

Implementasi Voice to Text pada Invoice Checking Berbasis Web

Gangsar Swapurba¹, Irfan Pratama²

^{1,2} Program Studi Sistem Informasi Universitas Mercu Buana Yogyakarta, Jl. Jembatan Merah No. 84.C. Gejayan, Sleman 55283, Indonesia

Info Artikel

Kata Kunci:

voice-to-text
speech recognition
web speech API
web app

Keywords:

First keyword
Second keyword
Third keyword
Fourth keyword
Fifth keyword

ABSTRAK

INDOMASCOT adalah sebuah perusahaan yang bergerak di bidang produksi kostum badut maskot. Tim penjualan sering mengalami kendala dalam pengiriman faktur kepada pelanggan, terutama saat berada di workshop. Oleh karena itu, diperlukan solusi berupa aplikasi pemeriksa faktur yang dapat meringankan beban kerja tim penjualan dalam proses pengiriman faktur. Selain itu, disarankan untuk menambahkan fitur kecerdasan buatan (AI) berupa teknologi voice-to-text, yang memungkinkan pelanggan memasukkan alamat email dan nomor faktur mereka melalui suara. Dari beberapa uji coba yang dilakukan, Web Speech API menunjukkan hasil yang kurang memuaskan dalam mengenali alamat email, tetapi berhasil dengan baik dalam mengenali nomor faktur berbentuk angka. Fitur voice-to-text ini menggunakan Web Speech API yang pada saat penulisan ini masih berstatus eksperimental.

ABSTRACT

INDOMASCOT is a company engaged in the production of mascot costumes. The sales team often faces difficulties in sending invoices to customers, especially when at the workshop. Therefore, a solution in the form of an invoice checking application is needed to reduce the workload of the sales team in the invoice delivery process. Additionally, it is suggested to implement artificial intelligence (AI) technology in the form of voice-to-text to allow customers to input their email addresses and invoice numbers through voice. Based on several trials, the Web Speech API showed unsatisfactory results in recognizing email addresses but performed well in recognizing numerical invoice numbers. This voice-to-text feature utilizes the Web Speech API, which is still experimental at the time of writing.

This is an open access article under the [CC BY](#) license.



Corresponding Author:

Irfan Pratama
Email: irfanp@mercubuana-yogya.ac.id

1. PENDAHULUAN

INDOMASCOT adalah sebuah perusahaan kecil yang berfokus pada produksi kostum maskot, yang berlokasi di Bandung. Perusahaan ini memberikan kontribusi besar pada divisi komunikasi pemasaran (marcom) berbagai perusahaan dengan memproduksi kostum maskot yang digunakan untuk promosi dan sosialisasi produk atau layanan. Salah satu masalah yang saat ini dihadapi oleh tim sales adalah proses pembuatan dan pengiriman invoice yang diminta oleh pelanggan ketika mereka sedang berada di lapangan. Dalam penelitian berjudul "Penerapan Web Speech API pada Game Catur Berbasis Suara," disebutkan bahwa aplikasi yang dikembangkan dalam penelitian ini berfokus pada teknologi input dengan memanfaatkan Web Speech API dari aplikasi catur yang sudah ada, yang memungkinkan pengguna untuk menggerakkan bidak hanya dengan menggunakan suara baik dalam Bahasa Indonesia maupun Bahasa Inggris. Pengguna dapat

memberikan perintah suara dalam permainan melawan AI atau secara online antar pengguna, menggunakan notasi catur standar atau kalimat biasa, seperti "Gxe5" atau "Gajah makan pion," sesuai dengan langkah yang tersedia [1][11]. Penelitian serupa, "Voice Command Pengendali Perangkat Elektronik Rumah Tangga Menggunakan Raspberry Pi," menyimpulkan bahwa aplikasi perintah suara berhasil dibuat untuk mengendalikan perangkat elektronik rumah tangga. Pengujian menunjukkan bahwa perintah suara dapat digunakan untuk mengendalikan perangkat elektronik dengan baik dan tanpa kendala [2][12]. Dalam penelitian berjudul "Perancangan Aplikasi Web Berbasis Usability," disarankan bahwa pengembangan website harus didasarkan pada kebutuhan pengguna atau pengalaman pengembang. Mengingat banyaknya website yang ada, desain antarmuka harus memperhatikan pengalaman pengguna (User Experience) untuk memastikan situs web mudah digunakan dan efektif bagi pengunjung. Beberapa hal yang perlu dipertimbangkan dalam perancangan website termasuk desain antarmuka dan kegunaan (usability) untuk memastikan pengguna dapat mencapai tujuan yang diharapkan dengan kepuasan [3][13].

Speech recognition (pengenalan wicara) adalah metode atau teknik yang digunakan komputer untuk mengenali ucapan manusia dan mengubahnya menjadi teks. Komputer menangkap sinyal suara dan mengkonversinya menjadi data yang dapat dipahami oleh komputer, yang kemudian digunakan untuk mengeksekusi satu atau lebih perintah yang telah diprogram sebelumnya. *Speech recognition* juga dikenal sebagai *automatic speech recognition* (ASR) atau *speech-to-text* (STT) [4, 14-15].

Web Speech API adalah API eksperimental yang memungkinkan kita memasukkan suara kita ke dalam aplikasi yang kita kembangkan. Web Speech API memiliki dua komponen utama:

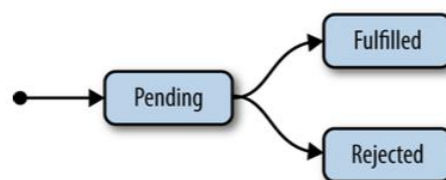
1. Interface Speech Recognition, yang mampu mengenali suara dari perangkat input seperti mikrofon bawaan atau handsfree dan merespons sesuai dengan yang telah ditentukan oleh programmer.
2. Interface SpeechSynthesis, yang mampu membaca teks dan mengubahnya menjadi suara.

Web Speech API ini dapat dianggap sebagai salah satu penerapan AI (Artificial Intelligence) dalam sebuah mesin karena memanfaatkan suara manusia untuk diproses dan mendapatkan hasil yang diinginkan oleh pengguna. Saat ini, Web Speech API didukung dan dapat digunakan oleh browser Google Chrome versi 33 ke atas, Mozilla Firefox versi 49 ke atas, dan Microsoft Edge (hanya *speech synthesis*). Web Speech API juga mendukung berbagai bahasa termasuk Bahasa Indonesia dengan menggunakan mesin Microsoft Speech Synthesis dan Google Speech Synthesis. Untuk memudahkan pemahaman, API dapat diibaratkan sebagai gerbang yang memungkinkan seseorang untuk menambahkan, mengubah, menghapus, atau melihat barang-barang di dalamnya, yang dikenal dengan istilah CRUD (Create, Read, Update, Delete) [6].

World Wide Web (www) atau yang dikenal juga sebagai web, site, website, atau situs adalah aplikasi dan layanan internet yang mencakup sumber daya multimedia. Secara fisik, web adalah kumpulan komputer pribadi, web browser, koneksi ke ISP, komputer server, router, dan switch yang digunakan untuk mengalirkan informasi. Secara umum, website dikategorikan menjadi beberapa jenis, yaitu:

1. Web search engine, yang memiliki kemampuan untuk mencari dokumen berdasarkan kata kunci tertentu, seperti Google dan Alltheweb.
2. Web portal, yang berisi kumpulan link, search engine, dan informasi, seperti Yahoo dan AOL.
3. Web perusahaan, yang menggambarkan profil perusahaan, layanan, fasilitas, dan segala sesuatu tentang perusahaan tersebut, seperti Telkomsel dan Smartfren [7].

Promise adalah salah satu konsep yang muncul di ES6 (ES2015). Konsep ini hadir untuk mengatasi masalah dalam proses asynchronous yang sebelumnya rumit. Sebelumnya, kita menggunakan callback untuk menangani proses asynchronous, tetapi lama kelamaan kita akan merasa kesulitan ketika callback yang digunakan semakin banyak, bahkan bisa terjadi callback di dalam callback (callback hell) [8].



Gambar 1. State dalam Promise

Gambar 1 merupakan *Promise*, dimana *Promise* memiliki 3 state yang sering digunakan, yaitu: *Pending* (sedang dalam proses), *Fulfilled* (berhasil), dan *Rejected* (gagal). *Fetch* adalah bentuk *Promise* yang paling

umum digunakan sejak hadirnya *callback ES6* yang mana membuat kode penerapan promise jadi lebih humanis, seperti yang ditunjukkan pada Gambar 2.

```
fetch('https://jsonplaceholder.typicode.com/users/1')
  .then(function(response) { // #1
    | return response.json();
  })
  .then(function(user) { // #2
    | console.log(user);
  })
```

Gambar 2. Contoh *Fetch*

Istilah pengalaman pengguna (User Experience, UX) pertama kali diperkenalkan oleh Don Norman, seorang akademisi dalam bidang ilmu kognitif, desain, dan rekayasa kegunaan, yang juga pernah menjabat sebagai wakil presiden Apple Inc. Dia adalah yang pertama menggagas pendekatan desain yang berfokus pada kebutuhan dan keinginan pengguna, yang kemudian dikenal sebagai desain berpusat pada pengguna (user-centered design). Seiring waktu, penerapan konsep UX telah berkembang luas, mencakup bidang produk industri, sistem, hingga layanan/jasa [9].

Selain itu, dengan melihat kemajuan zaman yang semakin dipengaruhi oleh teknologi kecerdasan buatan (AI), perusahaan ini juga ingin mengikuti perkembangan tersebut. Oleh karena itu, perusahaan merasa perlu untuk menerapkan AI sebagai solusi atas permasalahan yang sedang dihadapi. Dengan demikian, akan sangat bermanfaat jika terdapat fitur yang mampu mengimplementasikan AI secara mudah dan memberikan manfaat signifikan bagi pengguna.

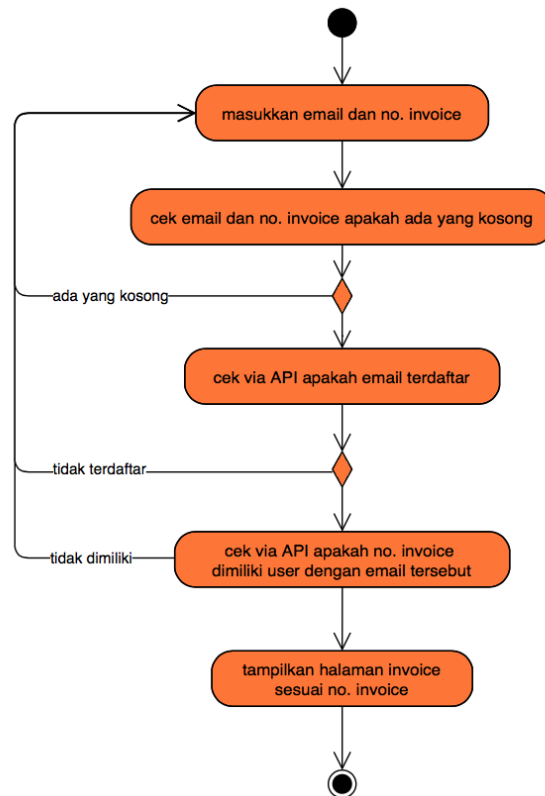
4. METODE

Dalam penelitian pembuatan aplikasi *invoice checking* ini peneliti menggunakan metode *ADCT (Analysis, Design, Coding & Testing)* dengan langkah-langkah sebagai berikut:

1.1 Analisis

Pada fase ini, ditetapkan kebutuhan fungsional dari aplikasi yang dibuat dan didefinisikan alur kerja aplikasi, seperti ditunjukkan pada Gambar 3, yaitu:

1. *Fetching customer data by email* via *API*.
2. *Fetching* daftar *invoice* by no. *invoice* via *API*.
3. Menampilkan *URL* detail *invoice*.



Gambar 3. Diagram Alur Umum Kerja Aplikasi

1.2 Desain

Membuat *mockup* desain terutama *user interface* untuk mendapatkan visualisasi aplikasi yang akan dibuat mulai dari *Input*, *Recording Session*, *Validation*, hingga *Output Redirection*.



Gambar 4. User Interface Mockup Design

1.3 Pengkodean

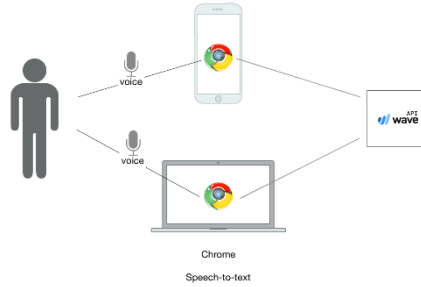
Pada penelitian ini memanfaatkan *JavaScript ES6* karena telah dilengkapi fitur *fetch* yang berguna untuk mengambil data dari *API*. Pada penelitian ini juga memanfaatkan *CSS3* yang sudah mendukung fitur *animation* untuk digunakan pada halaman *redirecting* khususnya pada elemen *spinning icon*. *Source code* pada penelitian ini memanfaatkan *HTML5* yang paling terkini yang membuat *source code* semakin mudah dibaca apalagi jika nantinya *development* akan diteruskan atau berkolaborasi dengan developer lain, salah satunya karena sudah tersedianya elemen seperti *header*, *section* & *footer* yang sangat *readable*.

1.4 Testing

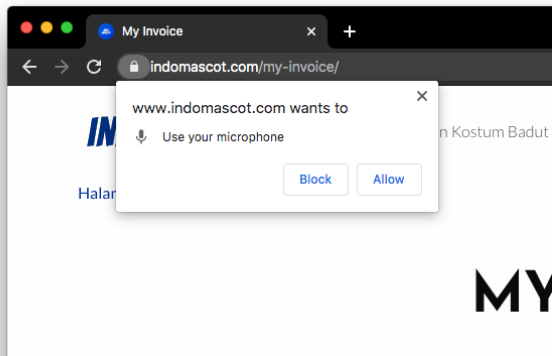
Setelah sistem selesai dibangun, maka harus diuji apakah sistem dapat berjalan baik. Pengujian dilakukan dengan menggunakan metode *black-box testing* [10].

5. HASIL DAN DISKUSI

Gambar 5 menunjukkan proses input suara, karena selain input via text, aplikasi ini dapat menangkap input via suara dengan memanfaatkan Web Speech API. Di sini, user diminta untuk memberikan hak akses microphone untuk digunakan meng-input suaranya, seperti yang ditunjukkan pada Gambar 6.



Gambar 5. *Input by Voice*



Gambar 6. Permintaan Hak Akses *Microphone*

Kemudian setelah berhasil mendapatkan data email dan no. *invoice*, selanjutnya adalah memproses data-data tersebut di proses *authentication* di mana harus cocok kedua-duanya dengan yang sudah tersimpan. Sesi *recording* diawali dengan dikliknya tombol *recording* berikon *mic* di sebelah kolom input email & no. *invoice*, seperti yang ditunjukkan pada Gambar 7. Hasil user yang valid dari proses *authentication* pada saat *testing* disajikan dalam Tabel1.



Gambar 7. Halaman *My Invoice* & Sesi *Recording*

Tabel 1. Daftar *Valid User* Untuk *Testing*

No	Email	No. Invoice
1	anastasia.carolina@jny.sch.id	457
2	adzkiawidodo@gmail.com	112
3	alexandermario@hotmail.co.id	108
4	sentulpm@astonhotelsinternational.com	255
5	purchasing2@milko.co.id	126
6	fadel.muhammed@nussaofficialstore.com	364
7	arifin.siregar@f7ame.com	330
8	procurement@quickchicken.id	358
9	dimas@ardgroup.co.id	317
10	dwicahyono@cemaragroup.org	138
11	felix@monotaro.id	169
12	aswin@bigo.tv	259
13	kevin_pratama@bcainsurance.co.id	336
14	info@eomik.co.id	111
15	bhukari.fauzul@bps.go.id	385
16	mochammad.marsidiq@marriott-hotels.com	229
17	yulia.rosiana@carmudi.co.id	16
18	purchasing@novotelbandung.com	321
19	daya.aruna@bsn.go.id	113
20	suriani@dyandra.com	74
21	leonard.sinaga@grandtjokro.com	231
22	andhita.apriyanti@aeonindonesia.co.id	135
23	frisca.juliana@ef.com	246
24	mira@findmydash.com	130
25	sylvia.irene@ultrajaya.co.id	167

Interception di proses input *validation* ini menguntungkan sekali bagi semua pihak karena dapat menghemat *bandwidth & resources* yang dari pada membiarkan input data yang belum divalidasi tersebut dibiarkan berlalu-lalang memakan *bandwidth* dan juga memerlukan *resources* untuk memprosesnya di server. Imbas positif dari *validation* ini juga akan dirasakan oleh *visitor* dalam bentuk *loading* halaman maupun data yang lebih cepat. Sehingga tidak perlu berulang kali melakukan *full loading* halaman setiap validasi berlangsung di mana hal ini berdampak baik pula terhadap *user experience*-nya. Adapun alur validasinya sebagai berikut, yang dapat disajikan dalam Gambar 8:

1. Jika email dan atau no. *invoice* belum diisi, maka kedua-duanya harus diisi terlebih dahulu.
2. Jika email dengan no. *invoice* tidak cocok, maka tidak bisa, harus cocok terlebih dahulu.



Gambar 8. Input Validation

Setelah melalui proses validasi, maka saatnya diberikan informasi yang diinginkan dengan melewati proses *redirecting* menuju halaman detail *invoice* yang bersangkutan, seperti yang ditunjukkan pada Gambar 9.



Gambar 9. Redirecting Ke Halaman Detail Invoice

Menurut halaman resmi dari *Web Speech API* sendiri, *API* ini hanya dapat mengenali suara dari dari daftar kosakata yang sudah dikembangkan oleh mereka sendiri. Dan menurut mereka, kosakata tersebut yang banyak dipakai ketika menggunakan fitur *speech recognition* ini. Melihat hal ini, perlu mencoba seberapa akurat *API* ini bekerja. Baik dalam mengenali kosakata baik itu yang berupa angka maupun kata. Percobaan dilakukan dengan sumber suara yang sama. Hasilnya tidak memuaskan untuk mengenali alamat email. Tabel 2 menyajikan hasil 10 kali percobaan *Voice-to-text*.

Tabel 2. Hasil *Voice-to-text* Pada 10 Kali Percobaan

No	Suara	Hasil yang diharapkan	Akuras i	Hasil lain yang muncul
1	felix at monotaro dot id	felix@monotaro.id	0%	velg monotaro.id Felix feat monotaro.id Kalisat monotaro.id Felix monotaro.id mixed monotaro.id helix@monotaro.id
2	kph dot indonesia at gmail dot com	kph.indonesia@gmail.com	60%	kph indonesia@gmail.com

No	Suara	Hasil yang diharapkan	Akurasi	Hasil lain yang muncul
3	suriani at dyandra dot com	suriani@dyandra.com	0%	suryani@gmail.com suriani at Diandra dotcom

Lalu mencoba untuk mencari tahu apa yang terjadi ketika dimasukkan alamat email yang lazim banyak ditemui untuk nama-nama orang luar negeri seperti pada Tabel 3.

Tabel 3. Hasil Input Email Nama-Nama Inggris Pada 10 Kali Percobaan

No	Suara	Hasil yang diharapkan	Akurasi	Hasil lain yang muncul
1	bob at facebook dot com	bob@facebook.com	0%	bob at facebook.com map at facebook.com
2	kim at yahoo dot com	kim@facebook.com	0%	kim at yahoo.com came at yahoo.com frank at google.com
3	frank at google dot com	frank@google.com	0%	frank app google.com frank at google account
4	jim at apple dot com	jim@apple.com	0%	gym at apple.com jim at apple.com gym at appalachian

Sebelumnya, telah memberikan nilai untuk *property SpeechRecognition.lang* menjadi "en-US" agar dapat mengenali suara dengan aksent *American English*. Namun, hasilnya tetap sama saja yaitu *Web Speech API* tidak dapat mengenali email sama sekali. Namun, ada kabar baiknya yaitu *Web Speech API* ini dapat mengenali angka dengan baik, seperti yang ditunjukkan pada Tabel 4.

Tabel 4. Hasil Input No. Invoice Via Voice-to-text Pada 10 Kali Percobaan

No	Suara	Hasil yang diharapkan	Akurasi
1	empat lima enam	456	100%
2	empat empat enam	446	100%
3	tiga delapan lima	385	100%
4	tiga lima delapan	358	100%
5	tiga lima tujuh tiga ratus lima puluh	357	100%
6	dua tiga ratus lima puluh	352	100%
7	tiga	353	100%
8	tiga ratus dua puluh dua	322	100%
9	tiga ratus enam	306	100%
10	seratus lima puluh enam	156	100%
11	seratus lima puluh lima	155	100%

Hasil pengujian sistem menggunakan metode *Black-Box Testing*, dapat disajikan dalam Tabel 5.

Tabel 5. *Black-Box Testing*

No	Kasus Uji	Skenario	Hasil yang didapat	Keterangan
1	Melakukan <i>input by voice</i>	<i>Visitor</i> mengaktifkan <i>voice-to-text</i> pada kolom input email dan melakukan sesi <i>recording</i>	Menampilkan text di kolom input email hasil penangkapan <i>voice-to-text</i>	Tidak berhasil (hasilnya selalu salah)
		<i>Visitor</i> mengaktifkan <i>voice-to-text</i> pada kolom input <i>invoice</i> dan melakukan sesi <i>recording</i>	Menampilkan text di kolom input no. <i>invoice</i> hasil penangkapan <i>voice-to-text</i>	Berhasil
2	Melakukan validasi input	Kolom input email kosong atau email tidak terdaftar	Menampilkan <i>alert</i> pesan <i>error</i> kesalahan pada kolom input email	Berhasil
		Kolom input no. <i>invoice</i> kosong atau no. <i>invoice</i> tidak terdaftar	Menampilkan <i>alert</i> pesan <i>error</i> kesalahan pada kolom input no. <i>invoice</i>	Berhasil
3	Menampilkan halaman detail <i>invoice</i>	Email yang diinput benar-benar ada dimiliki oleh salah satu <i>customer</i> dan no. <i>invoice</i> yang diinput memang dimiliki oleh <i>customer</i> tersebut	Menampilkan halaman detail <i>invoice</i>	Berhasil

6. KESIMPULAN

Penelitian ini menghasilkan beberapa kesimpulan, yaitu:

1. Aplikasi invoice checking berbasis web yang dihasilkan dari penelitian ini memungkinkan customer untuk dapat melakukan pengecekan invoice-nya secara mandiri. Dengan begitu, diharapkan pula dapat mengurangi beban pekerjaan tim sales karena tidak perlu lagi ada pekerjaan mengirimkan invoice kepada customer.
2. Dengan diimplementasikannya AI dalam bentuk voice-to-text di dalam aplikasi ini, dapat menjadi bentuk alternatif input customer untuk menggunakan suaranya. Meskipun, fitur voice-to-text ini tidak memuaskan dalam mengenali email, namun hasilnya memuaskan untuk mengenali no. invoice.

Ada beberapa saran untuk memperbaiki penelitian ini kedepannya, yaitu:

1. Menonaktifkan voice-to-text untuk input email sampai ada update Web Speech API dapat mengenali input email dengan baik.
3. Menggunakan protokol HTTPS untuk mengatasi celah keamanan yang memanfaatkan full access token GraphQL.
4. Backup data & persiapan migrasi layanan invoicing karena telah ada peringatan bahwa layanan invoicing yang digunakan saat ini hanya akan berfokus pada user yang berbasis di US & Canada saja.
5. Migrasi dari Web Speech API ke layanan berbayar voice-to-text yang lebih baik lagi mengenali kosakatanya, lebih baik lagi dalam meminta bantuannya via premium support-nya, dan lebih luas pula cakupan browser compatibility-nya. Memberikan pernyataan bahwa apa yang diharapkan sebagaimana tertuang pada bab “Pendahuluan” pada akhirnya dapat menghasilkan bab “Hasil dan Pembahasan”, sehingga terdapat kesesuaian. Selain itu juga dapat ditambahkan prospek pengembangan hasil penelitian dan prospek penerapan penelitian lanjutan ke tahap berikutnya (berdasarkan hasil dan pembahasan).

REFERENSI

- [1] I. Gede Hendra Parwata, I. Ketut Gede Darma Putra, dan N. Putu Sutramiani, “Penerapan Web Speech API pada Game Catur Berbasis Suara,” *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 7, no. 1, hal. 21, 2019, doi: 10.24843/jim.2019.v07.i01.p03.
- [2] M. A. Latief, “Voice Command Pengendali Perangkat Elektronik Rumah Tangga menggunakan Raspberry Pi,” *Tek. Inform. Fak. Sains dan Teknol. UIN Sunan Kalijaga, Yogyakarta*, 2015.

- [3] T. F. Tambuwun, R. Sengkey, Y. D. Y. Rindengan, T. Sam, dan R. Manado, "Perancangan Aplikasi Web Berbasis Usability," *J. Tek. Inform.*, vol. 12, no. 1, 2017, doi: 10.35793/jti.12.1.2017.17856.
- [4] F. Aulia, F. Baskoro, D. T. Informatika, F. Teknologi, dan I. Teknologi, "Rancang Bangun Interactive Voice Response (IVR) Sebagai Pengingat Medikasi Pasien Lansia Berbasis Web," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 1, 2018.
- [5] M. Contributors, "Web Speech API - Web APIs | MDN." https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API (diakses Sep 30, 2020).
- [6] Feridi, "Mengenal RESTful Web Services - CodePolitan.com." <https://www.codepolitan.com/mengenal-restful-web-services> (diakses Sep 18, 2020).
- [7] J. T. Informatika, "Jurnal Teknik Informatika, Vol. 10, No. 2, April 2018," vol. 10, no. 2, hal. 27–32, 2018.
- [8] S. Nasution, "Memahami lebih dalam konsep Promise di JavaScript | Sutan Nasution." <https://sutanlab.id/blog/memahami-konsep-promise-di-javascript> (diakses Des 17, 2020).
- [9] J. Saptari, R. Iswandari, dan R. Setyawati, "User Experience (UX) dalam pemanfaatan fasilitas Informal Learning Space (ILS) perpustakaan," *Berk. Ilmu Perpust. dan Inf.*, vol. 14, no. 1, hal. 26, 2018, doi: 10.22146/bip.27502.
- [10] P. Santi, R. Afwani, M. Albar, S. Anjarwani, and A. Z. Mardiansyah, "Black Box Testing with Equivalence Partitioning and Boundary Value Analysis Methods (Study Case: Academic Information System of Mataram University)," 2022. doi: 10.2991/978-94-6463-084-8_19.
- [11] P. Bhandari, "Interactive Voice Response with Speech Recognition," 2017. doi: 10.13140/RG.2.2.16148.60802.
- [12] R. Corkrey and L. Parkinson, "Interactive voice response: Review of studies 1989–2000," *Behavior Research Methods, Instruments, & Computers*, vol. 34, pp. 342-353, 2002. doi: 10.3758/BF03195462.
- [13] M. Dholvan and S. Anitha, "Speech Enabled Interactive Voice Response Systems - Review," vol. IX, pp. 73-81, 2015.
- [14] H. Lee, M. Friedman, P. Cukor, and D. Ahern, "Interactive voice response system (IVRS) in health care services," *Nursing Outlook*, vol. 51, pp. 277-283, 2003. doi: 10.1016/S0029-6554(03)00161-1.
- [15] I. Inam, A. Azeta, and O. Daramola, "Comparative analysis and review of interactive voice response systems," pp. 1-6, 2017. doi: 10.1109/ICTAS.2017.7920660.