

# Pengembangan Application Programming Interface Whatsapp Berbasis Node.js

Eny Maria<sup>1</sup>, Gusti Agung Kurniawan<sup>2</sup>, Eko Junirianto<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknologi Rekayasa Perangkat Lunak, Politeknik Pertanian Negeri Samarinda  
e-mail: <sup>1</sup>enymaria@politanisamarinda.ac.id, <sup>2</sup>e4gustiagung@gmail.com,

<sup>3</sup>eko@politanisamarinda.ac.id

## Abstrak

*Penelitian ini berfokus pada pengembangan Application Programming Interface (API) WhatsApp berbasis Node.js yang dirancang untuk mendukung kebutuhan sistem di lingkungan perusahaan CV. Britech. Menggunakan metode ADDIE yang terdiri dari tahap Analisis, Desain, Pengembangan, Implementasi, dan Evaluasi untuk digunakan dalam pengembangan program. Penelitian ini bertujuan untuk menciptakan solusi komunikasi yang efisien dan efektif melalui platform WhatsApp. Hasil pengembangan menunjukkan kemampuan API untuk mengirimkan pesan ke nomor WhatsApp dengan panjang 10 digit. Pengujian Black Box dilakukan untuk memastikan fungsi aplikasi bekerja dengan baik dan sesuai dengan tujuan yang diharapkan. Fitur utama dari aplikasi ini adalah Single Sender, yang memungkinkan pengiriman pesan teks dan media secara personal. Hasil waktu rata-rata pengiriman pesan personal pada saat kondisi server normal ialah 2.24 detik. Kesuksesan pengembangan API ini menunjukkan potensi besar dalam penerapannya di lingkungan perusahaan CV. Britech, memberikan solusi komunikasi yang inovatif dan dapat diandalkan.*

**Kata Kunci**— API, WhatsApp, Node.js, Metode ADDIE

## 1. PENDAHULUAN

CV. Britech merupakan perusahaan yang dinamis dan memerlukan sistem informasi yang efisien untuk mendukung proses operasional, administrasi, dan komunikasi internal. Penggunaan WhatsApp sebagai saluran komunikasi telah menjadi umum di kalangan karyawan dan manajemen. Meskipun WhatsApp menjadi alat komunikasi yang populer, pengintegrasian WhatsApp dengan sistem informasi perusahaan CV. Britech terkadang masih terbatas. Oleh karena itu, pengembangan Application Programming Interface (API) WhatsApp berbasis Node.js dianggap sebagai solusi potensial untuk meningkatkan integrasi dan efisiensi komunikasi di lingkungan perusahaan. Integrasi WhatsApp dengan sistem perusahaan melalui API dapat membawa sejumlah keuntungan.

Pengembangan API WhatsApp berbasis Node.js memiliki manfaat langsung bagi karyawan dan manajemen. Karyawan dapat menerima pemberitahuan penting, informasi rapat, atau perubahan jadwal dengan cepat dan langsung melalui WhatsApp. Manajemen dapat dengan mudah berkomunikasi dengan karyawan, memfasilitasi diskusi, dan memberikan arahan secara *real-time*.

Penelitian terdahulu yang menjadi dasar penulisan landasan penelitian ini antara lain penelitian pertama yang mana pada penelitian sistem informasi dibuat suatu sistem yang mampu mengirimkan pesan pemberitahuan kejadian, setiap ketidakhadiran siswa akan dikirimkan melalui pesan WhatsApp kepada orang tua, hingga membantu guru memastikan kehadiran dan memudahkan orang tua memantau kehadiran siswa di sekolah [1]. Dan penelitian berikut ini

bertujuan untuk merencanakan dan menerapkan teknologi chatbot yang memberikan mahasiswa akses informasi yang cepat dan mudah tentang akademik Fakultas Teknologi Industri dan Informatika UHAMKA secara mudah dan cepat. Pengguna tidak perlu menunggu terlalu lama untuk menerima balasan dari chatbot, sehingga membuat chatbot yang digunakan dapat beroperasi dengan cepat dan tepat [2].

Berdasarkan yang telah dipaparkan pada latar belakang, maka yang menjadi rumusan masalah dalam penelitian ini ialah, “Bagaimana cara membuat Application Programming Interface WhatsApp Berbasis Node.js?”

Dengan keterbatasan waktu yang dimiliki, maka peneliti membatasi penelitian ini dengan mengerjakan fitur utama pada WhatsApp seperti mengirim pesan maupun media berupa gambar, video, audio, dan dokumen dan menggunakan *library* Node.js dalam pengembangan API WhatsApp.

Tujuan utama dari penelitian ini adalah mengembangkan *Application Programming Interface* (API) WhatsApp berbasis Node.js yang dapat mendukung keperluan sistem di lingkungan perusahaan CV. Britech.

Hasil penelitian ini diharapkan dapat mengungkapkan potensi keunggulan Node.js dalam meningkatkan kinerja dan responsivitas API, sehingga pengguna dapat merasakan pengalaman berkomunikasi yang lebih lancar dan *real-time*. Hasil penelitian ini juga dapat menjadi landasan bagi pengembangan solusi serupa di perusahaan lain serta memberikan wawasan berharga terkait potensi penerapan teknologi Node.js dalam pengembangan API berbasis pesan di industri dan bisnis.

## 2. METODE PENELITIAN

### 2.1. Landasan Teori

#### 2.1.1. Sistem Informasi

Sistem informasi merupakan sebuah sistem dalam organisasi yang dirancang untuk memenuhi kebutuhan pengelolaan transaksi harian. Sistem ini mendukung fungsi operasional organisasi sekaligus menyediakan laporan-laporan yang dibutuhkan oleh pihak eksternal tertentu [3]. Menurut Khristianto dalam [4] sistem informasi adalah implementasi sistem teknologi informasi serta komunikasi yang diselenggarakan oleh perusahaan bisnis.

#### 2.1.2. Application Programming Interface

*Application Programming Interface* (API) adalah interface yang dapat menghubungkan satu aplikasi dengan aplikasi lainnya. Dapat juga diartikan sebagai kumpulan suatu fungsi, perintah dan protokol yang bisa digunakan programmer saat membuat sebuah perangkat lunak untuk sistem operasi tertentu. API juga memungkinkan programmer untuk memakai fungsi standar untuk melakukan interaksi dengan *Operating System* [5].

Menurut Choirudin dan Adil dalam [6] API berarsitektur REST memungkinkan sistem-sistem berinteraksi satu sama lain, dan memudahkan pengiriman dan penerimaan data. Dalam arsitektur ini, sumber daya *Database* dapat dihubungkan dengan endpoint pada REST API.

Representational State Transfer yang disingkat REST yang merupakan gaya arsitektur untuk penerapan web service dalam menerapkan konsep perpindahan antar state. Contoh dari standarisasi ini adalah interaksi berbasis REST semua berkomunikasi status mereka menggunakan kode status HTTP standar. Jadi, 404 berarti sumber daya yang diminta tidak ditemukan; kode 401 berarti permintaan tidak diotorisasi; kode 200 berarti semuanya OK; dan 500 berarti ada kesalahan aplikasi yang tidak dapat dipulihkan pada server [7].

#### 2.1.3. WhatsApp

WhatsApp adalah sebuah aplikasi *chatting* pada yang biasanya tersedia di bursa *smartphone* yang memungkinkan penggunaannya berbagi gambar dan pesan. Aplikasi WhatsApp merupakan aplikasi yang sangat unik, dengan tampilan yang sederhana, dan tanpa memakan kuota internet yang banyak, tidak seperti aplikasi *chatting* yang lainnya [8].

Pengguna internet di Indonesia pada tahun 2022 ada 84,8 juta jiwa atau 88,7 persen yang menggunakan internet, dan 88,7 persen dari 84,8 juta jiwa itu adalah pengguna aplikasi whatsapp yang dimana banyak digunakan dari berbagai macam kalangan dan profesi seperti mahasiswa, pengusaha, politisi, selebriti, pelajar serta yang lainnya. Dengan ini perkembangan informasi yang sekarang semakin gencar maka perlu diimbangi juga dengan penyebaran yang sangat cepat [9].

#### 2.1.4. Node.js

Menurut Huda dalam [6] untuk mendukung Javascript sebagai berjalan dalam server, terdapat sebuah perangkat lunak yang dikenal dengan nama Node.js. Perangkat lunak tersebut dibuat untuk mengembangkan aplikasi berbasis web, yang ditulis dalam bahasa pemrograman Javascript, dan memiliki library HTTP tersendiri sehingga tidak perlu menjalankan aplikasi pendukung web seperti Apache.

Menurut Huang dalam [6] Node JS memiliki keunggulan yaitu penggunaan teknologi non-blocking, yang memungkinkan eksekusi tugas sekaligus tanpa harus menunggu tugas lain terselesaikan, sehingga performanya sangat responsif dan efisien.

Pada dasarnya, Node.js adalah sebuah runtime environment dan script library. Sebuah runtime environment adalah sebuah software yang berfungsi untuk mengeksekusi, menjalankan dan mengimplementasikan fungsi-fungsi serta cara kerja inti dari suatu bahasa pemrograman. Sedangkan script library adalah kumpulan, kompilasi atau bank data berisi skrip/kode-kode pemrograman [7].

#### 2.1.5. Laravel

Laravel merupakan *framework* PHP yang menekankan pada kesederhanaan dan fleksibilitas pada desainnya. Keunggulan ini didapatkan karena Laravel menggunakan konsep MVC (*Model View Controller*). Dengan migration, pengembangan dapat dengan mudah untuk melakukan modifikasi sebuah *Database* pada sebuah platform secara independen karena implementasi skema *Database* yang direpresentasikan dalam sebuah class [10]. Laravel menggunakan arsitektur MVC (*Model, View, and Controller*) dan OOP (*Object Oriented Programming*). Laravel memiliki pertahanan dari serangan-serangan dasar yang cukup mumpuni karena komunitas yang besar, yang bersuka rela untuk melakukan *Penetration Test*, jadi pada saat ditemukan *bug* atau celah keamanan maka para *developer* bersama-sama mencari solusi untuk memecahkan masalah tersebut. Laravel menyediakan beberapa fitur yang mencegah serangan dari SQL Injection, xsrif, xss [11].

#### 2.1.6. Database

Menurut Rachmadi dalam [12] *Database* (basis data) terdiri dari kata basis dan data. Basis disebut juga sebagai markas, gudang atau tempat pengumpulan. Sedangkan data merupakan catatan atas kumpulan fakta dunia nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan lain sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasi lainnya.

#### 2.1.7. UML

UML (*Unified Modelling Language*) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek.

#### 2.1.7.1. Use Case Diagram

*Use Case Diagram* terdiri dari sebuah aktor dan interaksi yang dilakukannya, aktor tersebut dapat manusia, perangkat keras, sistem lain, ataupun yang berinteraksi dengan sistem [13].

#### 2.1.7.2. Activity Diagram

*Activity Diagram* menggambarkan berbagai aliran aktivitas dalam sistem yang sedang di rancang, bagaimana masing-masing aliran berawal, kemungkinan terjadi dan bagaimana mereka berakhir. *Activity Diagram* menggambarkan proses paralel terjadi pada beberapa eksekusi [13].

#### 2.1.7.3. Class Diagram

*Class Diagram* adalah jenis diagram struktur statis dalam UML yang menggambarkan struktur sistem menunjukkan sistem class, atributnya, metode, dan hubungan antar objek [14].

### 2.2. Prosedur Penelitian

Metode yang digunakan dalam penelitian ini yaitu metode ADDIE karena menurut Pertiwi dalam [15] metode ini seringkali digunakan dalam pengembangan sistem. Model yang digunakan dalam penelitian ini akan dibagi menjadi 5 tahap yaitu *Analyze, Design, Development, Implementation and Evaluations*.

#### 2.2.1. Analisis (Analyze)

Menurut Pamungkas dan Rosid dalam [15] pada tahapan analisis ini peneliti melakukan analisis kebutuhan, mengidentifikasi masalah serta kebutuhan dan melakukan analisis tugas.

#### 2.2.2. Desain (Design)

Menurut Myint dalam [15] pada tahapan desain adalah perancangan produk awal yang akan dikembangkan. Dalam tahap ini akan menghasilkan antarmuka pengguna dari perancangan desain.

#### 2.2.3. Pengembangan (Develop)

Menurut Sabirin dalam [15] tahap ini Pengembangan ini pihak Admin akan memberi saran jika terdapat tampilan dari sistem tersebut kurang nyaman untuk dipakai user.

#### 2.2.4. Implementasi (Implement)

Menurut Putra dalam [15] tahap Implementasi, setelah melakukan perancangan sistem jika prototipe yang diinginkan sudah sesuai dengan kebutuhan. Sehingga dari sistem yang telah dibuat berguna bagi user sehingga dapat menerima notifikasi langsung melalui Whatsapp masing-masing pengguna.

#### 2.2.5. Evaluasi (Evaluate)

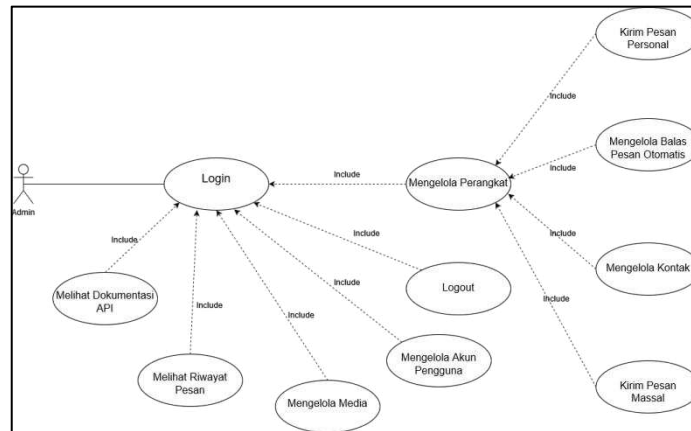
Mengutip dari Ismail dalam [15] Dari dilakukannya pengujian sistem, diharapkan sistem dapat berjalan sesuai dengan rancangan awal dan sesuai dengan yang diharapkan dari hasil analisa sistem dan tahap prototype.

## 3. HASIL DAN PEMBAHASAN

### 3.1. Use Case Diagram

Admin bertanggung jawab untuk melakukan berbagai aktivitas seperti login ke dalam sistem, mengelola perangkat mengelola akun pengguna, mengelola media yang akan digunakan dalam pengiriman pesan, dan mengelola daftar kontak. Selain itu, Admin juga dapat mengatur balasan otomatis berdasarkan kata kunci tertentu, mengirim pesan baik secara individual maupun

massal, serta melihat riwayat pesan yang telah dikirim. Admin juga memiliki akses ke dokumentasi API dan dapat melihat dokumen-dokumen terkait yang dibutuhkan. Secara keseluruhan, diagram ini menggambarkan kontrol penuh yang dimiliki oleh Admin dalam mengelola dan mengoperasikan seluruh aspek sistem pengelolaan pesan. Berikut *Use Case* ditampilkan pada **Gambar 1**.

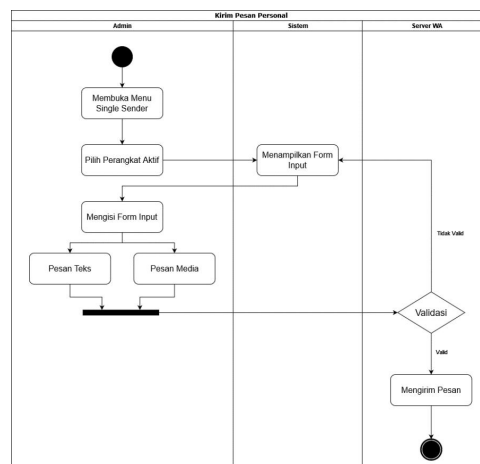


Gambar 1. Use Case

### 3.2. Activity Diagram

*Activity Diagram* "Kirim Pesan Personal" ini dimulai ketika Admin membuka menu "Single Sender" dalam *Dashboard*. Admin kemudian memilih perangkat aktif yang akan digunakan untuk mengirim pesan. Setelah perangkat dipilih, sistem menampilkan form input yang harus diisi oleh Admin dengan informasi yang diperlukan, seperti nomor penerima, tipe pesan (teks atau media), dan isi pesan.

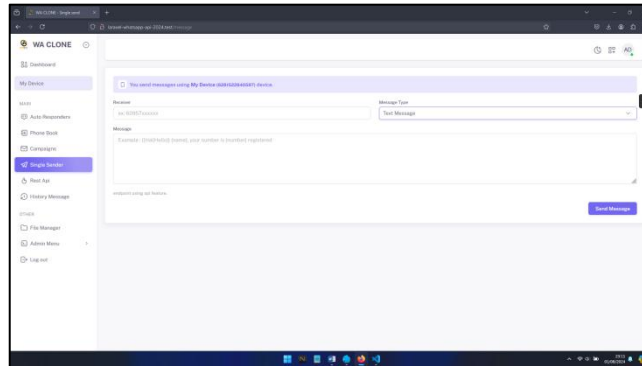
Saat form input diisi, Admin dapat memilih untuk mengirim pesan dalam bentuk teks atau media. Jika validasi berhasil (misalnya, format nomor penerima benar), pesan akan dikirim melalui server WhatsApp ke penerima yang dituju. Jika validasi gagal, pesan tidak akan dikirim, dan Admin mungkin perlu melakukan koreksi. Diagram ini menunjukkan langkah-langkah penting dan interaksi antara Admin, sistem, dan server WhatsApp dalam proses pengiriman pesan personal. Berikut *Activity Diagram* Kirim Pesan Personal ditampilkan pada **Gambar 2**.



Gambar 2. Activity Diagram Kirim Pesan Personal

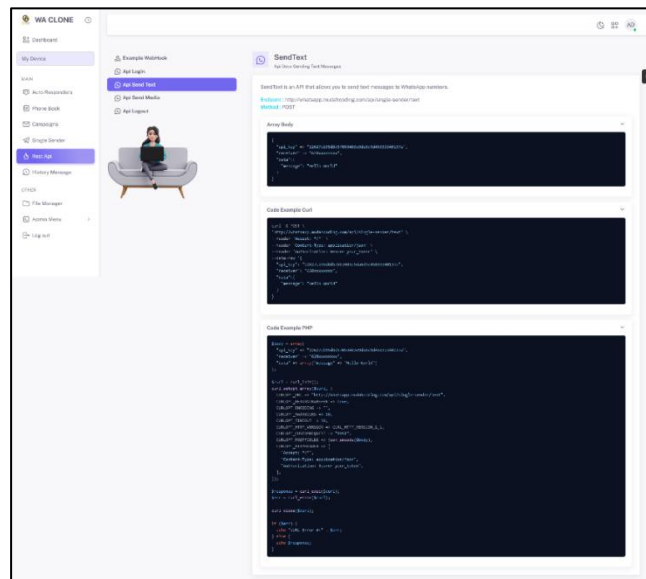
### 3.3. Implementasi Desain dan Dokumentasi API

*Single Sender Text Message* memungkinkan pengguna mengirim pesan teks langsung kepada satu penerima. Pengguna memasukkan nomor penerima, menulis pesan teks, dan mengirimkannya. Fitur ini cocok untuk komunikasi langsung dan personal antara pengguna dan penerima tanpa melalui proses yang rumit. Berikut tampilan *Single Sender Text Message* pada **Gambar 3**.



Gambar 3. *Single Sender Text Message*

Dalam menu Rest API, API *Send Text Message* memungkinkan pengguna untuk mengirim pesan teks secara otomatis melalui panggilan API. Pengguna dapat mengirim pesan teks ke nomor tujuan tertentu dengan menyediakan nomor penerima dan teks pesan dalam permintaan API. Ini memungkinkan integrasi dengan sistem lain untuk mengirimkan pesan teks tanpa interaksi manual. Berikut tampilan REST API *Send Text* pada **Gambar 4**.



Gambar 4. REST API *Send Text*

### 3.4. Pengujian

Pengujian ini dilakukan untuk memastikan bahwa berbagai fitur dan fungsionalitas dari aplikasi berjalan sesuai dengan yang diharapkan. Berikut tabel Pengujian Black Box pada **Tabel 1**.

Tabel 1. Pengujian Black Box

| Page                        | Testing                                                                                                                                                                                   | Status |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| Login                       | Memasukkan <i>username:</i> admin dan <i>password:</i> admin                                                                                                                              | Sukses |
| Add Device                  | Menambahkan nama perangkat: Joe                                                                                                                                                           | Sukses |
| Scan QR Code                | Melakukan scan dari aplikasi WhatsApp                                                                                                                                                     | Sukses |
| Setting                     | Melihat API Key perangkat                                                                                                                                                                 | Sukses |
| Setting                     | Generate ulang API Key perangkat                                                                                                                                                          | Sukses |
| Select Device               | Memilih perangkat aktif                                                                                                                                                                   | Sukses |
| Auto Responder              | 1. Menambah keyword, 2. Tipe pesan ( <i>Text</i> , media), 3. Keyword type (equal, contain), 4. Quoted (No, Yes), 5. Reply only when (all, personal, group), 6. Status (active, inactive) | Sukses |
| Detail Auto Responder       | Melihat salah satu data <i>Auto Responder</i> yang telah dibuat serta dapat melakukan perubahan data                                                                                      | Sukses |
| Phone Book                  | Menambahkan label kontak                                                                                                                                                                  | Sukses |
| Detail Phone Book           | Menambahkan kontak personal dan grup                                                                                                                                                      | Sukses |
| Campaigns                   | 1. Membuat nama kampanye dan memilih kontak, 2. Memilih tipe pesan ( <i>Text</i> , media) dan mengisi pesan, 3. Mengatur jeda pengiriman 10 detik dan waktu pengiriman                    | Sukses |
| Detail Campaigns Info       | Melihat status informasi pengiriman <i>Campaigns</i>                                                                                                                                      | Sukses |
| Detail Campaigns Bulks Info | Melihat informasi seluruh penerima pesan <i>Campaigns</i>                                                                                                                                 | Sukses |
| Single Sender Text Message  | Memasukkan nomor penerima pesan lalu memilih tipe pesan berupa teks dan input isi pesan                                                                                                   | Sukses |
| Single Sender Media Message | Memasukkan nomor penerima pesan lalu                                                                                                                                                      | Sukses |

| Page                       | Testing                                                                     | Status |
|----------------------------|-----------------------------------------------------------------------------|--------|
|                            | memilih tipe pesan berupa media kemudian upload dokumen dan input isi pesan |        |
| Rest API Login             | Melihat dokumentasi Login                                                   | Sukses |
| Rest API <i>Send Text</i>  | Melihat dokumentasi <i>Single Sender Text Message</i>                       | Sukses |
| Rest API <i>Send Media</i> | Melihat dokumentasi <i>Single Sender Media Message</i>                      | Sukses |
| Rest API Logout            | Melihat dokumentasi Logout                                                  | Sukses |
| <i>History</i>             | Melihat riwayat pengiriman pesan                                            | Sukses |
| <i>File Manager</i>        | Mengelola dokumen yang diupload                                             | Sukses |
| Admin Menu                 | Mengelola user                                                              | Sukses |
| User <i>Dashboard</i>      | Menampilkan halaman user tanpa menampilkan menu Admin Menu                  | Sukses |

### 3.5. Evaluasi

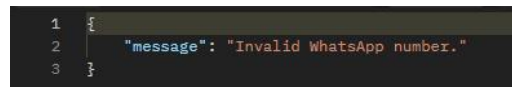
Pengiriman diproses dalam hitungan detik sekitar 2 detik saat kondisi server sedang normal, namun jika kondisi server sedang berat karena banyaknya akses maka tidak dapat dipastikan dalam 2 detik akan terkirim sehingga membutuhkan waktu yang lebih lama dari sebelumnya. Kemudian dilakukan lagi pengujian pada beberapa nomor yang terdaftar pada WhatsApp untuk mengetahui hasil pengiriman dalam waktu rata-rata yang tertera pada tabel 7. Hasil waktu rata-rata pengiriman pesan personal pada saat kondisi server normal ialah 2.24 detik. Berikut tabel Pengujian Efektivitas API WhatsApp pada **Tabel 2**.

Tabel 2. Pengujian Efektivitas API WhatsApp

| No Pengujian | Digit Nomor | Waktu Pengiriman | Status   |
|--------------|-------------|------------------|----------|
| Pengujian 1  | 10 digit    | 2.90 detik       | Berhasil |
| Pengujian 2  | 11 digit    | 1.80 detik       | Berhasil |
| Pengujian 3  | 12 digit    | 3.10 detik       | Berhasil |
| Pengujian 4  | 12 digit    | 2.06 detik       | Berhasil |
| Pengujian 5  | 12 digit    | 2.00 detik       | Berhasil |
| Pengujian 6  | 12 digit    | 1.89 detik       | Berhasil |
| Pengujian 7  | 12 digit    | 1.69 detik       | Berhasil |
| Pengujian 8  | 12 digit    | 3.29 detik       | Berhasil |
| Pengujian 9  | 12 digit    | 1.93 detik       | Berhasil |
| Pengujian 10 | 13 digit    | 1.78 detik       | Berhasil |

Kemudian pengujian yang dilakukan pada nomor yang tidak terdaftar pada WhatsApp maka berikut hasilnya pada **Gambar 5**. Yang menampilkan pesan bahwa “*Invalid WhatsApp Number*”.





```

1 {
2   "message": "Invalid WhatsApp number."
3 }

```

Gambar 5. Pengujian API Kirim Pesan Ke Nomor Tak Terdaftar

#### 4. KESIMPULAN

Hasil pengujian black box menunjukkan bahwa aplikasi berfungsi dengan baik. Pengembangan yang dilakukan telah berjalan dengan sesuai yaitu pengiriman pesan untuk WhatsApp aktif bernomor 10 angka. Fitur utama dari aplikasi ialah *Single Sender* yang dapat mengirimkan pesan berupa teks atau media secara personal. Hasil waktu rata-rata pengiriman pesan personal pada saat kondisi server normal ialah 2.24 detik.

#### 5. SARAN

Terdapat ruang untuk dilakukan peningkatan lebih lanjut mengenai aplikasi yang sudah dikerjakan meliputi penambahan fitur, penambahan antarmuka, dan cara kerja aplikasi.

#### UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada pihak-pihak yang telah mendukung terlaksananya penelitian ini.

#### DAFTAR PUSTAKA

- [1] S. Aprilia, "Sistem Informasi Absensi Berbasis Website Menggunakan API WhatsApp dengan Metodologi Incremental (Studi Kasus: SMP Negeri 29 Pekanbaru)," *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 38–44, 2020, doi: 10.30871/jaic.v4i1.1966.
- [2] D. Isnafirlah and M. Kamayani, "Information Center Chatbot in Higher Education Using Dialogflow," *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 8, no. 2, p. 393, 2023, doi: 10.24114/cess.v8i2.48186.
- [3] H. Kambivi, E. Junirianto, and N. R. Fadhliyah, "Development of Inventory Management Application Points Of Sale Using Laravel," *Tepian*, vol. 1, no. 1, pp. 9–17, 2020, doi: 10.51967/tepian.v1i1.48.
- [4] E. Putri Primawanti and H. Ali, "Pengaruh Teknologi Informasi, Sistem Informasi Berbasis Web Dan Knowledge Management Terhadap Kinerja Karyawan (Literature Review Executive Support Sistem (Ess) for Business)," *J. Ekon. Manaj. Sist. Inf.*, vol. 3, no. 3, pp. 267–285, 2022, doi: 10.31933/jemsi.v3i3.818.
- [5] F. Alfaridzi, J. Dedy Irawan, and M. Orisa, "Perancangan Sistem Manajemen User Hotspot Berbasis Web Menggunakan Application Programming Interface (API) Mikrotik," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 6, no. 2, pp. 974–981, 2023, doi: 10.36040/jati.v6i2.5412.

- [6] R. Gelar Guntara and V. Azkarin, "Implementasi dan Pengujian REST API Sistem Reservasi Ruang Rapat dengan Metode Black Box Testing," *J. Minfo Polgan*, vol. 12, no. 1, pp. 1229–1238, 2023, doi: 10.33395/jmp.v12i1.12691.
- [7] R. Maulana, A. Sulistyanto, and A. S. Rini, "Perancangan sistem informasi pengajuan dan pelaporan pembayaran tunjangan kinerja pada lembaga pemasyarakatan salemba berbasis web menggunakan Skema Rest API," *J. Manajemen Inform. Jayakarta*, vol. 1, no. 4, p. 283, 2021, doi: 10.52362/jmijayakarta.v1i4.507.
- [8] L. F. Viera Valencia and D. Garcia Giraldo, "Pemanfaatan Aplikasi Whatsapp Sebagai Sarana Diskusi Antara Pengawas Dan Guru Di Madrasah Binaan," *Angew. Chemie Int. Ed. 6(11)*, 951–952., vol. 2, no. April, pp. 47–63, 2019.
- [9] J. Kaganga, "PENGARUH TERPAAN INFORMASI HOAKS DI APLIKASI WHATSAPP TERHADAP SIKAP MAHASISWA (Studi Kuantitatif Pada Mahasiswa Jurusan Ilmu Komunikasi Universitas Bengkulu)," *Kaganga*, vol. 7, no. 1, pp. 83–91, 2023.
- [10] Fitri Annisa Purba, "Sistem Informasi Penjualan Pakaian Berbasis Web Pada TrendyStore Fitri Menggunakan Framework Laravel," *J. Comput. Digit. Bus.*, vol. 2, no. 2, pp. 57–64, 2023, doi: 10.56427/jcbd.v2i2.152.
- [11] N. L. M. E. Yuniawati, P. P. G. P. Pertama, and I. G. N. N. Bagiarta, "Sistem Informasi Pelayanan Customer Pada Wahyu Service Elektronik Berbasis Web Dengan Menggunakan Framework Laravel," *Naratif J. Nas. Riset, Apl. dan Tek. Inform.*, vol. 5, no. 2, pp. 186–197, 2023, doi: 10.53580/naratif.v5i2.213.
- [12] W. Gede Endra Bratha, "Literature Review Komponen Sistem Informasi Manajemen: Software, Database Dan Brainware," *J. Ekon. Manaj. Sist. Inf.*, vol. 3, no. 3, pp. 344–360, 2022, doi: 10.31933/jemsi.v3i3.824.
- [13] T. B. Kurniawan and Syarifuddin, "Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman Pada Cafeteria NO Caffe di TANjung Balai Karimun Menggunakan Bahasa Pemrograman PHP dan MySQL," *J. Tikar*, vol. 1, no. 2, pp. 192–206, 2020, [Online]. Available: [https://ejurnal.universitaskarimun.ac.id/index.php/teknik\\_informatika/article/download/153/121](https://ejurnal.universitaskarimun.ac.id/index.php/teknik_informatika/article/download/153/121)
- [14] D. Hermansyah and F. P. Sihotang, "Sistem Pendukung Keputusan Penentuan Staf Marketing Terbaik Menggunakan Metode SAW," *J. Teknol. Sist. Inf.*, vol. 3, no. 2, pp. 303–312, 2022, doi: 10.35957/jtsi.v3i2.3039.
- [15] D. B. Anjasmara, M. A. Rosid, and A. Eviyanti, "Implementasi Fitur Notifikasi Whatsapp API pada Sistem Manajemen Tugas Akhir," pp. 1–14, 2024.