



Perancangan Aplikasi Todo menggunakan Node.js dan REST API

Rara Syifa Qaisa¹, Andriani Putri², Hayatun Maghfirah³

rarasifaqaisa@gmail.com, andrianiputri@utu.ac.id, hayatunmaghfirah@utu.ac.id

^{1,2,3} Teknologi Informasi, Universitas Teuku Umar, Indonesia

Informasi Artikel

Diterima : 6 Des 2023

Direview : 11 Des 2023

Disetujui : 30 Des 2023

Kata Kunci

Manajemen Tugas,
Aplikasi Produktivitas,
Pengembangan
Perangkat Lunak,
Antarmuka Pengguna,
Teknologi Informasi

Abstrak

Dalam perancangan aplikasi Todo, penulis menyoroti tantangan terkini dalam manajemen tugas sehari-hari pada perusahaan Chickin. Penelitian ini memperkenalkan pendekatan yang menggabungkan identifikasi kebutuhan pengguna, desain antarmuka yang intuitif, implementasi menggunakan Node.js, serta pembuatan REST API untuk menyediakan solusi yang responsif dan efisien. Tujuan penelitian ini adalah merancang aplikasi manajemen tugas yang mengakomodasi kebutuhan pengguna dan meningkatkan produktivitas mereka. Hasil perancangan menunjukkan bahwa penggunaan Node.js, Express.js, dan REST API dapat memberikan fondasi yang kokoh dan efisien dalam pengembangan aplikasi Todo pada perusahaan Chickin.

Keywords

Task Management,
Productivity Applications,
Software Development, User
Interface, Information
Technology

Abstrak

In designing the Todo application, the author highlights current challenges in day-to-day task management at Chickin Company. This research introduces an approach that integrates user needs identification, intuitive interface design, implementation using Node.js, and the development of a REST API to provide a responsive and efficient solution. The goal of this research is to design a task management application that accommodates user needs and enhances their productivity. The design outcomes demonstrate that the utilization of Node.js, Express.js, and REST API can provide a robust and efficient foundation in developing the Todo application at Chickin Company.

A. Pendahuluan

Dalam era digital yang terus berkembang, tugas-tugas sehari-hari kita semakin kompleks dan beragam. Manajemen tugas menjadi krusial untuk meningkatkan produktivitas dan mengurangi stres yang terkait dengan banyaknya tanggung jawab [1]. Di tengah tantangan ini, aplikasi Todo telah menjadi tonggak dalam membantu individu mengatur dan menyelesaikan tugas-tugas mereka.

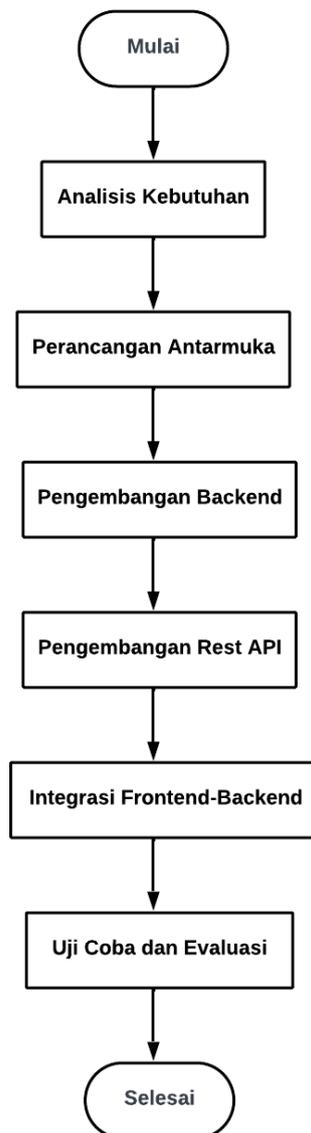
Dalam teori-teori pendukung, konsep manajemen tugas telah dipelajari secara intensif. Teori manajemen waktu oleh Peter F. Drucker, yang menekankan pentingnya mengalokasikan waktu dengan bijak untuk meningkatkan efisiensi, masih relevan dalam konteks perancangan aplikasi Todo [2]. Selain itu, teori psikologi seperti teori self-regulation dari Albert Bandura menyoroti pentingnya pengaturan diri dalam menyelesaikan tugas-tugas, yang menjadi landasan bagi fitur-fitur motivasi yang dapat diimplementasikan dalam aplikasi.

Penelitian sebelumnya menunjukkan bahwa aplikasi Todo memiliki peran penting dalam membantu individu dalam memprioritaskan, mengatur, dan menyelesaikan tugas-tugas mereka. Penggunaan aplikasi Todo secara signifikan dapat meningkatkan produktivitas dan mengurangi tingkat stres. Penggunaan aplikasi Todo yang memiliki antarmuka pengguna yang intuitif dan responsif memberikan pengalaman yang lebih positif bagi pengguna [3].

Tujuan dari penelitian ini adalah untuk merancang dan mengembangkan aplikasi Todo yang menggabungkan keandalan Node.js sebagai platform *backend* dan *Representational State Transfer (REST) Application Programming Interface (API)* untuk memfasilitasi komunikasi antara server dan klien. Aplikasi Todo pada perusahaan Chickin memperkenalkan integrasi yang mulus antara fitur yang berpusat pada pengguna, desain antarmuka yang intuitif, dan pemanfaatan teknologi terkini seperti Node.js, Express.js, dan REST API. Kombinasi ini memastikan solusi yang sangat responsif, efisien, dan dapat disesuaikan untuk tantangan manajemen tugas sehari-hari di perusahaan Chickin. Hasil dari perancangan ini diharapkan dapat memberikan solusi yang efisien dan responsif dalam manajemen tugas sehari-hari, membantu pengguna untuk meningkatkan produktivitas dan mengurangi beban stres terkait tugas-tugas yang harus diselesaikan [4].

B. Metode Penelitian

Penelitian ini mengadopsi pendekatan pengembangan perangkat lunak dengan metodologi *agile*. Tahapan penelitian meliputi analisis kebutuhan, perancangan antarmuka, pengembangan *backend*, pengembangan *REST API* dan integrasi *frontend-backend*. Adapun pengguna dalam penelitian ini adalah ditujukan kepada perusahaan Chickin Indonesia. Diagram alir penelitian dapat dilihat pada Gambar 1.



Gambar 1. Diagram Alir Penelitian

1. Analisis Kebutuhan

Identifikasi kebutuhan pengguna terkait manajemen tugas dan penentuan fitur-fitur yang esensial untuk aplikasi Todo, seperti membuat, mengedit, dan menghapus tugas, serta pengelompokan atau penandaan tugas. Data dikumpulkan melalui wawancara pengguna untuk mengidentifikasi kebutuhan dan preferensi pengguna terkait fitur aplikasi. Metode ini digunakan untuk mengidentifikasi kebutuhan, preferensi, dan harapan pengguna terkait fitur aplikasi. Wawancara dengan pengguna membantu dalam memahami secara mendalam kebutuhan yang mereka miliki dalam manajemen tugas sehari-hari.

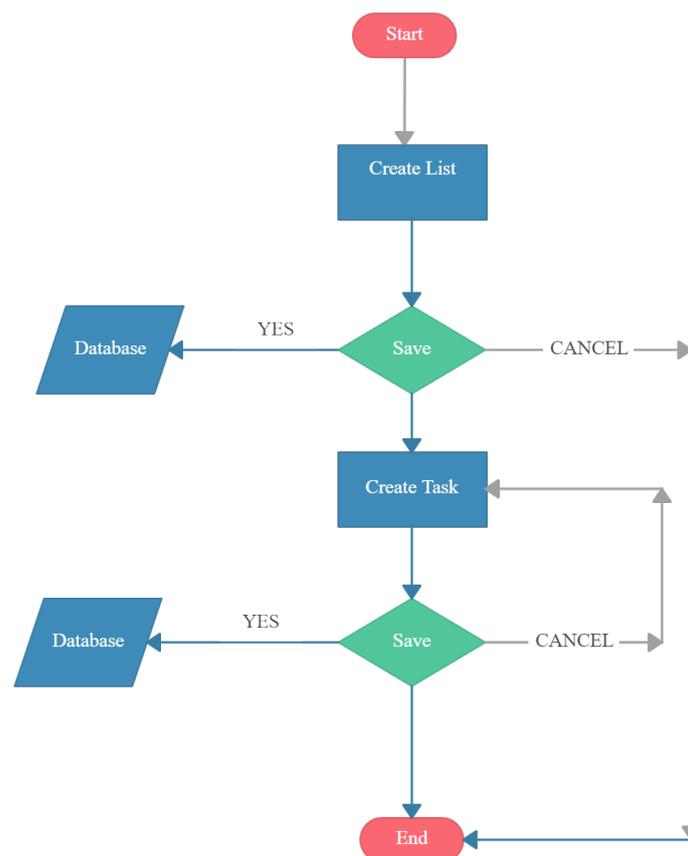
Tabel 1. Tabel Analisis Kebutuhan Pengguna

Metode	Deskripsi
<i>Get</i>	Digunakan untuk mengambil data dari server.
<i>Post</i>	Mengirimkan data baru ke server untuk diproses atau disimpan.
<i>Put</i>	Memperbarui atau mengganti data yang ada di server.
<i>Delete</i>	Menghapus data dari server.
<i>Patch</i>	Digunakan untuk memperbarui sebagian data di server.

2. Perancangan Antarmuka

Pembuatan desain antarmuka pengguna (*UI/UX*) menggunakan *wireframe* dan *mockups* dan penyesuaian desain dengan prinsip-prinsip antarmuka untuk memastikan aplikasi mudah digunakan dan responsif. Figma dan Sketch digunakan untuk perancangan antarmuka pengguna (*UI/UX*) dengan membuat *wireframe* dan *mockups*. Ini membantu dalam visualisasi awal desain aplikasi sebelum implementasi.

Skenario saat pengguna membuat daftar dan tugas dalam aplikasi digambarkan dalam sebuah diagram alir. Misalnya, dari awal membuat daftar, menyimpannya ke database, kemudian ke opsi untuk membuat tugas dan menyimpannya atau membatalkan tindakan tersebut. Diagram ini membantu untuk memvisualisasikan alur tindakan yang mungkin dilakukan oleh pengguna saat menggunakan aplikasi. Diagram alir aplikasi dapat dilihat pada Gambar 2.

**Gambar 2.** Diagram Alir Aplikasi

Keterangan Warna

Biru : Representasi antarmuka pengguna.

Hijau : Proses *backend* dan logika bisnis.

Merah : Komunikasi melalui *REST API*.

3. Pengembangan *Backend*

Implementasi sistem *backend* menggunakan Node.js sebagai basis untuk logika bisnis dan manajemen data dan pengembangan fungsionalitas yang dibutuhkan untuk menyimpan, memperbarui, dan mengelola tugas dalam database. Instrumen dan teknik yang digunakan dalam perancangan aplikasi Todo pada perusahaan Chickin meliputi *wireframe* dan *mockup tools*, Node.js, Express.js, Postman dan metode Scrum. Penjelasan instrumen dan teknik perancangan dapat dilihat pada Tabel 1.

Tabel 2. Instrumen dan Teknik

Instrumen dan Teknik	Deskripsi
Node.js	Digunakan sebagai platform utama untuk pengembangan <i>backend</i> . Node.js memberikan kecepatan eksekusi tinggi dan memungkinkan pengembangan aplikasi yang skalabel.
Express.js	Framework untuk pengembangan aplikasi web menggunakan Node.js. Express.js membantu dalam membangun aplikasi web dengan cara yang efisien dan terstruktur.
Postman	Alat yang digunakan untuk pengujian dan debugging <i>REST API</i> . Postman memfasilitasi pengujian endpoint-API untuk memastikan fungsionalitas yang tepat sebelum integrasi dengan antarmuka pengguna.
Metode Scrum	Digunakan untuk manajemen proyek <i>agile</i> . Metode Scrum membantu dalam mengatur iterasi dan perencanaan proyek secara terstruktur, memungkinkan fleksibilitas dalam pengembangan aplikasi.

```
const router = require("express").Router();
const Todo = require("../models/Todo");

// Route untuk halaman utama
router.get('/', async (req, res) => {
  try {
    const allTodo = await Todo.find();
    res.render('index', { todo: allTodo });
  } catch (err) {
    console.log(err);
    res.status(500).json({ error: 'Internal Server Error' });
  }
});

// Route untuk halaman pendaftaran pengguna baru
router.get('/register', (req, res) => {
  res.render('register');
});

// Route untuk halaman login
router.get('/login', (req, res) => {
  res.render('login');
});

module.exports = router;
```

Gambar 3. Potongan Code Perancangan Aplikasi

Penggunaan alat dan teknik ini membantu dalam mengatur dan melaksanakan tahapan-tahapan pengembangan aplikasi dengan lebih terstruktur dan efisien, dari perancangan awal hingga pengujian dan implementasi akhir [5].

4. Pengembangan *REST API*

Pembuatan *REST API* untuk memungkinkan komunikasi yang efisien antara *frontend* (antarmuka pengguna) dan *backend* (server). *REST API* umumnya memiliki beberapa fitur inti yang digunakan atau dikembangkan, terutama dalam konteks pengembangan aplikasi web atau layanan daring. Dalam perancangan aplikasi *Todo* pada perusahaan *Chickin* fitur yang digunakan adalah *endpoints*, *verbs*, format data, *Authentication* dan *Authorization*.

5. Integrasi *Frontend-Backend*

Penggabungan antarmuka pengguna dengan sistem *backend* menggunakan *API* yang telah dibuat sebelumnya dan memastikan bahwa data dapat ditampilkan dan diubah secara akurat melalui antarmuka pengguna dan disimpan dengan benar di *backend*.

C. Hasil dan Pembahasan

Todo pada perusahaan *Chickin* adalah aplikasi manajemen tugas yang menggunakan *Node.js* dan *REST API*, telah berhasil direalisasikan sesuai dengan rancangan sistem yang telah disusun. Aplikasi ini menyajikan antarmuka pengguna

yang sederhana dan efisien untuk membantu pengguna mengatur tugas-tugas mereka. Pengujian aplikasi telah dilakukan dan hasilnya menunjukkan kinerja yang memuaskan serta kehandalan dalam pengelolaan tugas[6].

Hasil penelitian ini menegaskan bahwa integrasi Node.js sebagai basis *backend* dan *REST API* memungkinkan aplikasi untuk berfungsi dengan baik dalam melakukan pertukaran data antara klien dan server. Antarmuka pengguna yang didesain dengan baik memperlihatkan responsivitas yang tinggi dan kesederhanaan penggunaan, sesuai dengan tujuan pengembangan aplikasi [7]. Diskusi pada tahap ini menyoroti kesesuaian hasil penelitian dengan kebutuhan pengguna yang membutuhkan aplikasi *Todo* yang efektif dan mudah digunakan.

Setiap tahapan ini penting untuk memastikan bahwa aplikasi yang dikembangkan sesuai dengan kebutuhan pengguna, memiliki antarmuka yang ramah pengguna, berfungsi dengan baik, aman, dan responsif dalam menjalankan tugas-tugas manajemen yang dituju [8-9].

Rancangan aplikasi *Todo* pada perusahaan *Chickin* terdiri dari beberapa bagian halaman dengan desain antarmuka yang bersih dan *user-friendly* sehingga pengguna dapat dengan mudah membuat daftar baru, menambah, mengedit, atau menandai tugas sebagai selesai. Berikut adalah tampilan rancangan aplikasi *Todo* pada perusahaan *Chickin* yang mencakup *home screen*, *registration*, *login*, *create task*, dan *dashboard*.



Gambar 4. Halaman *Home*

Halaman ini menampilkan selamat datang bagi pengguna baru yang menggunakan aplikasi Todo pada perusahaan Chickin. Tujuannya adalah mengarahkan pengguna ke halaman selanjutnya dan memudahkan pengguna dalam menavigasi antarmuka.

Masuk'."/>

15:36

ISI FORMULIR UNTUK MEMULAI

Nama

Email

Password

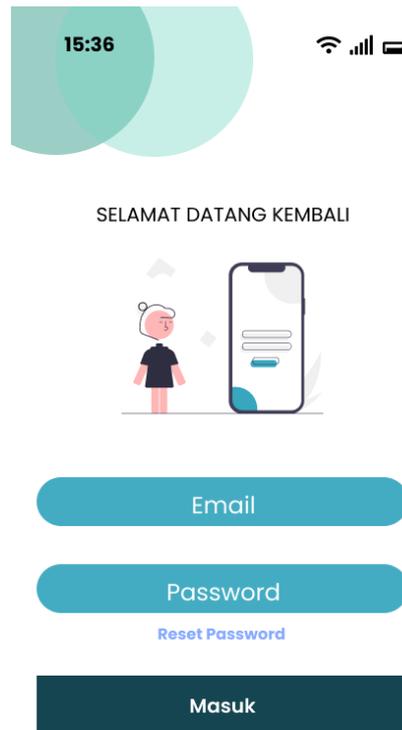
Konfirmasi Password

DAFTAR

sudah memiliki akun? [Masuk](#)

Gambar 5. Halaman *Registration*

Halaman ini menyediakan formulir pendaftaran untuk pengguna yang baru. Biasanya mencakup bidang seperti nama pengguna, email, password, dan konfirmasi password. Tujuannya adalah memudahkan pengguna baru untuk membuat akun dengan cepat dan mudah.



Gambar 6. Halaman *Login*

Halaman ini memberikan akses kepada pengguna yang sudah memiliki akun. Terdapat bidang untuk memasukkan kredensial *login* seperti nama pengguna dan kata sandi. Halaman ini didesain untuk memastikan bahwa hanya pengguna yang terautentikasi yang bisa mengakses fitur aplikasi.

Setelah pengguna login ke dalam aplikasi, halaman yang ditampilkan selanjutnya adalah halaman create task. Halaman ini memungkinkan pengguna untuk menambahkan tugas baru ke dalam daftar mereka. Tujuannya adalah memberikan cara yang mudah bagi pengguna untuk membuat dan mengatur tugas baru.

Selanjutnya pengguna akan dibawa ke halaman *dashboard* tampilan utama yang memberikan gambaran keseluruhan tentang progres pengguna dalam menyelesaikan tugas. Halaman ini berisi tugas-tugas yang belum selesai. Tujuannya adalah memberikan pandangan menyeluruh yang dapat membantu pengguna dalam mengelola dan memprioritaskan tugas-tugas mereka dengan lebih baik.

Hasil temuan ini mencerminkan pentingnya penggunaan teknologi modern dalam pengembangan aplikasi [10]. Ketersediaan Node.js sebagai platform *backend* yang andal dan *REST API* untuk komunikasi antar bagian sistem memainkan peran penting dalam keberhasilan aplikasi Todo pada perusahaan Chickin. Interpretasi ini mengonfirmasi bahwa penggunaan teknologi-teknologi ini

memberikan solusi yang sesuai dan responsif dalam mendukung manajemen tugas-tugas sehari-hari.

Temuan ini sejalan dengan teori-teori manajemen waktu dan efisiensi kerja yang menekankan pentingnya alat bantu yang efektif dalam meningkatkan produktivitas [11]. Temuan ini juga mendapat konfirmasi dari penelitian-penelitian sebelumnya yang menyoroti kebutuhan akan aplikasi *Todo* yang memiliki antarmuka pengguna yang ramah dan kinerja yang handal.

Rancangan sistem yang diimplementasikan berhasil mengintegrasikan komponen-komponen utama aplikasi dengan baik. Alur kerja aplikasi dapat divisualisasikan dengan jelas melalui diagram alur yang menggambarkan interaksi antara pengguna, antarmuka, *backend*, dan komunikasi melalui *REST API*. Perancangan ini mengidentifikasi penggunaan *REST API* untuk mengintegrasikan aplikasi *Todo* dalam manajemen tugas dan mendukung kolaborasi tim. *REST API* memfasilitasi pertukaran data yang efisien antara tugas yang berbeda dan dampaknya terhadap pengguna dalam manajemen tugas. Implementasi rancangan sistem ini memastikan aplikasi berjalan dengan lancar dan sesuai dengan tujuan pengembangan [12].

Dengan demikian, hasil penelitian ini menunjukkan bahwa *Todo* pada perusahaan *Chickin* sebagai aplikasi manajemen tugas sukses menerapkan konsep-konsep teori terkait manajemen tugas dan teknologi modern untuk memberikan solusi yang efektif bagi pengguna dalam mengatur dan menyelesaikan tugas-tugas mereka.

D. Simpulan

Perancangan aplikasi *Todo* pada perusahaan *Chickin* menghasilkan platform manajemen tugas yang responsif dan efisien. Proses dalam perancangan aplikasi *Todo* melibatkan identifikasi kebutuhan, perancangan antarmuka, implementasi *backend*, pembuatan *REST API*, serta integrasi *frontend-backend*.

Hasil penelitian menunjukkan bahwa penerapan teknologi modern seperti *Node.js* dan metodologi *agile* dalam pengembangan aplikasi dapat menghasilkan solusi yang responsif dan efektif bagi pengguna. Kesimpulan ini menegaskan bahwa aplikasi *Todo* pada perusahaan *Chickin* memberikan solusi yang sesuai dan komprehensif dalam manajemen tugas sehari-hari.

Dalam kesimpulan ini, ditekankan bahwa perancangan aplikasi *Todo* pada perusahaan *Chickin* tidak hanya memenuhi kebutuhan pengguna, tetapi juga memperkuat penggunaan teknologi-teknologi terkini untuk menciptakan solusi yang responsif dan mendukung produktivitas pengguna dalam manajemen tugas mereka.

E. Ucapan Terima Kasih

Penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan dukungan selama proses perancangan aplikasi *Todo* pada perusahaan *Chickin* ini. Terima kasih kepada *Chickin Indonesia* yang telah menjadi wadah penulis selama magang serta terima kasih atas kontribusi, dukungan dan dedikasi dari semua pihak yang telah turut serta dalam perancangan aplikasi *Todo* pada perusahaan *Chickin*.

F. Referensi

- [1] N. Eli, Agussalim, "Rancang Bangun Back-end API pada Aplikasi Mobile AyamHub Menggunakan Framework Node JS Express," *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, vol. 11, no. 3, pp. 524-531, 2023.
- [2] N. S. Glenn, K. R. Wahyu, Jalinus, "Perancangan Aplikasi Informasi Cuaca Berbasis Android Menggunakan API dan JSON," *Jurnal Ilmiah KOMPUTASI*, vol. 21, no. 1, pp. 115-122, 2022.
- [3] S. Wahfiuddin, S. Andrian, "Perancangan Aplikasi Mobile E-Commerce Perangkat Elektronik dengan Menggunakan REST API Berbasis Android," *IT Journal*, vol. 8, no. 2, pp. 173-183, 2020.
- [4] G. Syam, Y. Maulana, "Perancangan Aplikasi Tracking Barang Berbasis Android Menggunakan Google Map API pada PT. Sukma Jaya Abadi," *Jurnal Informatika & Komputasi*, vol. 15, no. 1, pp. 54-60, 2021.
- [5] F. F. Royyan, K. Muhammad & P. Mahendra, "Perancangan Microservice Berbasis REST API pada Google Cloud Platform Menggunakan NodeJS dan Python," *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol. 11, no. 3, pp. 1103-1112, 2023.
- [6] R. D. A. Muhammad, S. E. Arda, "Rancang Bangun Aplikasi CBT Berbasis Android Menggunakan REST API," *SUBMIT (Jurnal Ilmiah Teknologi Informasi dan Sains)*, vol. 2, no. 2, pp. 38-42, 2022.
- [7] Supono, P. Awlita, "Perancangan Sistem Informasi Checking Tarif dan Tracking Barang Menggunakan API di PT. Dakota Indonesia Express," *COMPETITIVE*, vol. 14, no. 1, pp. 24-30, 2019.
- [8] T. Agus, F. Muhammad, E. Depandi, R. Fajar & S. Elgamar, "Perancangan Aplikasi Tanggap API Berbasis Android Menggunakan Metode Design Sprint," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 9, no. 2, pp. 215-223, 2022.
- [9] A. Efendi, "Perancangan Aplikasi E-Travel Wisata Malang Menggunakan Google Maps Api Berbasis Android," *Ubiquitous: Computers and its Applications Journal*, vol. 3, no. 1, pp. 23-26, 2020.
- [10] F. S. Handayani, R. Rosnelly, "Perancangan Aplikasi Galeri E-Jurnal Dengan REST API Berbasis Android," *InfoSys Journal*, vol. 5, no. 2, pp. 161-171, 2021.
- [11] Nasrul, A. Izhar "Pengembangan REST API Dengan Menggunakan Express JS Untuk Mencari Mentor Pribadi," *Jurnal Informatika Terpadu*, vol. 9, no. 2, pp. 92-102, 2023.
- [12] H. Nuryadin, "Perancangan Aplikasi Edukasi Nama Hewan Dalam Bahasa Inggris Menggunakan Android," *Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI)*, vol. 3, no. 2, pp. 299-306, 2022.