

Implementasi Pengujian Otomatis End-to-End pada Website Konseling Mahasiswa (SiKosa) Menggunakan Cypress

Muhammad Aidil¹, Muhammad Jibril²

^{1,2}Sistem Informasi, Teknik dan Ilmu Komputer, Universitas Islam Indragiri,

Email: mohammad.aidil09767@gmail.com¹, Jibril.unisi@gmail.com²

ABSTRAK

SiKosa (Sistem Konseling Mahasiswa) merupakan platform web yang dirancang untuk memudahkan mahasiswa dalam melakukan konseling secara daring secara fleksibel. Seiring bertambahnya fitur, kompleksitas sistem meningkat sehingga dibutuhkan pengujian untuk memastikan fungsionalitas berjalan sesuai kebutuhan. Penelitian ini bertujuan menerapkan pengujian otomatis pada website SiKosa dengan menyusun rancangan test case yang sistematis dan mengimplementasikan metode pengujian End-to-End (E2E) menggunakan Cypress pada sisi frontend. Cypress dipilih karena mendukung pengujian yang efisien, konsisten, dan mudah dalam proses debugging. Pengujian dilakukan berdasarkan skenario penggunaan utama untuk memastikan alur interaksi pengguna berfungsi dengan baik. Hasil penelitian menunjukkan bahwa semua test case berhasil dijalankan tanpa kendala, menegaskan bahwa website SiKosa berfungsi secara optimal dan memenuhi kebutuhan serta harapan pengguna dalam menjalankan berbagai fitur utamanya.

Kata Kunci: SiKosa, Pengujian Otomatis, End-to-End Testing, Cypress, Frontend

ABSTRACT

SiKosa (Student Counseling System) is a web platform designed to facilitate flexible online counseling for students. As more features are added, the complexity of the system increases, requiring testing to ensure that its functionality meets requirements. This study aims to implement automated testing on the SiKosa website by developing a systematic test case design and implementing the End-to-End (E2E) testing method using Cypress on the frontend. Cypress was chosen because it supports efficient, consistent, and easy testing in the debugging process. Testing was conducted based on key usage scenarios to ensure that user interaction flows functioned properly. The results of the study show that all test cases were successfully executed without any obstacles, confirming that the SiKosa website functions optimally and meets the needs and expectations of users in running its various main features.

Keywords: SiKosa, Automated Testing, End-to-End Testing, Cypress, Frontend

1 PENDAHULUAN

SiKosa (Sistem Konseling Mahasiswa) adalah sebuah platform berbasis web yang dirancang khusus untuk memfasilitasi kebutuhan konseling mahasiswa. Dengan adanya SiKosa, proses konseling dapat dilakukan secara online, sehingga mahasiswa memiliki keleluasaan lebih dalam mengatur waktu dan tempat untuk berkonsultasi sesuai dengan kondisi pribadi mereka.

Seiring dengan bertambahnya jumlah dan keragaman fitur yang diimplementasikan pada SiKosa, tingkat kompleksitas perangkat lunak juga semakin meningkat. Kondisi ini menuntut adanya proses pengujian yang tepat guna memastikan sistem dapat berjalan sesuai dengan kebutuhan pengguna. Pengujian berperan penting sebagai sarana verifikasi untuk memastikan bahwa setiap komponen sistem yang dikembangkan benar-benar memenuhi kebutuhan pengguna[1]. Lebih jauh, pengujian juga merupakan tahap krusial dalam *Software Development Life Cycle* (SDLC), yang bertujuan menjaga kualitas dan keandalan perangkat lunak secara keseluruhan.

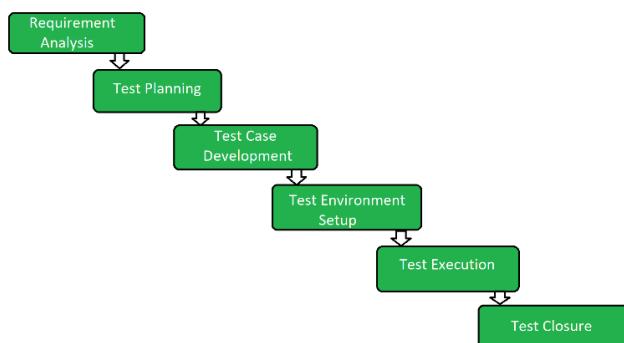
Secara umum, pengujian perangkat lunak dibagi menjadi dua pendekatan, yaitu pengujian manual dan pengujian otomatis. Pengujian manual relatif sederhana untuk dilakukan, tetapi memiliki tingkat kerentanan terhadap kesalahan yang tinggi serta kurang efisien[2]. Untuk itu, dibutuhkan pendekatan pengujian otomatis yang lebih sistematis dan andal. Salah satu pendekatan yang relevan adalah pengujian *End-to-End* (E2E), yaitu teknik pengujian yang memvalidasi alur kerja aplikasi secara menyeluruh dari sudut pandang pengguna, mulai dari tahap awal interaksi hingga akhir[3].

Dalam penerapan pengujian otomatis *End-to-End*, salah satu framework yang banyak digunakan adalah Cypress. Cypress menawarkan berbagai fitur unggulan seperti real-time reloading, dashboard interaktif, serta kemudahan debugging yang memungkinkan proses pengujian dilakukan lebih cepat, menyeluruh, dan efisien. Selain itu, dukungan komunitas yang luas dan dokumentasi yang lengkap menjadikan Cypress sebagai pilihan tepat dalam mendukung pengujian perangkat lunak[4].

Dengan demikian, tujuan dari penelitian ini adalah untuk menerapkan Cypress sebagai alat bantu dalam melaksanakan pengujian otomatis *End-to-End* (E2E) pada sistem SiKosa. Melalui penerapan ini, setiap alur fungsional yang ada pada website dapat diuji secara menyeluruh dan berulang dengan lebih efisien. Proses pengujian otomatis tersebut diharapkan mampu mendeteksi potensi kesalahan lebih cepat, memastikan kesesuaian fitur dengan kebutuhan pengguna, serta meningkatkan tingkat keandalan dan kualitas keseluruhan dari website SiKosa.

2 METODE PENELITIAN

Untuk memastikan bahwa proses pengujian berjalan secara sistematis dan terukur, penelitian ini menggunakan pendekatan *Software Testing Life Cycle* (STLC). STLC merupakan serangkaian tahapan yang sistematis dalam proses pengujian perangkat lunak untuk memastikan bahwa aplikasi yang dikembangkan berfungsi sesuai dengan kebutuhan[5].



Gambar 1 Metode STLC

1. Requirement Analysis

Tahap awal dalam Software Testing Life Cycle (STLC) adalah analisis kebutuhan[6]. Pada tahap ini, fungsionalitas utama dari sistem SiKosa yang menjadi objek pengujian ditentukan. Karena fitur-fitur ini sangat penting dan sering digunakan oleh pengguna, harus dipastikan bahwa mereka berfungsi dengan baik. Fitur yang akan diuji adalah sebagai berikut:

1. Login, memastikan pengguna dapat masuk dengan data yang valid.
2. Registrasi, memastikan akun baru dapat dibuat dengan benar.
3. Logout, memastikan pengguna dapat melakukan logout
4. Fitur konsultasi, memastikan pengguna dapat mengajukan konsultasi dan menerima tanggapan dari psikolog.
5. Fitur chatbot, memastikan pengguna berinteraksi dengan chatbot dan memperoleh jawaban yang relevan sesuai dengan pertanyaan yang diajukan.

2. Test Planning

Tahap ini bertujuan untuk merencanakan strategi pengujian, termasuk pemilihan tools yang digunakan[7]. Beberapa hal yang ditetapkan antara lain:

1. Teknik pengujian yang digunakan adalah End-to-End testing.
2. Tools yang digunakan adalah cypress sebagai framework utama karena mendukung pengujian end-to-end testing secara otomatis dan berbasis browser yang interaktif.
3. Target pengujian adalah validasi apakah sistem SiKosa berfungsi sesuai dengan kebutuhan.

3. Test Case Development

Test case disusun berdasarkan kebutuhan fungsional yang telah diidentifikasi. Test case berfungsi sebagai panduan formal untuk menggambarkan bagaimana sebuah fitur diuji, kondisi awal yang harus dipenuhi, langkah-langkah pengujian yang dilakukan, serta hasil yang diharapkan[8]. Setiap test case memuat:

1. ID test case.
2. Fitur
3. Deskripsi pengujian.
4. Input atau langkah yang dilakukan.
5. Output atau hasil yang diharapkan.

4. Test Environment Setup

Tahap ini berfokus pada penyiapan lingkungan pengujian otomatis agar skenario yang telah dibuat dapat dieksekusi menggunakan Cypress[9]. Langkah-langkah yang dilakukan pada tahap ini meliputi:

1. Instalasi Cypress

Cypress dipasang sebagai dependensi pengujian untuk memastikan dapat menjalankan pengujian dengan konsisten.

2. Konfigurasi file Cypress

Hal ini bertujuan agar setiap test case dapat memanggil halaman yang sesuai tanpa menuliskan URL penuh secara berulang.

3. Penyusunan struktur folder pengujian

Misalnya folder cypress/e2e yang berisi file pengujian berdasarkan fitur. Struktur ini membantu pengelolaan test case agar mudah dipelihara.

4. Persiapan dataset uji

Yaitu data dummy seperti akun mahasiswa dan psikolog. Dataset ini digunakan untuk menguji berbagai kondisi, seperti akun valid, akun duplikat, atau input kosong, sehingga hasil pengujian konsisten.

5. Test Execution

Pada tahap ini, pengujian dilakukan sesuai dengan kasus yang sudah dirancang[10]. Cypress mengeksekusi semua test case melalui GUI dan Command Line Interface (CLI). Hasil dari setiap eksekusi direkam secara otomatis oleh Cypress, yang mencakup:

- Status pengujian, menunjukkan apakah pengujian berhasil (pass) atau gagal (fail).
- Screenshot halaman, Cypress secara otomatis mengambil gambar halaman ketika terjadi kegagalan pengujian, yang memudahkan analisis bug.
- Error log, menunjukkan detail error, seperti elemen tidak ditemukan atau validasi gagal.

6. Test Closure / Evaluation

Tahap ini melibatkan evaluasi hasil pengujian untuk mengetahui sejauh mana sistem telah memenuhi kebutuhan pengguna. Laporan hasil pengujian disusun untuk mendokumentasikan hasil uji, temuan bug, dan rekomendasi perbaikan.

3 HASIL DAN PEMBAHASAN

3.1 Test Case Design

Berdasarkan hasil analisis kebutuhan pengguna dan fungsionalitas sistem, disusunlah serangkaian test case yang mencakup berbagai skenario penggunaan utama, untuk memastikan setiap fitur pada sistem dapat diuji secara menyeluruh dan hasilnya terdokumentasi dengan jelas.

Tabel 1 test case

ID	Fitur	Deskripsi	Langkah Uji / Input	Hasil yang Diharapkan
TC-001	Login	Login dengan akun valid	Masukkan email dan password terdaftar, klik Login	Pengguna berhasil masuk dan diarahkan ke halaman beranda serta memiliki access token
TC-002		Login dengan menekan tombol Enter	Isi email dan password valid, tekan tombol Enter	Pengguna berhasil login seperti saat menekan tombol Login
TC-003		Login dengan akun tidak valid	Masukkan email dan password salah, klik Login	Muncul pesan error "Kredensial tidak valid"
TC-004		Login dengan kolom email kosong	Biarkan kolom email kosong, isi password, klik Login	Muncul pesan error "Email wajib diisi"
TC-005		Login dengan kolom password kosong	Isi email valid, biarkan kolom password kosong, klik Login	Muncul pesan error "Password wajib diisi"
TC-006		Login dengan format email tidak valid	Masukkan email tanpa format yang benar (misal: user@), isi password, klik Login	Muncul pesan error "Format email tidak valid"
TC-007	Register	Registrasi akun baru	Isi form registrasi dengan data valid, klik Daftar	Akun baru berhasil dibuat dan dapat login
TC-008		Registrasi dengan akun yang sudah ada	Isi form registrasi dengan data yang sudah terdaftar, klik Daftar	Muncul pesan error "Email sudah digunakan"
TC-009		Registrasi dengan data tidak valid	Isi form dengan data yang tidak valid, klik Daftar	Muncul pesan validasi

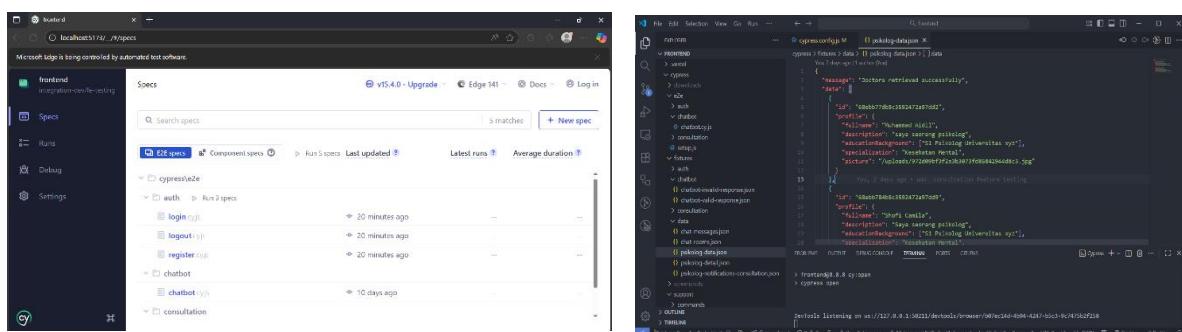
ID	Fitur	Deskripsi	Langkah Uji / Input	Hasil yang Diharapkan
TC-010	Register	Registrasi dengan kolom email kosong	Biarkan kolom email kosong, isi data lainnya, klik register	Muncul pesan error "Email wajib diisi"
TC-011		Registrasi dengan format email tidak valid	Isi kolom email tanpa format yang benar (misal: user@), isi kolom lain dengan benar, klik Daftar	Muncul pesan error "Format email tidak valid"
TC-012		Registrasi dengan kolom nama lengkap kosong	Biarkan kolom nama lengkap kosong, isi data lainnya, klik register	Muncul pesan error "nama lengkap wajib diisi"
TC-013		Registrasi dengan kolom password kosong	Biarkan kolom password kosong, isi data lainnya, klik register	Muncul pesan error "password wajib diisi"
TC-014		Registrasi dengan kolom konfirmasi password kosong	Biarkan kolom konfirmasi password kosong, isi data lainnya, klik register	Muncul pesan error "konfirmasi password wajib diisi"
TC-015		Registrasi dengan password kurang dari 6 karakter	Isi form dengan password pendek (misal: abc123), klik Daftar	Muncul pesan error "Password minimal 6 karakter"
TC-016		Register dengan konfirmasi password tidak sama	Isi form dengan password dan konfirmasi password berbeda, klik Daftar	Muncul pesan error "Konfirmasi password tidak cocok"
TC-017	Logout	Logout	Klik tombol Logout setelah login	Pengguna keluar dan diarahkan ke halaman login
TC-018		Klik tombol Logout berkali-kali dengan cepat	Setelah login, klik tombol Logout berulang kali dengan cepat	Logout tetap hanya dilakukan sekali, pengguna diarahkan ke halaman login tanpa error ganda
TC-019		Logout di tab lain	Login di tab A, buka tab B (halaman sama), lalu Logout di tab A	Tab B otomatis diarahkan ke halaman Login saat mendeteksi token dihapus
TC-020		Logout, lalu tekan back di browser	Login, klik Logout, lalu tekan tombol Back di browser	Pengguna tetap di halaman login, tidak bisa kembali ke halaman private
TC-021		Akses halaman setelah logout	Login, klik Logout, lalu coba akses halaman yang membutuhkan login melalui URL langsung	Pengguna dialihkan kembali ke halaman Login (tidak bisa mengakses halaman private)
TC-022	konsultasi sebagai user	Pengajuan konsultasi baru	Setelah memilih psikolog, pengguna mengklik tombol chat dan kemudian mengklik tombol "Ajukan Konsultasi"	Permintaan konsultasi tercatat, status menunggu

ID	Fitur	Deskripsi	Langkah Uji / Input	Hasil yang Diharapkan
TC-023	konsultasi sebagai user	Tidak bisa mengajukan konsultasi ketika belum login	Klik detail pada card psikolog	Pengguna diarahkan ke halaman login
TC-024		Tidak bisa mengajukan konsultasi ketika status pengajuan masih pending	akses ke halaman untuk mengajukan konsultasi	Tombol ajukan konsultasi tidak muncul
TC-025	konsultasi sebagai psikolog	Menampilkan notifikasi pengajuan konsultasi	login sebagai psikolog, buka notifikasi	menampilkan notifikasi pengajuan yang statusnya pending
TC-026		Konsultasi diterima	Login sebagai psikolog, buka permintaan konsultasi, klik Terima	Status konsultasi berubah jadi diterima
TC-027		Konsultasi ditolak	Login sebagai psikolog, buka permintaan konsultasi, klik Tolak	Status konsultasi berubah jadi ditolak
TC-028		Chat pengguna dan psikolog	Login sebagai psikolog/pengguna, kirim pesan	Pesan terkirim dan diterima oleh kedua pihak secara real-time
TC-029	chatbot	Chat dengan pertanyaan valid	Ketik pertanyaan valid, klik Kirim	Chatbot memberikan jawaban relevan
TC-030		Chat dengan pertanyaan kosong	Klik Kirim tanpa mengetik pertanyaan	Pesan tidak akan terkirim
TC-031		Respons terhadap pertanyaan tidak dikenal	Ketik pertanyaan yang tidak dikenali	Chatbot memberikan jawaban default dan memberikan saran untuk kembali ke topik psikologi

Berdasarkan tabel tersebut, terdapat sebanyak 31 skenario pengujian yang dirancang untuk menguji dan memastikan seluruh fungsionalitas utama pada website SiKosa berjalan sesuai dengan yang diharapkan.

3.2 Test Environment Setup

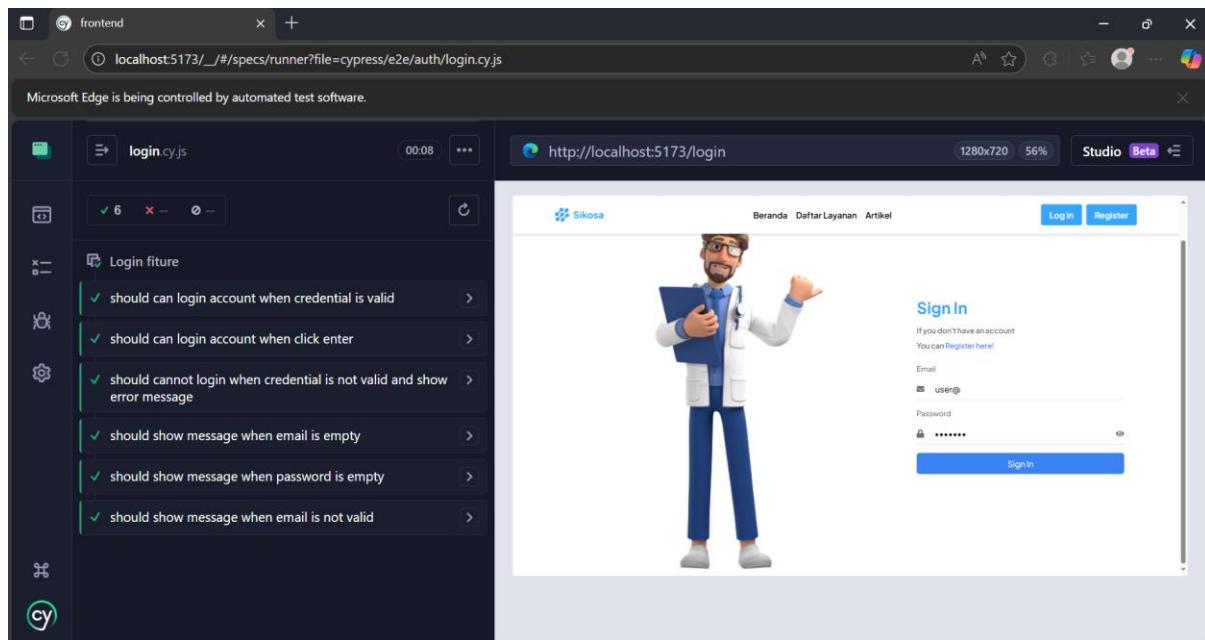
Setelah proses penyusunan test case selesai dilakukan, tahap selanjutnya adalah menyiapkan lingkungan pengujian yang berperan sebagai fondasi utama agar proses pengujian menggunakan Cypress dapat berjalan secara konsisten, stabil, dan otomatis. Pada tahap ini, dilakukan beberapa langkah penting, seperti instalasi serta konfigurasi Cypress melalui file cypress.config.js untuk menyesuaikan pengaturan sesuai kebutuhan proyek. Selain itu, disiapkan pula data uji (fixtures) yang berfungsi untuk mensimulasikan berbagai skenario pengujian dengan kondisi yang menyerupai data nyata, namun tanpa bergantung pada data produksi. Dengan adanya tahap ini, proses pengujian dapat dilakukan secara lebih terkontrol, efisien, dan dapat direproduksi kapan pun dibutuhkan.



Gambar 2 Konfigurasi lingkungan pengujian cypress

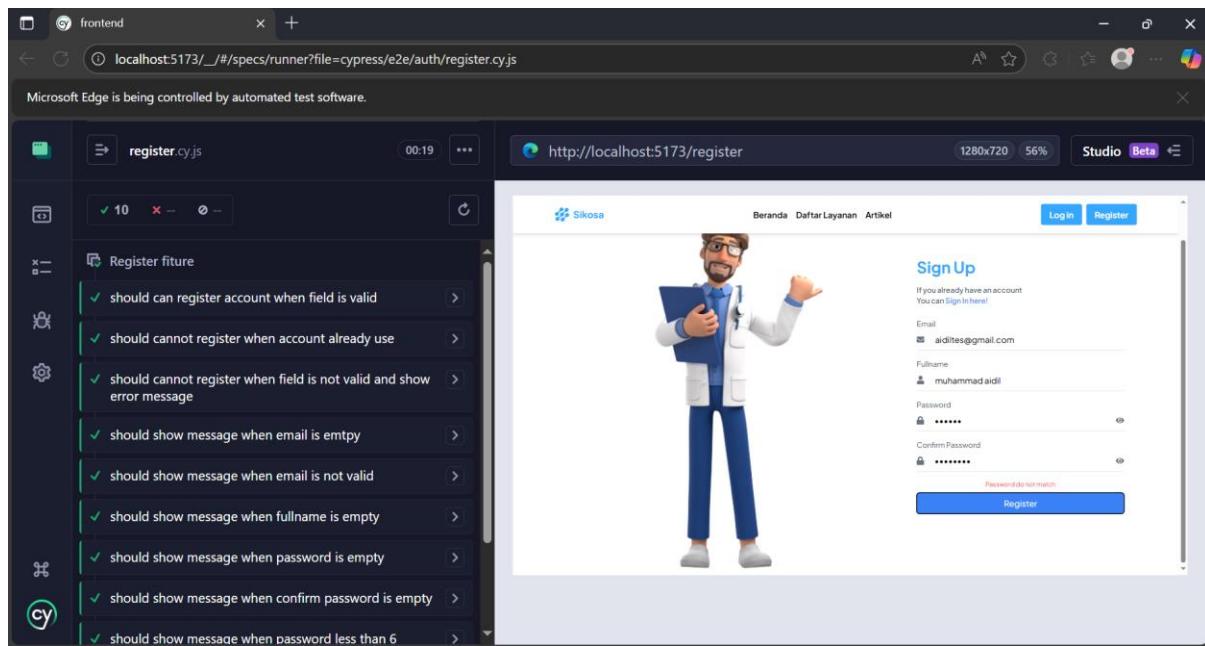
3.3 Test Execution

Merupakan proses pelaksanaan dari seluruh test case yang telah disusun sebelumnya dengan menggunakan Cypress. Pada tahap ini, setiap skenario pengujian dijalankan untuk memverifikasi apakah fungsi-fungsi pada sistem bekerja sesuai dengan yang diharapkan. Selama proses ini, Cypress akan mengeksekusi langkah-langkah yang telah didefinisikan dalam test case, seperti mengunjungi halaman, mengisi formulir, memicu event, dan memverifikasi respons API menggunakan assertion. Selain itu, setiap hasil pengujian, baik yang berhasil maupun yang gagal, akan direkam dalam bentuk screenshot dan video secara otomatis. Data hasil eksekusi tersebut digunakan untuk menganalisis bug atau ketidaksesuaian yang ditemukan pada sistem. Dengan demikian, tahap Test Execution memastikan bahwa seluruh fitur aplikasi telah berfungsi sesuai dengan kebutuhan dan standar kualitas yang telah ditetapkan.



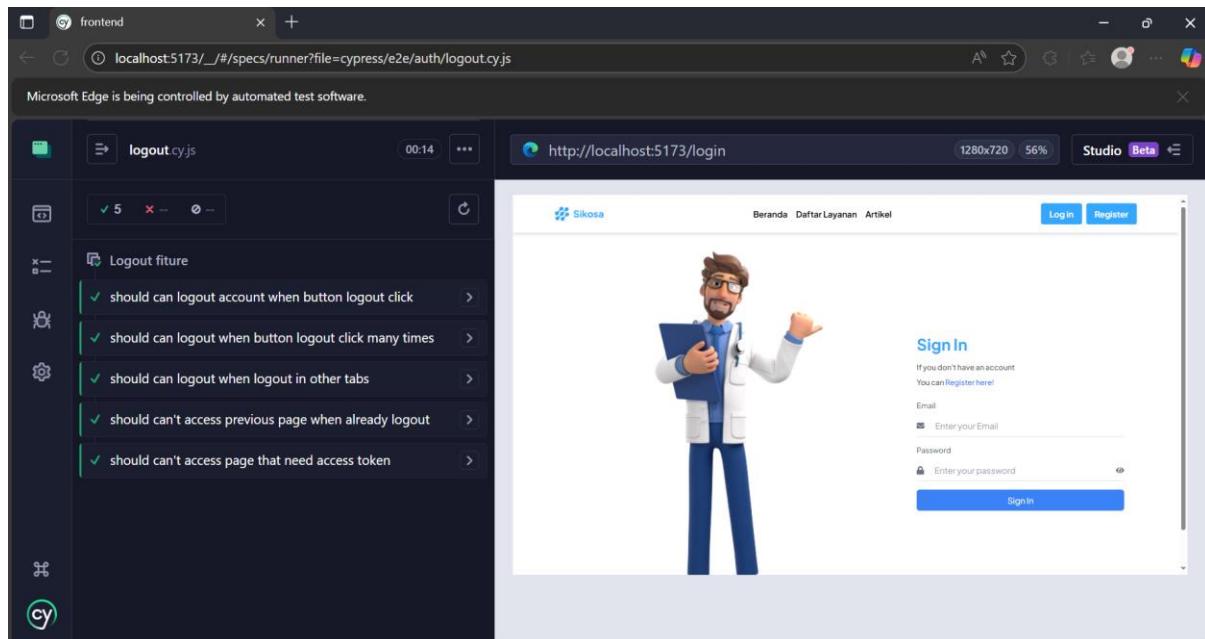
Gambar 3 Hasil test case fitur login

Gambar 3 menunjukkan hasil pengujian pada fitur login. Berdasarkan hasil tersebut, terdapat enam test case yang telah dieksekusi, dan seluruhnya menunjukkan status *pass*, yang menandakan bahwa setiap skenario pengujian pada fitur login berhasil dijalankan sesuai dengan hasil yang diharapkan.



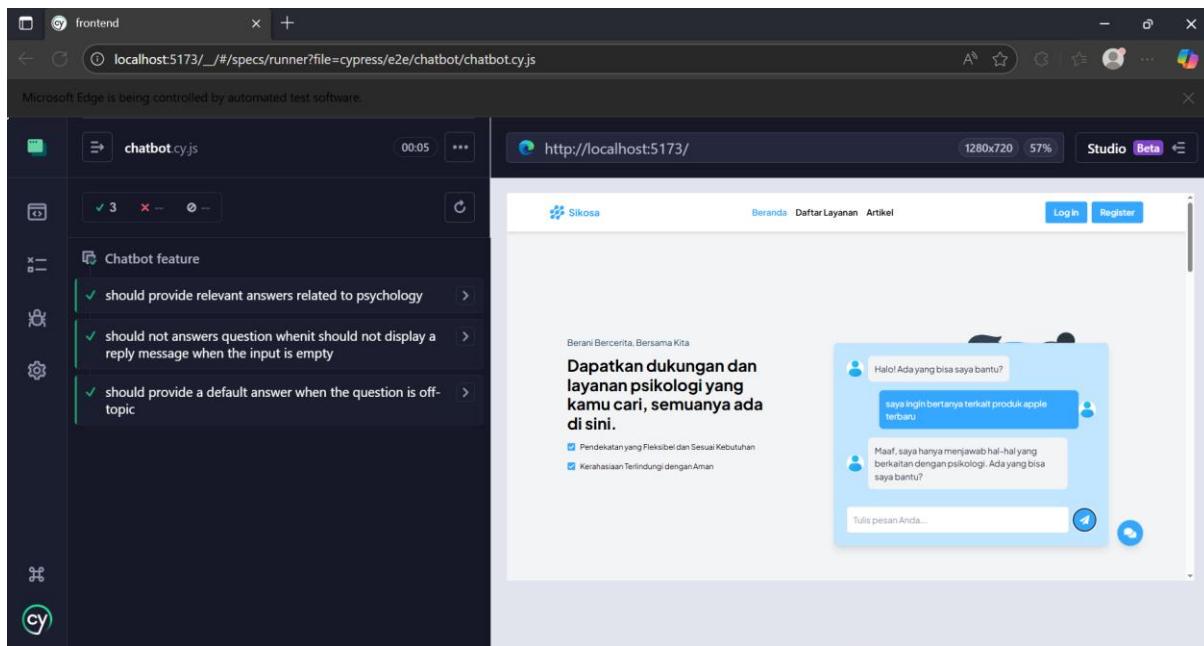
Gambar 4 Hasil test case fitur register

Gambar 4 memperlihatkan hasil pengujian pada fitur registrasi pengguna, di mana terdapat sepuluh skenario pengujian (test case) yang telah dijalankan. Seluruh test case tersebut berhasil dieksekusi dengan status “pass”, yang menunjukkan bahwa setiap fungsi pada proses registrasi berjalan sesuai dengan perilaku yang diharapkan tanpa ditemui kesalahan.



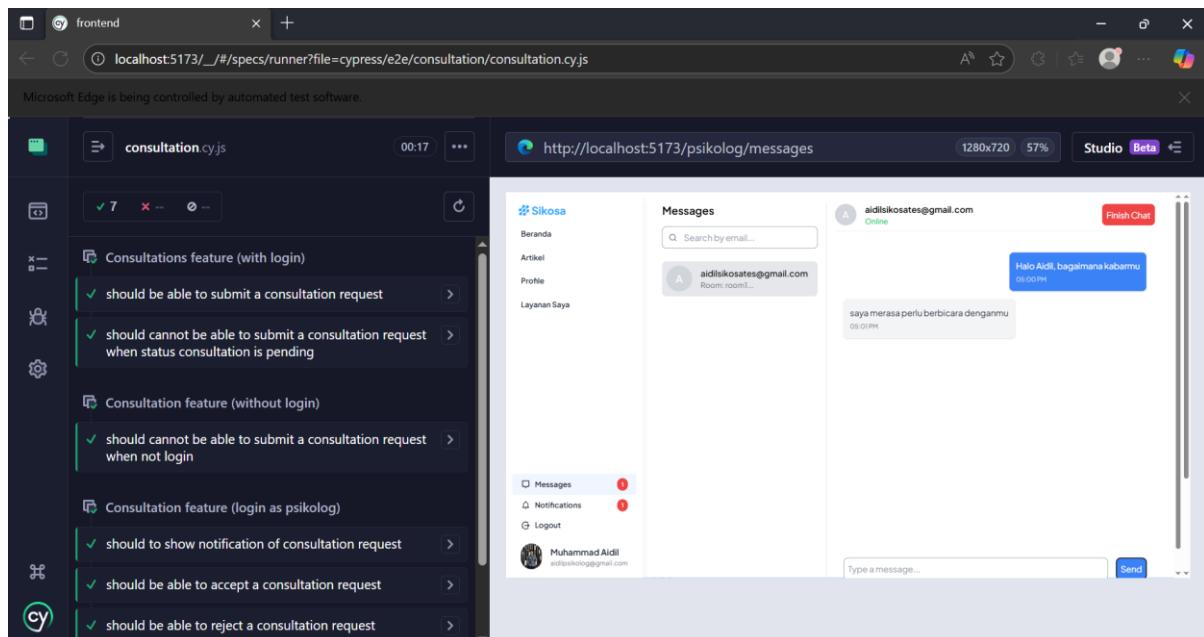
Gambar 5 Hasil test case fitur logout

Gambar 5 menunjukkan hasil pengujian pada fitur logout, yang terdiri dari lima skenario pengujian (test case). Seluruh test case tersebut berhasil dijalankan dengan status “pass”, menandakan bahwa proses logout berfungsi dengan baik sesuai yang diharapkan.



Gambar 6 Hasil test case fitur chatbot

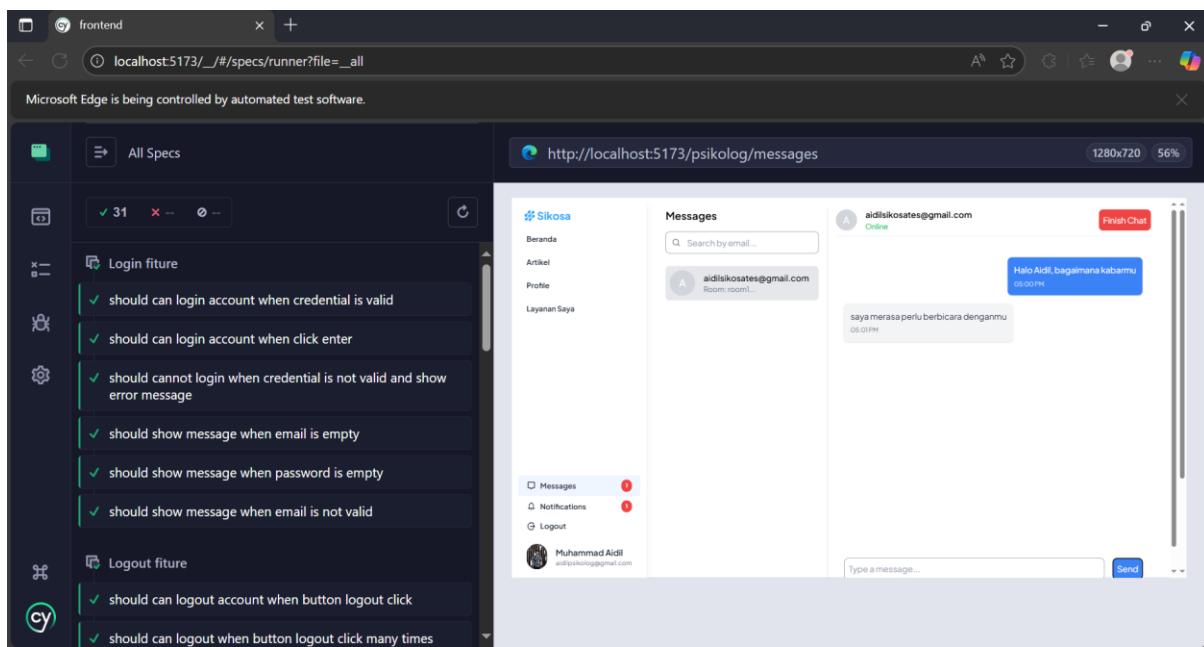
Gambar 6 memperlihatkan hasil pengujian pada fitur chatbot. Dari hasil tersebut diketahui bahwa terdapat tiga test case yang telah dijalankan, dan semuanya memperoleh status *pass*, yang mengindikasikan bahwa setiap skenario pengujian pada fitur chatbot telah berfungsi dengan baik serta menghasilkan keluaran sesuai dengan yang diharapkan.



Gambar 7 Hasil test case fitur konsultasi

Gambar 7 menampilkan hasil pengujian pada fitur konsultasi. Berdasarkan hasil tersebut, terdapat tujuh test case yang telah dieksekusi, dan seluruhnya memperoleh status *pass*. Hal ini menunjukkan bahwa setiap skenario pengujian pada fitur konsultasi berjalan dengan baik dan menghasilkan output yang sesuai dengan ekspektasi sistem.

3.4 Test Closure / Evaluation



Gambar 8 Hasil keseluruhan test case

Gambar 8 menampilkan hasil keseluruhan dari pelaksanaan test case. Berdasarkan proses penyusunan dan pengujian yang telah dilakukan, terdapat 31 skenario pengujian yang mencakup fitur-fitur utama, yaitu login, register, dan logout, serta fitur konsultasi dan chatbot. Seluruh test case pada fitur-fitur tersebut menunjukkan status “pass”, tanpa adanya kegagalan (fail) selama proses pengujian. Hasil ini menunjukkan bahwa setiap fungsi pada fitur berjalan sesuai dengan spesifikasi sistem dan memenuhi kriteria keberhasilan yang telah ditetapkan pada tahap perancangan pengujian. Dengan demikian, dapat disimpulkan bahwa sistem SiKosa telah beroperasi secara stabil, andal, dan sesuai dengan kebutuhan pengguna serta tujuan pengembangannya.

4 KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan terhadap web SiKosa menggunakan metode *End-toEnd Testing* dengan alat bantu Cypress, dapat disimpulkan bahwa seluruh fitur utama dalam sistem telah berfungsi dengan baik sesuai dengan spesifikasi dan kebutuhan yang telah ditetapkan. Pengujian dilakukan terhadap fitur-fitur utama, yaitu login, register, dan logout, serta fitur konsultasi dan chatbot, dengan total 31 test case yang dirancang secara sistematis untuk memverifikasi fungsionalitas aplikasi.

Hasil pengujian menunjukkan bahwa seluruh test case berhasil dieksekusi dengan status *pass*, yang menandakan bahwa sistem mampu memberikan keluaran sesuai dengan perilaku yang diharapkan. Proses pengujian dilakukan melalui beberapa tahapan dalam siklus Software Testing Life Cycle (STLC), meliputi penyusunan test case, penyiapan lingkungan pengujian (test environment setup), pelaksanaan pengujian (test execution), hingga tahap penutupan pengujian (test closure). Penggunaan Cypress sebagai framework pengujian end-to-end memberikan beberapa keunggulan, antara lain kemampuan untuk menjalankan pengujian secara real-time di browser, mendukung visual testing melalui tampilan interaktif, serta pencatatan hasil pengujian dalam bentuk screenshot dan video recording secara otomatis.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem SiKosa telah berfungsi secara optimal dan sesuai dengan kebutuhan yang telah ditetapkan. Seluruh fitur utama berhasil diuji dan menunjukkan hasil yang konsisten dengan harapan, hal ini menegaskan bahwa setiap komponen dalam sistem telah berjalan sesuai rancangan dan mampu mendukung proses kerja aplikasi tanpa ditemukan kegagalan fungsi. Dengan demikian, web SiKosa dapat dinyatakan telah memenuhi seluruh kriteria pengujian yang ditetapkan.

REFERENSI

- [1] Y. F. Achmad and A. Yulfitri, “PENGUJIAN SISTEM PENDUKUNG KEPUTUSAN MENGGUNAKAN BLACK BOX TESTING STUDI KASUS E-WISUDAWAN DI INSTITUT SAINS DAN TEKNOLOGI AL-KAMAL,” 2020.
- [2] J. Sains, T. dan Kesehatan, R. Amaliyah Kamisr, and R. Artikel, “STUDI KOMPARATIF PENGUJIAN MANUAL DAN PENGUJIAN OTOMATIS DENGAN CYPRESS PADA WEBSITE,” vol. 2, no. 2, pp. 354–364, 2025, doi: 10.62335.
- [3] F. D. Wicaksono, “RANCANG BANGUN AUTOMATION TEST JOURNER PADA E-COMMERCE,” 2022.
- [4] B. Prakoso, “PEMANFAATAN CYPRESS UNTUK PENGUJIAN END-TO-END (STUDI KASUS: PENGEMBANGAN APLIKASI INDICAR),” 2023.
- [5] Ruliansyah, Tukino, Baenil Huda, and April Lia Hananto, “Penerapan Software Testing Life Cycle Pada Pengujian Otomatisasi Platform Dzikra,” *CSRID (Computer Science Research and Its Development Journal)*, vol. 15, no. 1, pp. 01–11, May 2023, doi: 10.22303/csrif.15.1.2023.01-11.
- [6] A. Arfan, “Penerapan STLC dalam Pengujian Automation Aplikasi Mobile (Studi kasus: LMS Amikom Center PT.GIT Solution).”
- [7] A. Perbandingan Tools Pengujian Otomatis pada GUI Aplikasi Berbasis WEB Septi Melia, F. Profesio Putra, C. Author, and S. Melia, “Analisis Perbandingan Tools Pengujian Otomatis pada GUI Aplikasi Berbasis WEB,” 2023. [Online]. Available: <https://journal.irpi.or.id/index.php/sentimas>
- [8] A. N. Hasibuan and T. Dirgahayu, “Pengujian dengan Unit Testing dan Test case pada Proyek Pengembangan Modul Manajemen Pengguna.”
- [9] F. W. Azhari and D. F. Suyatno, “Pengujian Otomatis GUI dengan Katalon Studio pada Situs Lowongan Kerja Jobstreet dan Glints,” 2024.
- [10] I. Gusti, N. Putu, D. Dicky Diastama, I. Made Sukarsa, N. Kadek, and A. Wirdiani, “Pengembangan Test Script untuk Load Testing Web dengan metode Software Testing Life Cycle,” 2021.