

Pengujian Crud Link Pada Aplikasi *LinkStack* Berbasis Web Menggunakan *Black Box Testing*

¹Zakiah Mutiara Anggraini*

¹Sistem Informasi, Ilmu Komputer, Universitas Pamulang, Kota Tangerang Selatan, Indonesia

¹zakiahmutiaraanggraini@gmail.com *

Abstract

Bio-link applications such as LinkStack offer flexible and customizable solutions for managing personal links. Testing CRUD (Create, Read, Update, Delete) functions is essential to ensure the application's stability and reliability. This study aims to evaluate the performance of CRUD features in the web-based LinkStack application running within a Docker environment. The testing was conducted manually using a black box testing approach, covering scenarios such as registration, login, link addition, editing, deletion, and displaying the list of links. The results indicate that all CRUD features functioned as expected, with a responsive and user-friendly interface, and no bugs were found during the testing process. In conclusion, LinkStack is a viable platform for managing personal links with customization capabilities. Docker usage contributed to a consistent and stable testing environment. Further automated testing across different environments is recommended to ensure broader reliability.

Keywords: *LinkStack, CRUD, bio-link platform, web application testing, black box testing*

Abstrak

Aplikasi bio-link seperti *LinkStack* menawarkan solusi manajemen tautan pribadi yang fleksibel dan dapat dikustomisasi. Pengujian fungsi CRUD (*Create, Read, Update, Delete*) menjadi hal penting untuk memastikan stabilitas dan keandalan aplikasi tersebut. Penelitian ini bertujuan untuk menguji performa fitur CRUD pada aplikasi *LinkStack* berbasis web yang dijalankan di lingkungan Docker. Metode pengujian yang digunakan adalah *black box testing* secara manual dengan skenario mencakup registrasi, login, penambahan, pengeditan, penghapusan, serta tampilan daftar tautan. Hasil pengujian menunjukkan bahwa seluruh fitur CRUD berjalan dengan baik, antarmuka responsif, dan tidak ditemukan bug selama pengujian. Kesimpulannya, aplikasi *LinkStack* layak digunakan sebagai platform manajemen tautan pribadi, dengan dukungan *Docker* yang memberikan konsistensi dan kestabilan selama pengujian. Pengujian lanjutan secara otomatis dan lintas lingkungan direkomendasikan untuk meningkatkan jangkauan keandalan.

Kata Kunci: *LinkStack, CRUD, Platform Bio-Link, Pengujian Aplikasi Web, Pengujian Black Box*

A. PENDAHULUAN

Dalam era digital yang berkembang pesat, kebutuhan terhadap aplikasi *website* yang mampu mengelola *link* secara efektif semakin meningkat. Seiring dengan transformasi digital yang memberikan banyak kemudahan bagi pelaku usaha, seperti pengurangan biaya operasional dan peningkatan akses pasar yang lebih luas dan interaktif (Amali et al., 2025). Salah satu solusi yang banyak digunakan saat ini adalah *platform bio-link*, yang memungkinkan pengguna membuat satu halaman berisi beberapa tautan penting, seperti media sosial, portofolio, dan produk bisnis, yang dapat memaksimalkan pemasaran digital secara efisien (Makarim & Haryati, 2024). Aplikasi seperti Linktree telah membuktikan betapa pentingnya layanan tersebut dalam mendukung kehadiran digital, terutama di kalangan pelaku usaha mikro, kecil, dan menengah (UMKM) serta konten kreator (F Setiawan, 2023).

Tetapi, tidak semua aplikasi *bio-link* dapat diakses secara gratis atau memberikan fleksibilitas penyesuaian sesuai preferensi pengguna. Oleh karena itu, keberadaan proyek *open-source* seperti *LinkStack* memberikan solusi alternatif yang menarik bagi pengembang dan pengguna yang ingin membangun *platform* sejenis secara mandiri dan lebih bebas dikustomisasi. *LinkStack* adalah aplikasi *open-source* yang menyediakan fitur pembuatan halaman profil dengan daftar tautan, dengan antarmuka yang intuitif dan dapat dijalankan dengan mudah menggunakan *container Docker* (LinkStackOrg, 2024).

Dalam proses pengembangan maupun implementasi aplikasi berbasis *web*, pengujian fitur menjadi tahap yang sangat penting. Pengujian bertujuan untuk memastikan bahwa fitur yang disediakan bekerja sesuai harapan dan bebas dari *bug* (Sihombing et al., 2023). Salah satu fitur yang sering diuji adalah CRUD (*Create, Read, Update, Delete*), yang merupakan komponen fundamental dalam sistem manajemen data. CRUD memastikan bahwa

pengguna dapat menambah, membaca, memperbarui, dan menghapus data dengan aman dan efisien (Bonteanu & Tudose, 2024).

Pada aplikasi *LinkStack*, fitur CRUD diterapkan untuk pengelolaan tautan (*link*). Pengguna dapat menambahkan tautan baru, menampilkan daftar tautan, mengubah tautan yang sudah ada, serta menghapus tautan jika diperlukan (*LinkStackOrg*, 2024). Pengujian fitur-fitur ini berguna untuk mengevaluasi fungsionalitas aplikasi dari sudut pandang pengguna, tanpa perlu memahami detail teknis dari sistem yang diuji (Sakinah et al., 2024). Dengan metode ini, pengembang dapat memastikan bahwa setiap fitur ramah pengguna dan memenuhi standar pengguna akhir.

Pengujian secara manual dilakukan dengan menilai setiap fungsi aplikasi melalui simulasi langsung aktivitas pengguna. Proses ini melibatkan penyusunan skenario uji (*test case*), pelaksanaan langkah-langkah pengujian secara sistematis, serta pencatatan hasil aktual untuk kemudian dibandingkan dengan hasil yang diharapkan. Metode ini sangat berguna dalam skenario yang membutuhkan penilaian subjektif manusia, seperti analisis pengalaman pengguna (UX), dan dinilai lebih efektif dalam menemukan berbagai kesalahan sebelum aplikasi diluncurkan dibandingkan metode otomatis (Prasetyo & Silfianti, 2023).

Docker merupakan salah satu alat penting yang memungkinkan para pengembang *software* dan administrator sistem untuk membuat, mendistribusikan, serta menjalankan aplikasi-aplikasi yang bersifat terdistribusi (Butarbutar et al., 2024). Dalam konteks pengujian, penggunaan Docker menawarkan berbagai keuntungan seperti efisiensi sumber daya, kemudahan dalam mengelola konfigurasi lingkungan, serta konsistensi hasil uji lintas sistem operasi. Oleh sebab itu, pengujian terhadap aplikasi *LinkStack* juga difokuskan pada kemudahan instalasi dan kestabilan sistem saat dijalankan menggunakan Docker, sesuai dengan temuan penelitian yang menyebutkan bahwa teknologi container seperti Docker dapat meningkatkan efisiensi dan stabilitas dalam pengembangan perangkat lunak modern (Fachrudin & Affandi, 2025).

Melalui pengujian ini, diharapkan diperoleh gambaran menyeluruh mengenai performa dan keandalan fitur CRUD pada *LinkStack*, serta kemudahan pengguna dalam menjalankan dan mengoperasikan aplikasi tersebut. Evaluasi dilakukan terhadap beberapa aspek seperti keberhasilan register, login, penambahan tautan, pengeditan, penghapusan, hingga tampilan halaman publik yang dihasilkan. Selain pengujian teknis, pengujian ini juga memberikan nilai tambah dalam pemahaman penggunaan teknologi *open-source* dan *containerization* di lingkungan lokal. Dengan pendekatan ini, pengujian dapat dilakukan secara praktis tanpa memerlukan server eksternal atau konfigurasi kompleks. Hal ini relevan dengan kebutuhan praktikum, penelitian, maupun pengembangan produk digital skala kecil.

Mengingat konteks ini, tujuan penelitian ini adalah untuk menilai secara menyeluruh fitur CRUD pada aplikasi *LinkStack* berbasis *web*, yang dijalankan menggunakan Docker. Hasil dari pengujian ini diharapkan dapat memberikan *insight* mengenai keandalan sistem, kesesuaian fitur terhadap kebutuhan pengguna, dan potensi aplikasi ini sebagai solusi *bio-link* yang fleksibel dan dapat disesuaikan.

B. TEORI DAN METODE

2.1 CRUD (*Create, Read, Update, Delete*)

CRUD (*Create, Read, Update, Delete*) merupakan serangkaian operasi dasar yang digunakan dalam pengelolaan data pada sistem basis data. Siklus CRUD dirancang untuk mendukung proses penyimpanan data secara persisten, yang umumnya diimplementasikan melalui sistem manajemen basis data (Halomoan et al., 2021). Keempat operasi tersebut meliputi penambahan data baru (*CREATE*), pengambilan atau tampilan data (*READ* atau *RETRIEVE*), pemutakhiran data yang sudah ada (*UPDATE*), dan penghapusan data (*DELETE*) merupakan fungsi inti yang membentuk fondasi dalam pengembangan berbagai aplikasi berbasis data (firmansyah et al., 2024).

2.2 Pengujian Manual (Manual Testing)

Pengujian manual merupakan proses evaluasi sistem yang dilakukan secara langsung oleh pengguna atau penguji, dengan mengikuti langkah-langkah pengujian yang telah dirancang sebelumnya untuk menilai kinerja dan fungsionalitas aplikasi (Fauzan et al., 2023). Metode ini sangat bergantung pada ketelitian dan konsistensi penguji karena seluruh proses dilakukan tanpa bantuan alat uji otomatis, sehingga pengamatan dan pencatatan hasil menjadi aspek penting dalam pelaksanaannya (Amalia, 2022). Namun, pendekatan manual ini cenderung memerlukan waktu dan sumber daya yang cukup besar, sehingga kurang efisien jika diterapkan pada pengujian yang bersifat berulang atau berskala besar (Riski & Renanti, 2024).

```
docker run -d -p 8080:80 --name linkstack  
linkstackorg/linkstack
```

2.3 Pengujian Perangkat Lunak (Software Testing)

Pengujian perangkat lunak merupakan tahapan penting dalam proses pengembangan sistem yang bertujuan untuk menjamin kualitas, performa, serta aspek keamanan dari perangkat lunak yang dibangun (Abdillah et al., 2024). Melalui proses ini, pengembang dapat mengidentifikasi kesalahan (*bug*), ketidaksesuaian fungsi, hingga potensi celah keamanan sebelum sistem dirilis ke pengguna. Selama proses pengembangan berlangsung, peningkatan kualitas perangkat lunak dilakukan melalui tahapan pengujian, identifikasi *bug*, dan perbaikan

terhadap bug tersebut hingga perangkat lunak dianggap siap untuk dirilis (Ramadhani & Putro, 2023).

2.4 Black Box Testing

Pengujian yang dilakukan hanya dari luar sistem disebut pengujian “black box”. Pengujian black box adalah jenis pengujian yang berkonsentrasi pada kebutuhan fungsional sistem., berdasarkan spesifikasi kebutuhan sistem (Pratama et al., 2023a). Artinya *Black box testing* tidak perlu memiliki pemahaman terkait dengan pemrograman. Pengujian black box memiliki beragam pendekatan. Di antaranya adalah *equivalence partitioning*, *boundary value analysis*, dan *fuzzing hingga orthogonal array testing* (Pratama et al., 2023b).

2.5 Containerization & Virtualization (Docker)

Containerization dan *Virtualization* merupakan dua pendekatan teknologi yang memungkinkan aplikasi dijalankan dengan efisien dalam lingkungan yang terisolasi. *Virtualization* bekerja dengan membuat mesin virtual yang memiliki sistem operasi tersendiri, sementara *containerization* memanfaatkan *container* yang berbagi kernel dari sistem operasi induk namun tetap menjaga isolasi antar aplikasi (Isron & Putra, 2021). Docker adalah teknologi perangkat lunak yang digunakan untuk mengemas aplikasi ke dalam sebuah lingkungan yang terisolasi, lengkap dengan seluruh dependensi yang dibutuhkan agar aplikasi dapat berjalan dengan konsisten (Riyanto et al., 2022).

2.4 Metode

Penelitian ini menggunakan pendekatan pengujian manual berbasis *black box testing* terhadap aplikasi *LinkStack* untuk mengevaluasi fitur pengelolaan tautan (CRUD). *Source-code* aplikasi *LinkStack* dapat diakses melalui platform Github dengan link <https://github.com/LinkStackOrg/LinkStack>. Aplikasi dijalankan secara lokal menggunakan Docker untuk menjamin konsistensi lingkungan uji. Metode ini bertujuan untuk mengevaluasi sejauh mana aplikasi *LinkStack* dapat memenuhi kebutuhan dasar pengguna dalam menambah, membaca, memperbarui, dan menghapus tautan. Langkah pertama dalam metode ini adalah menyiapkan lingkungan pengujian dengan menjalankan perintah Docker berikut:

Perintah ini akan menarik *image* resmi *LinkStack* dari Docker Hub dan menjalankan aplikasi pada *port* 8080 di *localhost*. Setelah itu, pengguna dapat mengakses aplikasi melalui browser dengan alamat <http://localhost:8080>.

Setelah berhasil masuk ke aplikasi, dilakukan pengujian terhadap beberapa fitur utama menggunakan skenario pengujian (*test case*). Setiap skenario mencakup:

- Fitur yang diuji, seperti menambah tautan baru atau menghapus tautan.
- Langkah pengujian, yang dijelaskan secara berurutan.
- Output yang diharapkan, berdasarkan dokumentasi dan logika sistem.
- Output aktual, yaitu hasil yang muncul saat pengujian dilakukan.

- Status, berupa “pass” jika hasil sesuai harapan, atau “fail” jika terjadi penyimpangan.

Metode ini juga mencatat setiap hasil dalam bentuk tabel pengujian agar dapat dianalisis. Pengujian dilakukan pada satu sesi tanpa jeda, agar kondisi sistem tetap stabil dan hasil lebih representatif. Dengan pendekatan manual ini, pengujian dapat meniru perilaku pengguna sebenarnya, memberikan evaluasi langsung terhadap pengalaman pengguna (*user experience*), dan mengevaluasi keandalan fitur inti dalam aplikasi *LinkStack*.

C. HASIL DAN PEMBAHASAN

3.1 Output teknis

3.1.1 Hardware

Untuk mengoperasikan aplikasi software ini, perangkat keras yang digunakan adalah laptop Asus yang dilengkapi dengan CPU Intel Core i3-6006U yang beroperasi pada kecepatan 2.0 GHz. Parameter berikut meliputi:

Tabel 1. Spesifikasi Perangkat Keras (Hardware)

Jenis	Spesifikasi	Keterangan
Laptop	Prosesor	Intel Core i3-6006U 2.0 GHz
	RAM	8 GB DDR4
	Penyimpanan	512 GB SSD
	Sistem Operasi	Windows 10 Pro 64-bit

3.1.2 Software

Untuk menjalankan sistem ini, perangkat lunak berikut digunakan :

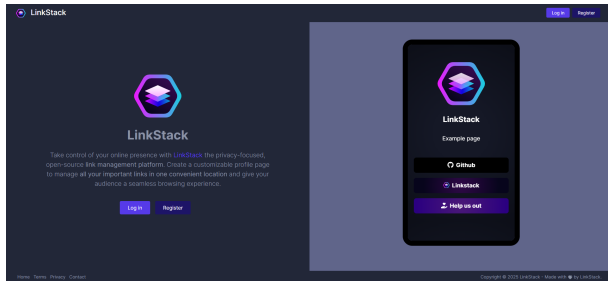
Tabel 2. Spesifikasi Perangkat Lunak (Software)

Jenis	Spesifikasi	Keterangan
Docker Software	Docker	Docker Desktop v24.0.7
	Browser	Google Chrome v125.0
	<i>LinkStack</i> Image	<i>LinkStackorg/LinkStack:latest</i> (Docker Hub)
	Akses Aplikasi	http://localhost:8080

3.2 Hasil Pengujian Fitur Aplikasi

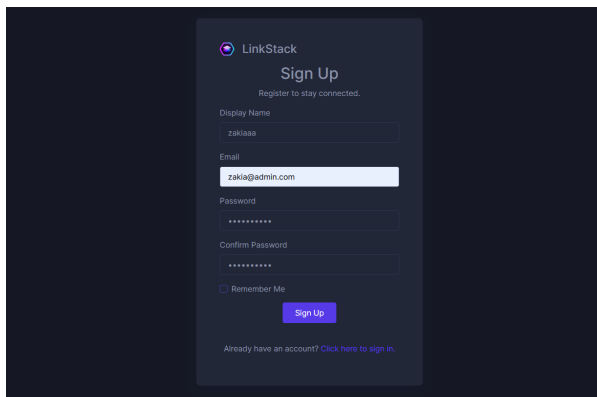
Pengujian dilakukan berdasarkan fitur utama yang tersedia dalam *LinkStack*, yaitu fitur CRUD (*Create, Read, Update, Delete*) tautan. Berikut adalah hasil pengujian yang dilakukan:

1) Fitur Register



Gambar 1. Tampilan Awal

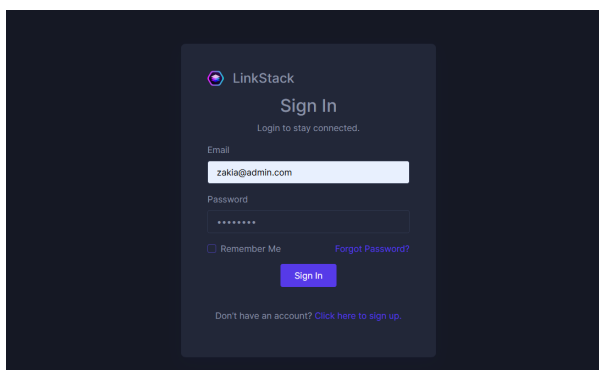
Gambar 1 menunjukkan tampilan awal saat aplikasi *LinkStack* pertama kali diakses melalui browser. Pada halaman utama ini, pengguna langsung disambut dengan antarmuka yang bersih dan intuitif. Terdapat dua pilihan utama bagi pengguna, yaitu Login untuk pengguna yang sudah memiliki akun, dan Register bagi pengguna baru yang ingin membuat akun.



Gambar 2. Tampilan Register

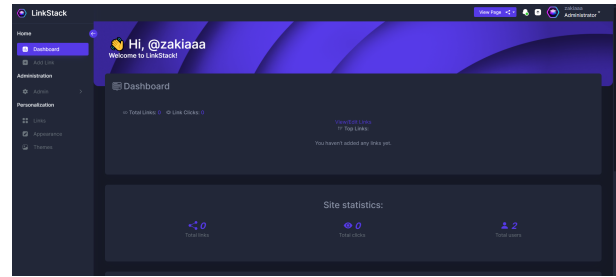
Gambar 2 menampilkan fitur Register untuk Pengguna baru dapat mendaftarkan akun melalui halaman registrasi dengan mengisi username, email, dan password. Setelah registrasi berhasil, pengguna langsung diarahkan ke halaman login.

2) Fitur Login



Gambar 3. Tampilan Login

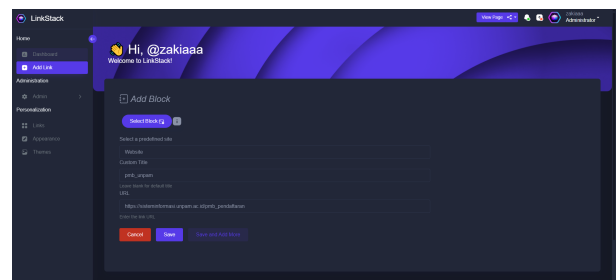
Gambar 3 memperlihatkan halaman login *LinkStack*, di mana pengguna diminta memasukkan email dan password yang valid. Tampilan ini sederhana dan responsif, memudahkan pengguna untuk mengakses akun mereka dengan cepat.



Gambar 4. Tampilan Dashboard

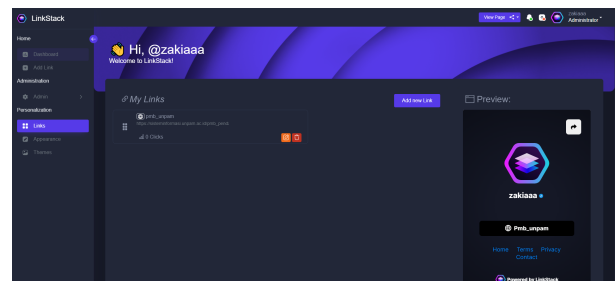
Setelah login berhasil, pengguna diarahkan ke dashboard utama yang menampilkan daftar tautan yang telah dibuat, serta opsi untuk menambah, mengedit, atau menghapus tautan. Tampilan Dashboard dapat dilihat pada Gambar 4.

3) Menambah Tautan (Create)



Gambar 5. Tampilan Add Link

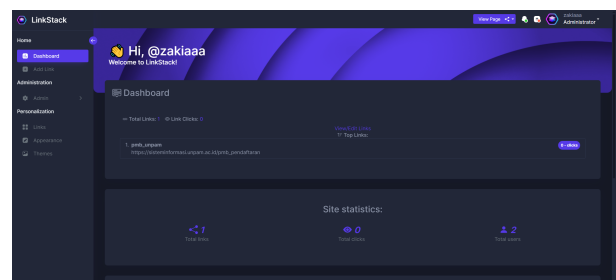
Gambar 5 menunjukkan form untuk menambahkan tautan baru. Pengguna dapat mengisi pilihan *site*, nama tautan (title) dan URL tujuan, lalu menekan tombol untuk menyimpan tautan tersebut.



Gambar 6. Tampilan Data Link

Gambar 6 menampilkan data tautan yang sudah berhasil ditambahkan. Informasi yang ditampilkan meliputi nama tautan, URL, dan tombol aksi seperti edit dan hapus.

4) Melihat Daftar Tautan (Read)

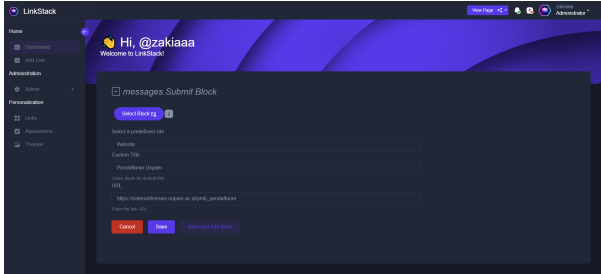


Gambar 7. Tampilan Daftar Link di Dashboard

Tautan yang telah ditambahkan dapat dilihat secara langsung di halaman dashboard pada Gambar 7. Tautan

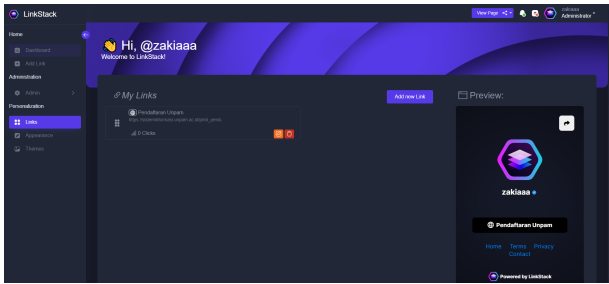
aktif dan dapat diakses oleh pengunjung halaman. Daftar seluruh tautan yang dimiliki pengguna ditampilkan secara vertikal di dashboard. Masing-masing tautan ditampilkan dengan rapi beserta tombol kontrol untuk pengelolaan.

5) Mengedit Tautan (Update)



Gambar 8. Tampilan Edit Link

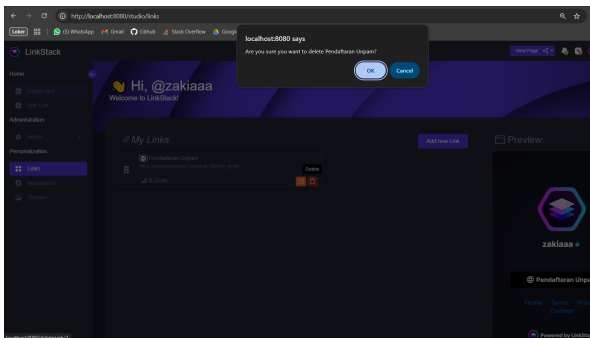
Gambar 8 memperlihatkan form edit tautan yang memungkinkan pengguna untuk memperbarui nama atau URL tautan yang sudah ada, sebelum disimpan kembali ke sistem.



Gambar 9. Tampilan Berhasil Edit Link

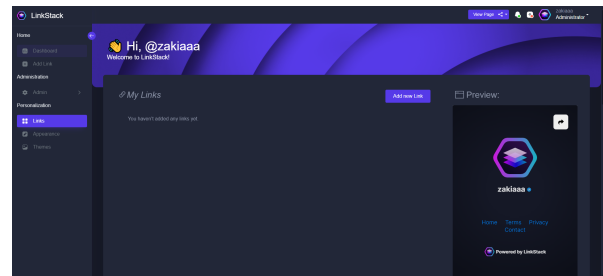
Pengguna menekan tombol edit pada tautan tertentu, mengubah nama atau URL, kemudian menyimpan perubahan. Setelah proses edit dilakukan, gambar 9 menunjukkan hasil akhir di mana data tautan sudah diperbarui sesuai input terbaru pengguna.

6) Menghapus Tautan (Delete)



Gambar 10. Tampilan Konfirmasi Delete Link

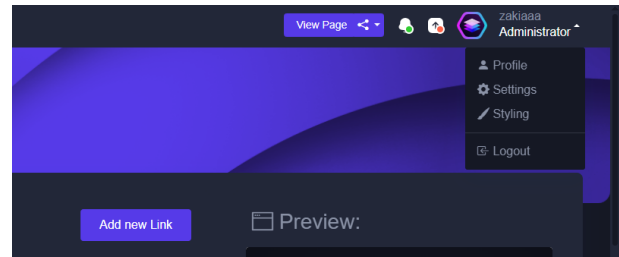
Sebelum tautan dihapus, sistem akan menampilkan jendela konfirmasi untuk memastikan bahwa pengguna benar-benar ingin menghapus tautan tersebut seperti pada Gambar 10.



Gambar 11. Tampilan Setelah Hapus Link

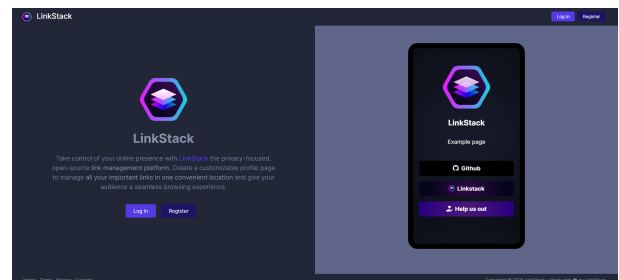
Gambar 11 menunjukkan bahwa tautan yang telah dihapus tidak lagi muncul dalam daftar di dashboard, menandakan proses delete berhasil dilakukan.

7) Logout



Gambar 12. Tampilan Letak Logout

Gambar 12 menunjukkan lokasi tombol logout pada dashboard *LinkStack*. Tombol logout terletak di pojok kanan atas halaman, memungkinkan pengguna keluar dari sesi akun mereka dengan mudah dan cepat.



Gambar 13. Tampilan Setelah Logout

Setelah pengguna menekan tombol logout, mereka akan diarahkan kembali ke halaman login seperti Gambar 13. Ini menandakan bahwa sesi pengguna telah berakhir dan akses ke dashboard tidak lagi tersedia tanpa login ulang.

3.3 Pengujian

Tabel 3. Pengujian Fitur

No	Fitur yang Diuji	Langkah Pengujian (Test Steps)	Output yang Diharapkan	Output Aktual	Status
----	------------------	--------------------------------	------------------------	---------------	--------

1	Register Pengguna	1. Klik tombol Register 2. Isi username, email, dan password Klik Submit	Akun berhasil dibuat dan diarahkan ke login	Akun berhasil dibuat dan diarahkan ke login	Pass
2	Login Pengguna	1. Buka halaman login 2. Masukkan username & password 3. Klik tombol login	Berhasil masuk ke dashboard	Berhasil masuk ke dashboard	Pass
3	Menambah Tautan	1. Klik tombol Add Link 2. Isi nama tautan dan URL 3. Klik tombol Save	Tautan baru muncul di dashboard	Tautan baru muncul di dashboard	Pass
4	Melihat Daftar Tautan	1. Login ke dashboard 2. Lihat daftar tautan yang sudah dibuat	Tautan ditampilkan lengkap dan bisa diakses	Tautan ditampilkan lengkap dan bisa diakses	Pass
5	Mengedit Tautan	1. Klik tombol Edit pada salah satu tautan 2. Ubah nama atau URL 3. Klik tombol Save	Tautan diperbarui sesuai input baru	Tautan diperbarui sesuai input baru	Pass
6	Menghapus Tautan	1. Klik tombol Delete pada salah satu tautan 2. Konfirmasi penghapusan	Tautan terhapus dari daftar	Tautan terhapus dari daftar	Pass
7	Logout	1. Klik tombol Logout di dashboard	Kembali ke halaman login	Kembali ke halaman login	Pass

Tabel 3 di atas menunjukkan hasil pengujian terhadap fitur-fitur utama pada aplikasi *LinkStack* berbasis *web* yang dijalankan menggunakan Docker. Seluruh proses pengujian dilakukan secara manual melalui antarmuka pengguna (UI) dengan mengikuti langkah-langkah sistematis untuk setiap fitur, mulai dari autentikasi pengguna hingga operasi CRUD (*Create, Read, Update, Delete*) terhadap tautan.

Hasil pengujian menunjukkan bahwa seluruh fitur inti *LinkStack* berjalan dengan baik dan sesuai dengan output yang diharapkan. Tidak ditemukan bug atau kesalahan fungsional pada saat proses login, registrasi, penambahan, pengeditan, maupun penghapusan tautan. Dengan demikian, dapat disimpulkan bahwa aplikasi *LinkStack* memiliki stabilitas yang baik untuk penggunaan dasar sebagai platform manajemen tautan pribadi.

D. KESIMPULAN

Berdasarkan hasil pengujian terhadap aplikasi *LinkStack* yang dijalankan menggunakan Docker, dapat disimpulkan bahwa seluruh fitur utama aplikasi telah berfungsi dengan baik. Proses autentikasi melalui fitur register dan login berhasil dilakukan tanpa kendala, serta operasi CRUD (*Create, Read, Update, Delete*) terhadap tautan dapat dijalankan secara lancar oleh pengguna.

Pengujian dilakukan secara menyeluruh pada fitur-fitur seperti menambahkan tautan, menampilkan daftar tautan, mengedit dan menghapus tautan, serta logout. Semua fitur menghasilkan output sesuai dengan yang diharapkan, dengan status pengujian *pass* untuk setiap skenario. Dengan demikian, *LinkStack* dinilai layak digunakan

sebagai aplikasi manajemen tautan berbasis *web* yang ringan dan mudah di-deploy menggunakan Docker.

E. DAFTAR PUSTAKA

- Abdillah, R., Hermawan, R., Hermawansyah, W., Adkha, I., & Arifin, H. (2024). Pengujian Perangkat Lunak Sistem Informasi Inventori pada Usaha Jasa Pengiriman Paket. *Polygon: Jurnal Ilmu Komputer Dan Ilmu Pengetahuan Alam*, 2(4), 166–175.
- Amali, M. S., Fatmawati, S., & Rosdiana, D. A. (2025). Peran platform digital terhadap pengembangan UMKM di Indonesia [The role of digital platforms in the development of MSMEs in Indonesia]. *Prosiding Seminar Nasional Manajemen*, 4(1), 338–341.
- Amalia, A. (2022). *Analisis Pemanfaatan Playwright Untuk Pengujian Aplikasi Berbasis Web (Studi Kasus: Sistem Manajemen Jaringan)*.
- Anwar, C. (2022). Application of Academic Information System With Extreme Programming Method (Case Study: Jakarta International Polytechnic).
- Anwar, C. (2024). Rekomendasi Teknis Untuk Pengolahan Data Berbasis Web. *Jurnal Informatika Utama*, 2(1), 50-54.

- Anwar, C., & Riyanto, J. (2019). Perancangan Sistem Informasi Human Resources Development Pada PT. Semacom Integrated. *International Journal of Education, Science, Technology, and Engineering (IJESTE)*, 2(1), 19-38.
- Anwar, C., Jagat, L. S., Yanti, I., Anjarsari, E., & Sholihah, N. A. (2023). Pengembangan Media Pembelajaran Berbasis Teknologi Untuk Meningkatkan Kemampuan Anak. *Caruban: Jurnal Ilmiah Ilmu Pendidikan Dasar*, 6(2), 154-163.
- Anwar, C., Kom, S., Kom, M., Santiari, C. N. P. L., & Sitorus, Z. (2023). Buku Referensi Sistem Informasi Berbasis Kearifan Lokal.
- Anwar, C., Nurhasanah, M., Aflaha, D. S. I., & Handayani, S. (2023). DEVELOPMENT OF INFORMATION TECHNOLOGY-BASED LEARNING MEDIA FOR EDUCATORS IN ELEMENTARY SCHOOLS. *Jurnal Konseling Pendidikan Islam*, 4(2), 345-353.
- Anwar, Chairul, et al. "The Application of Mobile Security Framework (MOBSF) and Mobile Application Security Testing Guide to Ensure the Security in Mobile Commerce Applications." *Jurnal Sistim Informasi dan Teknologi* (2023): 97-102.
- Bonteanu, A. M., & Tudose, C. (2024). Performance Analysis and Improvement for CRUD Operations in Relational Databases from Java Programs Using JPA, Hibernate, Spring Data JPA. *Applied Sciences*, 14(7), 2743.
- Butarbutar, L. E., Davina, S., & Tambunan, V. F. (2024). Implementasi Docker Untuk Cloud Computing Pada Fedora Linux: Studi Kasus Python Dalam Kontainer. *Jurnal Studi Multidisipliner*, 8(11).
- F Setiawan. (2023). Analisis kebutuhan bio-link di kalangan UMKM Indonesia. *Jurnal Teknologi Informasi Dan Bisnis*, 2(5), 123–130.
- Fachrudin, M. R., & Affandi, A. S. (2025). Implementation and Analysis of Container Image Optimization Using Alpine Linux and Multi-Stage Builds. *Teknika*, 14(1), 9–18.
- Fauzan, R., Soedjono, F. P., Permadani, A. A., & Yakin, M. A. (2023). Perbandingan Pengujian Manual dan Terotomasi pada Software Enterprise Resource Planning. *Journal of Advances in Information and Industrial Technology*, 5(1), 23–30.
- firmansyah, I., Ali Topan Nugroho, M., Rosandi, A., & Saifudin, A. (2024). *OKTAL : Jurnal Ilmu Komputer dan Science Pengujian Aplikasi CRUD Dengan MySQL dan PHPMyAdmin*. 3(6).
- Halomoan, M. A., Kharisma, A. P., & Marji, M. (2021). Pengembangan Domain Specific Language Untuk Aplikasi CRUD Berbasis Web. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 5(1), 34–41.
- Handayani, T., Silalahi, L. M., Nugroho, S. S. P., Anwar, C., Mursyidin, I. H., Sumantri, A., ... & Yulianti, B. (2025). PENGANTAR SISTEM INFORMASI: KONSEP, TEKNOLOGI, DAN IMPLEMENTASI.
- Indra, S., Anwar, C., Kom, S., Asparizal, S., Kom, M., Nur, R. A., ... & Hafrida, L. *KOMPUTER DAN MASYARAKAT*. CV Rey Media Grafika.
- Ison, M. H., & Putra, R. E. (2021). Implementasi Virtual Server Berbasis Container Pada Sistem Informasi Geografis Cagar Budaya Mojokerto. *Journal of Informatics and Computer Science (JINACS)*, 3(02), 143–154.
- LinkStackOrg. (2024). *LinkStack: The open-source Linktree alternative*. GitHub.
- Makarim, S. A., & Haryati, S. (2024). *Analisis Strategi Pemasaran Digital Menggunakan Linkumkm Bri Pada Usaha Kecil Menengah Kecamatan Panumbangan*. <https://journal-mandiracendikia.com/jip-mc>
- Prasetyo, D. S., & Silfianti, W. (2023). Analisis Perbandingan Pengujian Manual Dan Automation Testing Pada Website E-Commerce. *Jurnal Ilmiah Teknik*, 2(2), 127–131.
- Pratama, S. D., Lasimin, L., & Dadaprawira, M. N. (2023a). Pengujian Black Box Testing Pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence Dan Boundary Value. *Jurnal Teknologi Sistem Informasi Dan Sistem Komputer TGD*, 6(2), 560–569.
- Pratama, S. D., Lasimin, L., & Dadaprawira, M. N. (2023b). Pengujian Black Box Testing Pada Aplikasi Edu Digital Berbasis Website Menggunakan Metode Equivalence Dan Boundary Value. *Jurnal Teknologi Sistem Informasi Dan Sistem Komputer TGD*, 6(2), 560–569.
- Ramadhani, S. L., & Putro, H. P. (2023). Manajemen Pengujian Perangkat Lunak Menggunakan Aplikasi Qase. *AUTOMATA*, 4(1).
- Riski, M., & Renanti, M. D. (2024). Analisis Perbandingan Manual Testing Dan Automation Testing Pada

Sisitem Informasi Human Resource Development.
Kurawal-Jurnal Teknologi, Informasi Dan Industri,
7(2), 1–11.

Riyanto, D. J., Pizaini, P., Affandes, M., & others. (2022).
Implementasi Service Choreography Pattern
Arsitektur Microservice Classroom Akademik
Menggunakan Docker. *JUPI (Jurnal Ilmiah Penelitian
Dan Pembelajaran Informatika)*, 7(3), 768–779.

Samsumar, L. D., Nasiroh, S., Farizy, S., Anwar, C.,
Mursyidin, I. H., Rosdiyanto, R., ... & Prastyo, D.
(2025). KEAMANAN SISTEM INFORMASI:
PERLINDUNGAN DATA DAN PRIVASI DI
ERA DIGITAL.

Sakinah, F. A., Aditiawan, F. P., & Nurlaili, A. L. (2024).
Pengujian Pada Aplikasi Manajemen Aset
Menggunakan Black Box Testing. *JATI (Jurnal
Mahasiswa Teknik Informatika)*, 8(3), 2814–2823.

Sihombing, P. S. M., Pradana, F., & Purnomo, W. (2023).
Pengembangan Sistem Informasi Akademik
berbasis Web di SMP Negeri 3 Sidikalang. *Jurnal
Pengembangan Teknologi Informasi Dan Ilmu
Komputer*, 7(3), 1173–1182.

Wijayanti, R. R., S ST, M. M. S. I., Anwar, C., Kom, S.,
Indra, S., Kom, M., ... & Kom, M. (2023).
Arsitektur dan Organisasi Komputer. CV Rey
Media Grafika.