



Implementasi *Speech Recognition* Menggunakan API IBM Watson Pada Media Pembelajaran Bahasa Inggris

Acep Abdul Basit¹, R. Reza El Akbar²

¹Program Studi Informatika, Fakultas Teknik, Universitas Siliwangi, Indonesia

¹acep.abdul.basit14.student@unsil.ac.id, ²reza@unsil.ac.id

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 28-03-2021

Revisi Akhir: 30-03-2021

Diterbitkan Online: 31-03-2021

KATA KUNCI

English,
Speech Recognition,
API,
IBM Watson.

KORESPONDENSI

Telepon: +6282121734274

E-mail: acep.abdul.basit14.student@unsil.ac.id

ABSTRACT

Lack of English skills is a major problem, especially for students in Indonesia. Someone sometimes lacks confidence to speak English because they don't know how to pronounce it. A voice recognition application that provides features of how to pronounce words and sentences in good and correct English is the solution proposed in this study. Implementing the IBM Watson API on Speech Recognition is proposed to help resolve this issue. The user's voice recognition process of spoken phrases is assessed and the results are displayed. Based on the results of experiments conducted in this study, the IBM Watson API was successfully implemented in an English voice recognition application. Tests carried out in this study showed the level of feasibility of the application developed to be used got a score of 82%. This means the application is very feasible to use.

1. PENDAHULUAN

Pembelajaran bahasa Inggris merupakan salah satu hal yang perlu diperhatikan dalam dunia pendidikan saat ini. Penguasaan bahasa Inggris sangat dibutuhkan agar dapat menyerap perkembangan ilmu pengetahuan dan teknologi secara global. Pembelajaran bahasa Inggris di Indonesia masih menemui berbagai kendala. Penguasaan bahasa Inggris tamatan pendidikan dasar di Indonesia masih rendah dibandingkan dengan negara tetangga, seperti Malaysia dan Singapura [12]. Kegagalan penguasaan bahasa Inggris ini dipengaruhi oleh berbagai faktor, diantaranya: lingkungan, budaya, ekonomi, latar belakang keluarga, fasilitas pendidikan, sikap siswa, serta orang tua. Hasil ujian nasional tahun 2009, mata pelajaran bahasa Inggris berada di peringkat bawah dibanding mata pelajaran lainnya [10]. Solusi untuk menangani kurangnya pemahaman siswa-siswi sekolah terhadap bahasa Inggris diantaranya dapat dilakukan dengan melakukan pembelajaran melalui alat bantu *smartphone*. Pada era modern ini, *smartphone* merupakan salah satu perangkat yang umum digunakan oleh pelajar. Android merupakan salah satu sistem operasi yang banyak digunakan pada

smartphone saat ini. Android menyediakan berbagai layanan, termasuk layanan pengenalan suara manusia (*Speech Recognition*).

Speech Recognition merupakan subbidang interdisipliner ilmu komputer dan komputasi linguistik yang mengembangkan metodologi dan teknologi yang memungkinkan pengenalan dan terjemahan bahasa lisan ke dalam teks oleh komputer. Melalui teknik ini pengguna dapat dengan mudah melakukan sebuah pencarian dengan menggunakan perintah suara.

Suara merupakan getaran yang merambat sebagai gelombang akustik, melalui media transmisi seperti gas, cair atau padat. Dalam fisiologi dan psikologi manusia, suara adalah penerimaan gelombang yang dipersepsikan oleh otak.

Tujuan dari penelitian ini adalah menerapkan API IBM Watson pada media pembelajaran bahasa Inggris. Aplikasi dirancang agar dapat menyediakan sistem penilaian tiap pencapaian pada setiap tingkatan.

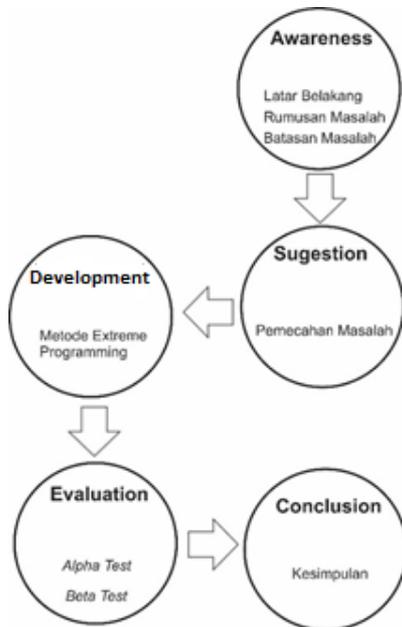
2. ULASAN PENELITIAN TERKAIT

Penelitian [1] mencoba menerapkan *speech recognition* untuk pengenalan nama hewan berbasis android. Hasil percobaan yang dilakukan, aplikasi ini dapat digunakan orang tua untuk membantu anak mengenali hewan di sekitar. Pada penelitian [2] digunakan algoritma untuk mengubah sinyal suara yang berbasis waktu menjadi berbasis frekwensi dengan membagi masalah menjadi beberapa sub masalah yang lebih kecil. Setiap sub masalah diselesaikan dengan cara melakukan pencocokan suara.

Penelitian [3] mencoba memecahkan masalah dari kurangnya keinginan belajar anak serta menerapkan pemberian poin dari soal yang dijawab oleh anak-anak sehingga lebih tertantang. Pada penelitian [4] Algoritma Hidden Markov digunakan dalam pencarian pengenalan huruf dalam sistem pengenalan suara. Pada penelitian [5] *Google speech* coba diintegrasikan pada fitur soal. Pengguna diminta untuk menjawab setiap soal menggunakan suara. *Google speech* diterapkan untuk mengkonversi suara menjadi teks.

3. METODOLOGI

Penelitian ini menggunakan metode *Design and Creation*, yang merupakan penggabungan antara metodologi penelitian dan metodologi pengembangan aplikasi [2]. Metode ini menggunakan proses iteratif[11] yang terdiri dari lima langkah seperti tidampilkan pada gambar 1.



Gambar 1. Metode *Design And Creation*

Awareness, pada tahap ini ditentukan masalah utama yang ditemui pada penelitian. *Sugestion* merupakan ide kreatif yang diusulkan untuk menyelesaikan masalah melalui pengumpulan data dan menganalisis kebutuhan. *Development* merupakan tahapan pengembangan aplikasi sesuai dengan rancangan yang telah dibuat sebelumnya. *Evaluation* merupakan tahapan memeriksa kesesuaian hasil akhir penelitian dengan tujuan penelitian. *Conclusion* adalah tahapan penarikan kesimpulan dari hasil penelitian yang telah dilakukan.

4. HASIL DAN PEMBAHASAN

1. Pengumpulan Data

a. Analisis Kebutuhan Data

Data yang dibutuhkan dalam pembangunan aplikasi ditampilkan pada tabel 1.

Tabel 1. Analisis Kebutuhan Data

Nama Data	Atribut
Data Input	Data input berupa suara yang akan di ucapkan oleh <i>user</i>
Data Proses	Data yang diproses berupa suara yang akan diolah menjadi <i>text</i>
Data Output	Data output menghasilkan <i>text</i> dari perekaman suara pada data <i>input</i> .

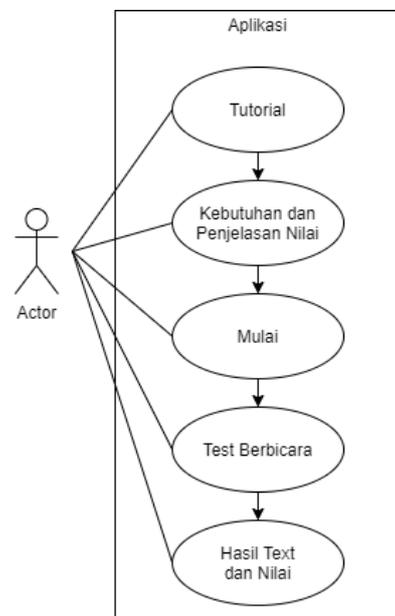
b. Identifikasi masalah

Masalah utama pada penelitian ini, bagaimana proses pelafalan bahasa inggris yang diinputkan oleh user dapat ditampilkan pada bagian *output* berupa *text*, hasil dari pelafalannya itu sendiri mempunyai beberapa tahapan nilai.

c. Kebutuhan Keluaran

Keluaran yang diharapkan yaitu berupa nilai terbagi menjadi 3 golongan yaitu: sangat bagus, bagus, dan kurang bagus. Sangat bagus didapatkan jika hasil dari nilai output $\geq 0,8$. Bagus didapatkan jika nilai output $\geq 0,5 - 0,79$. Kurang Bagus didapatkan jika nilai output $\leq 0,49$.

2. Perancangan (*Design*)



Gambar 2. *Use Case Diagram*

Gambar 2 menunjukkan tahapan proses pengenalan suara yang dirancang serta menggambarkan hubungan antara user dengan aplikasi pada use case. Pada *use case diagram* dalam proses tersebut *user* dapat masuk ke menu *tutorial* terlebih dahulu atau langsung ke menu mulai yang akan dilanjutkan ke menu *test* berbicara untuk melakukan

test berbicara. User dapat melihat hasil berupa *text* dan nilai yang diperoleh.

3. Pengkodean (Coding)

Tabel 2 menampilkan *source code* utama yang diterapkan dengan menggunakan *model view controller* dengan bahasa pemrograman C#.

Tabel 2. API IBM Watson Source Code

```
C#
#pragma warning disable 0649
using UnityEngine;
using System.Collections;
using System.Collections.Generic;
using UnityEngine.UI;
using IBM.Watson.SpeechToText.V1;
using IBM.Cloud.SDK;
using IBM.Cloud.SDK.Authentication;
using IBM.Cloud.SDK.Authentication.Iam;
using IBM.Cloud.SDK.Utilities;
using IBM.Cloud.SDK.DataTypes;
using UnityEngine.SceneManagement;
namespace IBM.Watson.Examples
{
    public class ExampleStreaming :
    MonoBehaviour
    {
        #region PLEASE SET THESE VARIABLES
        IN THE INSPECTOR
        [Space(10)]
        [Tooltip("The service URL
        (optional). This defaults to
        \"https://api.us-south.speech-to-
        text.watson.cloud.ibm.com/\")]
        [SerializeField]
        private string _serviceUrl;
        [Tooltip("Text field to display the
        results of streaming.")]
        public Text ResultsField;
        public Text textNilai;
        public Animator anim;

        [Header("IAM Authentication")]
        [Tooltip("The IAM apikey.")]
        [SerializeField]
        private string _iamApiKey;

        [Header("Parameters")]
        //
        https://www.ibm.com/watson/developercloud/s
        peech-to-text/api/v1/curl.html?curl#get-
        model
    }
}
```

4. Pengujian (Testing)

a. Alfa Testing

Spesifikasi *software* dan *hardware* yang digunakan saat pengujian aplikasi ditampilkan pada tabel 3.

Tabel 3 Spesifikasi *Software* dan *Hardware* yang digunakan

No	Spesifikasi <i>Software</i> dan <i>Hardware</i>
1	OS Android 10
2	Chipset Qualcomm SM7125 Snapdragon 720G (8 nm)
3	CPU Octa-core (2 x 2,3 Ghz & 6 x 1,8 Ghz)
4	Internal 128 GB, RAM 8 GB
5	Display 6,67 inches” Resolution 1080 x 2400 pixels, 20:9 ratio

Pengujian alpha diawali dengan menguji halaman awal aplikasi seperti ditampilkan pada Gambar 3.



Gambar 3. Halaman Awal

Halaman awal merupakan tampilan awal dari aplikasi setelah dibuka. Terdapat gambar yang menunjukkan logo dan *text* nama aplikasi dan tombol *speaking test* untuk melanjutkan ke menu berikutnya. Hasil pengujian halaman awal ditampilkan pada Tabel 4.

Tabel 4. Pengujian Halaman Menu Awal

Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
Menunggu loading dan mengklik tombol <i>Speaking Test</i>	Aplikasi akan menampilkan <i>loading</i> selama kurang lebih 3 detik kemudian akan menampilkan tombol <i>Speaking Test</i>	Aplikasi menampilkan <i>loading</i> dan menampilkan tombol <i>Speaking Test</i>	[√] diterima [] ditolak



Gambar 4. Halaman Tutorial 1

Tabel 5 Pengujian Halaman Menu Tutorial 1

Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
Mengklik / memilih tombol panah ke kanan	Aplikasi akan menampilkan halaman tutorial 2	Aplikasi menampilkan halaman tutorial 2	[√] diterima [] ditolak

Halaman tutorial 1 merupakan halaman penjelasan kebutuhan aplikasi. Pada halaman ini diberikan keterangan bahwa aplikasi membutuhkan koneksi internet.



Gambar 5 Halaman Tutorial 2

Tabel 6 Pengujian Halaman Menu Tutorial 2

No	Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
1	Mengklik / memilih tombol panah ke kanan	Aplikasi akan menampilkan halaman tutorial 3	Aplikasi menampilkan halaman tutorial 3	[√] diterima [] ditolak
2	Mengklik / memilih tombol panah ke kiri	Aplikasi akan menampilkan halaman tutorial 1	Aplikasi menampilkan halaman tutorial 1	[√] diterima [] ditolak

Halaman tutorial 2 merupakan penjelasan keterangan nilai. User akan mendapatkan nilai sangat bagus jika mendapatkan angka $\geq 0,8$.



Gambar 6 Halaman Tutorial 3

Tabel 7 Pengujian Halaman Menu Tutorial 3

No	Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
1	Mengklik / memilih tombol panah ke kanan	Aplikasi akan menampilkan halaman tutorial 4	Aplikasi menampilkan halaman tutorial 4	[√] diterima [] ditolak
2	Mengklik / memilih tombol panah ke kiri	Aplikasi akan menampilkan halaman tutorial 2	Aplikasi menampilkan halaman tutorial 2	[√] diterima [] ditolak

Halaman tutorial 3 merupakan halaman penjelasan keterangan nilai. User akan mendapatkan nilai bagus jika mendapatkan angka $\geq 0,5 - 0,79$.



Gambar 7 Halaman Tutorial 4

Tabel 8 Pengujian Halaman Menu Tutorial 4

No	Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
1	Mengklik / memilih tombol panah ke kanan	Aplikasi akan menampilkan halaman tutorial 5	Aplikasi menampilkan halaman tutorial 5	[√] diterima [] ditolak
2	Mengklik / memilih tombol panah ke kiri	Aplikasi akan menampilkan halaman tutorial 3	Aplikasi menampilkan halaman tutorial 3	[√] diterima [] ditolak

Halaman tutorial 4 merupakan halaman penjelasan keterangan nilai. User akan mendapatkan nilai kurang bagus jika mendapatkan angka $\leq 0,49$.



Gambar 8 Halaman Tutorial 5

Halaman tutorial 5 merupakan halaman akhir dari penjelasan tutorial. Halaman ini memiliki 2 tombol, tombol mulai dan tombol untuk kembali ke tutorial sebelumnya.

Tabel 9 Pengujian Halaman Menu Tutorial 5

No	Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
1	Mengklik / memilih tombol mulai	Aplikasi akan menampilkan halaman utama	Aplikasi menampilkan halaman utama	[√] diterima [] ditolak
2	Mengklik / memilih tombol panah ke kiri	Aplikasi akan menampilkan halaman tutorial 4	Aplikasi menampilkan halaman tutorial 4	[√] diterima [] ditolak



Gambar 9 Halaman Utama 1



Gambar 10 Halaman Utama 2

Tabel 10 Pengujian Halaman Menu Utama

No	Data Masukan	Hasil yang Diharapkan	Pengamatan	Keterangan
1	Input Suara	Aplikasi akan merekam dan menampilkan text hasil input suara berupa kata atau kalimat beserta kategori nilai dan angka nilainya	Aplikasi menampilkan text hasil input suara berupa kata atau kalimat beserta kategori nilai dan angka nilainya	[√] diterima [] ditolak
2	Mengklik / memilih tombol tanda tanya	Aplikasi akan menampilkan halaman tutorial 1	Aplikasi menampilkan halaman tutorial 1	[√] diterima [] ditolak

Halaman Utama merupakan hasil dari input suara user yang sudah direkam. Pada halaman ini terdapat kata atau kalimat yang sudah diucapkan *user* beserta nilai dan kategori nilai yang didapatkan. Halaman utama merupakan halaman test yang akan dilakukan. User dapat kembali ke halaman tutorial atau langsung menginputkan suara untuk melakukan test.

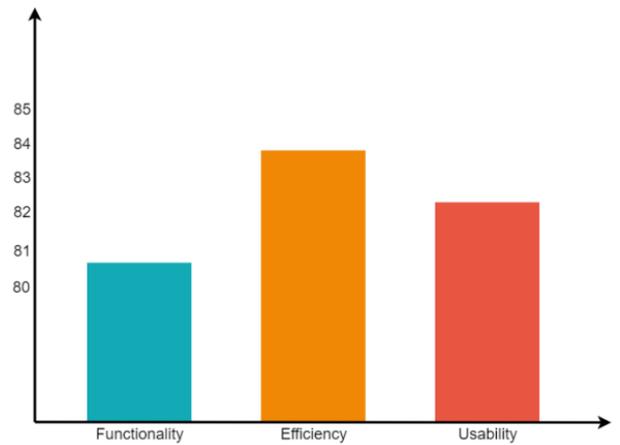
b. Beta Testing

Pengujian beta merupakan pengujian yang dilakukan secara objektif langsung menggunakan kuesioner mengenai tanggapan user terhadap aplikasi yang telah dibangun. Tabel 11 menampilkan data hasil pengujian *beta testing*.

Tabel 11 Data Hasil Uji Beta

Pengguna	Functionality	Efficiency	Usability
1	24	9	20
2	25	8	19
3	24	9	20
4	25	9	21
5	23	9	24
6	21	7	18
7	23	7	19
8	21	9	21
9	24	9	21
10	26	7	22
11	25	9	23
12	24	9	21
13	25	9	23
14	26	7	18
15	24	8	20
16	26	9	22
17	25	6	19
18	24	9	21
19	24	10	19
20	26	9	22
Total skor	485	168	413
Skor maksimum	600	200	500
Persentase	80,83%	84%	82,60%

Perhitungan persentase faktor kualitas aplikasi digambarkan dalam bentuk diagram. Gambar 12 menampilkan persentase faktor kualitas perangkat lunak pada uji *beta*.



Gambar 11 Diagram Persentase Kualitas Aplikasi Pada Uji *Beta*

Hasil kualitas aplikasi terkait aspek *functionality*, *efficiency* dan *usability* diukur dengan menggunakan skala *Likert* seperti ditampilkan pada Tabel 12.

Tabel 12 Interpretasi Persentase *Likert* Untuk Kualitas Perangkat Lunak

No	Persentase	Interpretasi
1	0%-19,99%	Sangat Tidak Layak
2	20%-39,99%	Tidak Layak
3	40%-59,99%	Cukup Layak
4	60%-79,99%	Layak
5	80%-100%	Sangat Layak

Tingkat kelayakan masing-masing faktor kualitas aplikasi setelah membandingkan dengan tabel interpretasi persentase kelayakan aplikasi ditampilkan pada Tabel 13.

Tabel 13 Persentase Kelayakan Tiap Faktor Pada Uji Coba *Beta*

No	Faktor	Persentase	Tingkat Kelayakan
1	Functionality	80,83%	Sangat Layak
2	Efficiency	84%	Sangat Layak
3	Usability	82,60%	Sangat Layak

Perhitungan persentase kualitas perangkat lunak secara keseluruhan dari data hasil pengujian *beta* menggunakan persamaan (1).

$$Persentase\ Kelayakan\ (\%) = \frac{Skor\ yang\ diobservasi}{Skor\ yang\ diharapkan} \times 100\% \quad (1)$$

$$Persentase\ Kelayakan\ (\%) = \frac{1066}{1300} \times 100\%$$

$$Persentase\ Kelayakan\ (\%) = 82\%$$

Hasil pengolahan data uji coba beta menyatakan bahwa aplikasi secara keseluruhan memiliki persentase kelayakan sebesar 82%. Hasil persentase tersebut kemudian dibandingkan dengan tabel Interpretasi Persentase *Likert*. Hasil yang didapatkan setelah membandingkan persentase kelayakan aplikasi adalah dapat disimpulkan bahwa kualitas aplikasi pembelajaran bahasa inggris memiliki kategori “Sangat Layak”.

5. Evaluasi (*Evaluation*)

Aplikasi menyediakan fitur merekam dan memberikan nilai dari hasil input. Proses perekaman hingga penampilan hasil bersifat berulang, sehingga *user* dapat melakukan pembelajaran bahasa Inggris dengan mudah secara terus-menerus dan tidak memerlukan waktu yang lama. Namun aplikasi belum menyediakan fitur riwayat penginputan suara yang sudah dilakukan. Karena sifatnya yang terus berulang, sehingga kata atau kalimat yang sudah terekam sebelumnya tidak tersimpan.

6. KESIMPULAN

Speech recognition dengan menerapkan API IBM Watson telah berhasil dilakukan. Deteksi masukan suara menjadi keluaran berupa nilai pelafalan suara berjalan sesuai rancangan, merekam serta menampilkan hasil dan nilai *confidence* sesuai dengan nilai yang ada pada IBM Watson.

DAFTAR PUSTAKA

- [1] Ardiansyah, Firdan. Pengenalan Dasar Android Programming. Biraynara. Depok. 2011.
- [2] Oates, B, J. *Design And Creation* Pada Buku *Researching Information Systems and Computing*. pp 108-109. 2005.
- [3] Bayu, Handi Dwi Rachma dan Miftahul Huda. Proyek Akhir : Text PreProcessing pada Text to Speech Synthesis System untuk Penutur Berbahasa Indonesia. Institut Teknologi Sepuluh November. Surabaya. 2015.
- [4] Buana, Yudhistira Cahya. dan Fendi Aji P. Skripsi: Augmented Reality for Anatomy Study with Speech Recognition. Universitas Sebelas Maret. Surakarta. 2015.
- [5] Churiyah, Yayah. Jurnal : Komunikasi Lisan dan Tertulis. Universitas Pendidikan Indonesia. Bandung. 2015.
- [6] Irawan. Membuat Aplikasi Android untuk Orang Awam. Maxikom. Palembang. 2012. Jelinek, Frederick. *Statistical Methods for Speech Recognition*. Massachusetts Institute of Technology. United States. 1997. Junapa,
- [7] Komala, Lukiati. Ilmu Komunikasi: Perspektif, Proses, dan Konteks. Widya Padjadjaran. Bandung. 2009.
- [8] Kristanto, Andri. Perancangan Sistem Informasi dan Aplikasinya. Gava Media. Jakarta. 2003.
- [9] Khasanah, Amalia. Naskah Publikasi : Perancangan Aplikasi Al Qur'an Menggunakan Voice Recognition sebagai Media Pencarian Terjemahan Al Qur'an Berbasis 99 Android. Sekolah Tinggi Manajemen Ilmu Komputer Amikom. Yogyakarta. 2013.
- [10] Pressman, Roger. S, *Software Engineering: A Praticioner's Approach*, Fifth 2009.
- [11] Vaishnavi, V., & Keuchler, W. *Design Researsch in Information Systems*. Associaton for Information Systems. (January) 2004.
- [12] Sukamerta, Made. Implementasi Kebijakan Bahasa Inggris pada SD Di Kota Denpasar. *Tesis*, Universitas Udayana. Bali. 2011.

BIODATA PENULIS

Acep Abdul Basit

Mahasiswa program studi Informatika Fakultas Teknik Universitas Siliwangi. Bidang minat pada multimedia dan informatika.

R. Reza El Akbar

Dosen program studi Informatika Fakultas Teknik Universitas Siliwangi.