

Report

Nama Penguji : Muhammad Ilham Rajo Sikumbang

Github : <https://github.com/ilham1412>

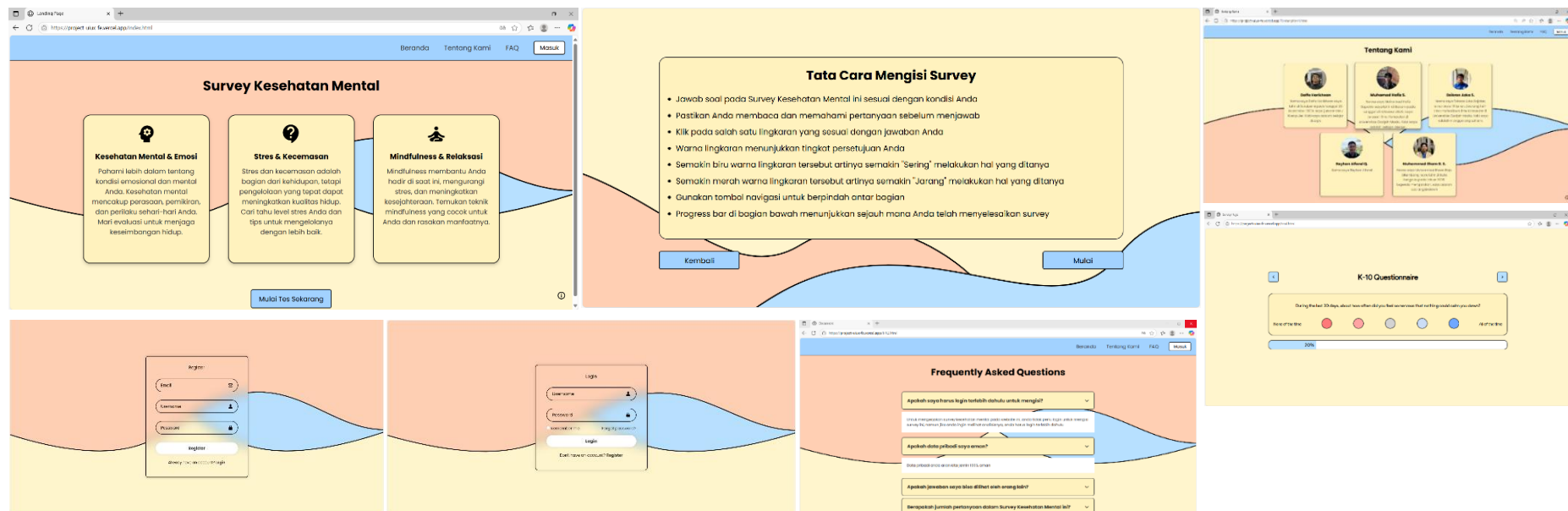
Project 1

Link public : <https://project-uiux-fe.vercel.app/>

Membuat web survey Kesehatan mental, dengan tujuan untuk memastikan Kesehatan mental pengguna selalu terjaga setiap saat. Web ini memungkinkan untuk diintegrasikan dengan layanan sekolah/kampus.

Tech stack yang digunakan :

HTML 5, CSS, JavaScript



Test Case dari web Survei Kesehatan Mental

ID	Fitur	Test Case	Input	Langkah	Expected Result	Status
TC01	Login	Login dengan data valid	email+password	Isi form login lalu klik login	User berhasil masuk dashboard	Fail
TC02	Login	Login dengan data Tidak Valid	email+password	Isi form login lalu klik login	User tidak berhasil masuk ke dashboard	Success + belum terintegrasi
TC03	Sign up	daftar dengan data valid	email>Nama+password	Isi form lalu klik register	User berhasil mendaftar	Pass + belum terintegrasi/tersimpan
TC04	Sign up	Daftar dengan email terdaftar	email>Nama+password	Isi form lalu klik register	User tidak berhasil masuk dashboard	Pass/belum terintegrasi
TC05	Mulai tes	Menekan tombol mulai tes sekarang	Klik	Menekan tombol “mulai tes sekarang”	Masuk ke panduan mengisi survei	Pass
TC06	Tombol kembali	Menekan tombol Kembali di page panduan	Klik	Menekan tombol Kembali saat di page panduan	Kembali ke beranda	Pass
TC07	Tombol Mulai	Menekan tombol mulai di page panduan	Klik	Menekan tombol Mulai	Masuk ke sesi pengisian survei	Pass
TC08	Progress Bar	Mengisi survei	Klik	Menekan salah satu option jawaban	Progress Bar bertambah	Pass
TC09	Tombol Kembali	Menekan tombol Kembali di soal nomor 2 dst	Klik tombol prev	Menekan tombol Kembali	Kembali ke soal sebelumnya	Pass
TC10	Tombol lanjut	Menekan tombol lanjut	Klik tombol next	Menekan tombol lanjut	Lanjut ke soal berikutnya	Pass
TC11	Save Jawaban	Setelah mengisi soal lalu menekan tombol Kembali secara manual dan Kembali ke beranda	Klik prev	Menekan tombol prev Kembali ke beranda	Jawaban user tersimpan dan bisa melanjutkan ke soal terakhir dan progress bar tersimpan	Fail
TC12	Submit Tes	Submit dengan soal telah selesai	Klik submit	Menekan tombol submit	Menampilkan Hasil tes user	Pass

TC13	Submit Tes	Submit dengan soal belum selesai	Klik submit	Menekan tombol submit	Tidak berpindah dan tidak bisa di submit	Pass
TC14	Tombol Selesai	Klik tombol selesai	Klik selesai	Menekan tombol selesai	Ke next page	Pass
TC15	Mengisi Feedback	Klik tombol mengisi feedback	Klik kirim feedback	Menekan tombol kirim feedback	Masuk ke g-form pengisian	Pass
TC16	Kembali ke beranda	Klik beranda	Klik tombol beranda	Menekan tombol beranda	Kembali ke halaman awal	Pass

Test otomatis menggunakan Selenium

No	Nama Pengujian	Deskripsi	Hasil Uji
1	Cek Judul Halaman	Membuka halaman dan mencetak judul tab browser	Pass : Landing Page
2	Page Load Strategy (Eager)	Mengatur strategi load menjadi "eager" dan memverifikasi pemuatan halaman	Pass : Loaded with strategy: eager
3	Pengaturan Timeout	Menerapkan timeout pada load halaman, script, dan elemen	Pass : Timeouts set and page loaded
4	Akses ke Situs dengan Sertifikat Tidak Aman	Mengatur agar sertifikat tidak aman tetap diterima oleh browser	Pass : Accessed insecure site

Masalah yang terdeteksi

Tipe Masalah	Pesan Error	Dampak
SSL Handshake Failure	handshake failed; SSL error code 1, net_error - 101	Fail : Tidak menghalangi pengujian
Shared Image GPU Error	SharedImageManager::ProduceSkia: Trying to Produce a Skia representation...	Fail : Hanya warning dari browser internal

Membuat dan menjalankan automated UI testing dengan Selenium dan Python pada project web survey tes Kesehatan mental. Pengujian mencakup verifikasi judul halaman, strategi pemuatan, pengaturan timeout, dan handling sertifikat tidak aman. Diuji menggunakan browser Edge.

Lampiran

The image shows two screenshots of a development environment. The top screenshot displays a Selenium IDE interface with a test case named 'test.py'. The test case includes steps for setting the driver path to 'msedge.exe' and running the driver. The bottom screenshot shows the corresponding Python code for the test case, which uses Selenium WebDriver to initialize the Edge driver and perform a simple test.

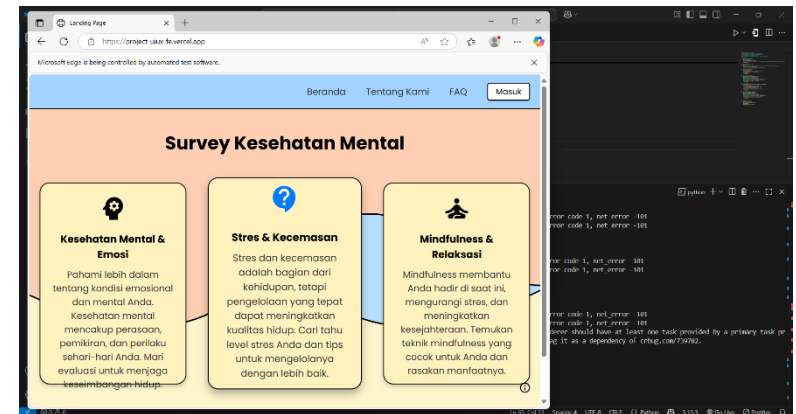
```
def test_accept_insecure_certs():
    driver.quit()

def test_my_website_title():
    test_page_load_strategy('eager')
    test_timeouts()
    test_accept_insecure_certs()
```

```
from selenium.webdriver.edge.service import Service
from selenium.webdriver.edge.options import Options
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
import time

def get_driver():
    options = Options()
    options.add_argument("--no-sandbox")
    options.add_argument("--headless") # opsi tanpa GUI/tidak membuka langsung web nya menggunakan adge
    return options

def test_my_website_title():
    driver = get_driver()
    driver.get("https://project-uke-leverallapp")
    print("Page title is: ", driver.title)
    time.sleep(2)
```



The image shows two screenshots of a development environment. The top screenshot displays a Selenium IDE interface with a test case named 'test.py'. The test case includes steps for setting the driver path to 'msedge.exe' and running the driver. The bottom screenshot shows the corresponding Python code for the test case, which uses Selenium WebDriver to initialize the Edge driver and perform a simple test.

```
def test_accept_insecure_certs():
    driver.quit()

def test_my_website_title():
    test_page_load_strategy('eager')
    test_timeouts()
    test_accept_insecure_certs()
```

```
from selenium.webdriver.edge.service import Service
from selenium.webdriver.edge.options import Options
from selenium.webdriver.common.by import By
from selenium.webdriver.common.action_chains import ActionChains
import time

def get_driver():
    options = Options()
    options.add_argument("--no-sandbox")
    options.add_argument("--headless") # opsi tanpa GUI/tidak membuka langsung web nya menggunakan adge
    return options

def test_my_website_title():
    driver = get_driver()
    driver.get("https://project-uke-leverallapp")
    print("Page title is: ", driver.title)
    time.sleep(2)
```