPATH, "//button[@data-testid='LoginForm_Login_Button']"))
)
login_button.click()
print("INFO: Mengklik tombol 'Log in'.")
return True
except TimeoutException as e:
    print(f"FAIL: Elemen login tidak ditemukan atau tidak bisa diklik. Error: {e}")
    return False
```

Hasil Pengujian

- Skenario 1: Gagal login berhasil dideteksi dan menampilkan pesan error
- Skenario 2: Login berhasil setelah input email, autentikasi username, dan password
- Skenario 2b: Kolom pencarian ditemukan, dan hasil pencarian "android developer" muncul
- Skenario 3: Logout berhasil dan pengguna dikembalikan ke halaman login

Log hasil print menunjukkan semua proses berhasil dilalui dengan PASS.

```
PS D:\TUGAS PRAKTIKUM HANDY smt(6)\tugas_testing\Tugas_Testing_Aplikasi> python testing_twitter.py

DevTools listening on ws://127.0.0.1:49233/devtools/browser/43b61e20-fcb4-4d30-9b91-79Fde4baaad9

=====
INFO: Memulai Skenario 1: Login Gagal (Password Salah)
=====
INFO: Membuka/Me-reset halaman login X.com.
INFO: Memasukkan username: hndynnd
INFO: Mengklik tombol 'Next'.
INFO: Memasukkan password...
INFO: Mengklik tombol 'Log in'.
PASS: Validasi berhasil. Pesan error ditemukan: 'Wrong password! gj175163503092693873:-1751635036769:rpoIq0G9LxTQmCloeJzAlYr:7'

=====
INFO: Memulai Skenario 2: Login Berhasil
=====
WARNING: All log messages before absl::InitializeLog() is called are written to STDERR
I0000 00:00:1751635057.083445 20248 voice_transcription.cc:58] Registering VoiceTranscriptionCapability
[10764:18524:0704/201738.734:ERROR:google_api\gcm\engine\registration_request.cc:291] Registration response error message: DEPRECATED_ENDPOINT
INFO: Membuka/Me-reset halaman login X.com.
INFO: Memasukkan username: hndynnd
INFO: Mengklik tombol 'Next'.
INFO: Memasukkan password...
INFO: Mengklik tombol 'Log in'.
INFO: Menunggu halaman Beranda dimuat (maksimal 30 detik)...
Created TensorFlow lite XNNPACK delegate for CPU.
Attempting to use a delegate that only supports static-sized tensors with a graph that has dynamic-sized tensors (tensor#58 is a dynamic-sized tensor).
PASS: Validasi berhasil. Tombol 'Post' ditemukan, menandakan berhasil login dan berada di halaman Beranda.

=====
INFO: Melanjutkan dengan pencarian 'android developer'
=====
INFO: Kata kunci diketik dan ENTER ditekan.
PASS: Hasil pencarian muncul. Menunggu 5 detik untuk observasi...
[10764:18524:0704/201758.976:ERROR:google_api\gcm\engine\registration_request.cc:291] Registration response error message: DEPRECATED_ENDPOINT

=====
INFO: Memulai Skenario 3: Logout
=====
INFO: Mencari tombol menu akun...
INFO: Tombol menu akun ditemukan. Mengklik...
INFO: Menunggu animasi menu (1 detik)...
INFO: Mencari link/teks 'Keluar' (/logout)...
INFO: Link 'Keluar' ditemukan. Mengklik...
INFO: Mencari tombol konfirmasi 'Keluar' dengan cara paling direct...
INFO: Tombol konfirmasi ditemukan. Mengklik...
INFO: Memvalidasi halaman setelah logout...
FAIL: Gagal melakukan proses logout karena elemen tidak ditemukan atau tidak bisa diklik.

=====
INFO: Semua skenario pengujian telah selesai dijalankan.
=====
INFO: Browser telah ditutup.
PS D:\TUGAS PRAKTIKUM HANDY smt(6)\tugas_testing\Tugas_Testing_Aplikasi> []
```

Permasalahan & Solusi

Permasalahan	Solusi
Autentikasi tambahan muncul setelah login	Ditambahkan input username kedua sebelum password
Lokasi tombol dan input sering berubah	XPath dinamis digunakan, dan menunggu (WebDriverWait) ditambahkan
TimeoutException saat hasil tidak muncul	Timeout ditambah hingga 20–30 detik, dan ditambahkan validasi hasil

Kesimpulan

Berdasarkan hasil pengujian otomatis menggunakan Selenium WebDriver terhadap aplikasi Twitter (X.com) versi desktop, dapat disimpulkan bahwa semua skenario yang dirancang—mulai dari login, pencarian konten, hingga logout—berjalan dengan baik. Pengujian ini juga berhasil menangani proses autentikasi tambahan yang muncul setelah pengguna memasukkan alamat email.

Penerapan automation testing terbukti efektif dalam mempercepat proses pengujian, khususnya untuk pengujian regresi yang berulang. Selain meningkatkan efisiensi waktu, metode ini juga mampu meminimalkan kesalahan manusia yang kerap terjadi pada pengujian manual.

Dengan keberhasilan skenario yang diuji, automation testing ini berpotensi untuk diintegrasikan ke dalam CI/CD pipeline guna mendukung proses pengembangan dan rilis aplikasi yang lebih cepat dan andal di masa mendatang.

Link Video YouTube

<https://youtu.be/ZvaKmIA1M78>