

# **Aplikasi Penunjang Kemampuan Bahasa Inggris Menggunakan Microsoft *Speech* API**

## **TUGAS AKHIR**

Oleh :

**Syahril Idwar            3310901031**

**Arif Nu'man            3310901057**

**Andrian Muslim        3310901058**

Disusun untuk memenuhi syarat kelulusan matakuliah Tugas Akhir



**PROGRAM STUDI TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BATAM**

**BATAM**

**2011**

## **ABSTRAK**

### **Aplikasi Penunjang Kemampuan Bahasa Inggris Menggunakan Microsoft *Speech API***

Penguasaan bahasa Inggris sebagai bahasa asing yang dominan dipakai dalam pergaulan internasional merupakan salah satu akses untuk meraih kesuksesan di segala bidang. Salah satu tujuan kurikulum nasional pendidikan bahasa Inggris adalah mengembangkan kemampuan berkomunikasi yang meliputi menyimak (listening), berbicara (speaking), membaca (reading), dan menulis (writing) dalam bahasa tersebut, dalam bentuk lisan dan tulisan. Teknologi Pengenalan Suara Otomatis (Automatic Speech Recognition) pada *Microsoft Speech API* dapat digunakan untuk membuat aplikasi penunjang kemampuan bahasa Inggris dalam mempraktikkan keempat keterampilan bahasa tersebut. Aplikasi dibangun berbasis GUI (Graphical User Interface) agar menarik, ramah pengguna (user friendly) dan dapat digunakan secara interaktif untuk melatih empat dasar keterampilan berbahasa Inggris.

Kata Kunci: Pembelajaran Bahasa Inggris, Speech Recognition, Speech Synthesis.

## **ABSTRACT**

### **Aplikasi Penunjang Kemampuan Bahasa Inggris Menggunakan Microsoft *Speech API***

Mastery of English as the dominant foreign language used in the international arena is one of an access to success in all areas. One of the goals of English education national curriculum is to develop communication skills include listening, speaking, reading, and writing in that language, in oral and written form. The introduction of Automatic Speech Recognition on the Microsoft Speech API can be used to make the application supporting English language skills in practicing the four language skills. Application are built based GUI (Graphical User Intefarce) so attractive, user friendly (user friendly) and can be used interactively to practice the four basic skills in English.

Key words: English Education, Speech Recognition, Speech Synthesis.

## DAFTAR ISI

Bab I	Pendahuluan.....	1
I.1	Latar Belakang.....	1
I.2	Rumusan Masalah.....	2
I.3	Batasan Masalah .....	2
I.4	Tujuan.....	3
I.5	Sistematika Penulisan .....	3
Bab II	Tinjauan Pustaka.....	4
II.1	Perspektif Keterampilan Berbahasa.....	4
II.2	Bahasa Inggris.....	5
II.3	Microsoft <i>Speech</i> API.....	7
II.4	<i>Text to Speech</i> .....	8
II.4.1	Definisi <i>Text to Speech</i> .....	8
II.5	<i>Speech</i> Recognition .....	8
II.5.1	Definisi <i>Speech</i> Recognition.....	8
II.6	Algoritma FFT (Fast Fourier Transform) .....	9
II.7	Konsep Digitalisasi Audio.....	9
II.7.1	Suara .....	9
II.7.2	Analog to Digital .....	10
II.8	Pemrograman C# .....	12
II.9	Proses Utama Sistem .....	13
II.9.1	Proses <i>Speech Recognition</i> (Pengenalan Suara) .....	13
II.9.2	Proses <i>Text to Speech</i> .....	14
Bab III	Analisis .....	15
III.1	Deskripsi Umum Sistem.....	17
III.2	Karakteristik Pengguna.....	18
III.3	Kebutuhan Fungsional.....	19
III.4	Kebutuhan Non Fungsional .....	19
III.5	Use Case .....	20

III.5.1	Diagram Use Case .....	20
III.5.2	Skenario Use Case .....	20
III.5.3	Use Case Pilih Kategori Soal.....	20
III.5.4	Use Case Jawab Soal Speak .....	21
III.5.5	Use Case Jawab Soal Listen .....	21
III.6	Analisis Kelas .....	22
Bab IV	Perancangan.....	24
IV.1	Sequence Diagram .....	24
IV.1.1	Sequence Diagram Pilih Kategori Soal .....	24
IV.1.2	Sequence Diagram JawabSoalSpeak .....	25
IV.1.3	Sequence Diagram JawabSoalListen .....	26
IV.2	Diagram Kelas .....	26
IV.3	Rancangan Kelas Rinci.....	28
IV.3.1	Kelas PilihKategoriTopik .....	28
IV.3.2	Kelas JawabSoalListen .....	29
IV.3.3	Kelas JawabSoalSpeak .....	30
IV.4	Perancangan Antarmuka.....	32
IV.4.1	Inteface welcome .....	32
IV.4.2	Interface Pilih Kategori dan Topik Soal .....	33
IV.4.3	Interface Jawab Soal Listen .....	34
IV.4.4	Interface Jawab Soal Speak .....	35
IV.4.5	Interface wacana .....	36
Bab V	Impementasi dan Pengujian.....	37
V.1	Implementasi Kelas .....	37
V.2	Pengujian Menampilkan Soal Sesuai Kategori dan Topik .....	38
V.2.1	Skenario Pengujian .....	38
V.2.2	Hasil Pengujian.....	38
V.3	Pengujian Pencocokan Suara dengan Teks.....	38
V.3.1	Skenario Pengujian .....	38
V.3.2	Hasil Pengujian.....	38

V.4	Pengujian Pencocokan Teks dengan Suara.....	41
V.4.1	Skenario Pengujian .....	41
V.4.2	Hasil Pengujian.....	41
Bab VI	Kesimpulan dan Saran .....	45
VI.1	Kesimpulan .....	45
VI.2	Saran .....	45

## DAFTAR GAMBAR

Gambar 1 Diagram Kern .....	4
Gambar 2 Modifikasi Diagram Kern .....	5
Gambar 3 Karakteristik Gelombang Suara .....	10
Gambar 4 Proses Digital Audio .....	11
Gambar 5 Proses Analog to Digital Conversion (ADC) .....	12
Gambar 6 Spektrum Suara .....	13
Gambar 7 Proses text to <i>Speech</i> .....	14
Gambar 8 Deskripsi Umum Sistem.....	17
Gambar 9 Diagram Use Case .....	20
Gambar 10 Analisis Kelas.....	22
Gambar 11 Sequence diagram Pilih Kategori Soal .....	24
Gambar 12 Sequence diagram JawabSoalSpeak .....	25
Gambar 13 Sequence diagram menjawab soal .....	26
Gambar 14 Diagram Interaksi Kelas .....	27
Gambar 15 interface welcome.....	32
Gambar 16 Tampilan Menu Kategori Soal .....	33
Gambar 17 Tampilan Jawab Soal Listen.....	34
Gambar 18 Tampilan Jawab Soal Speak.....	35
Gambar 19 interface wacana .....	36

## DAFTAR TABEL

Tabel 1 Spesifikasi Analisis Kelas .....	23
Tabel 2 keterangan Kelas Fix .....	27
Tabel 3 Deskripsi Tampilan interface welcome .....	33
Tabel 4 Deskripsi Tampilan Pilih Kategori Soal .....	33
Tabel 5 Deskripsi Tampilan Jawab Soal Listening .....	35
Tabel 6 Diskripsi tampilan interface jawab soal speak .....	36
Tabel 7 Deskripsi Tampilan interface wacana .....	36
Tabel 8 Implementasi Kelas .....	37
Tabel 9 Pengamatan I .....	38
Tabel 10 Pengamatan II .....	40
Tabel 11 Pengamatan III .....	43

## **Bab I Pendahuluan**

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan dan sistematika penulisan dalam tugas akhir.

### **I.1 Latar Belakang**

Bahasa memiliki peran sentral dalam perkembangan intelektual, sosial, dan emosional siswa dan merupakan kunci penentu menuju keberhasilan dalam mempelajari semua bidang studi. Penguasaan bahasa asing, terutama yang dominan dalam pergaulan internasional, merupakan salah satu akses untuk meraih keberhasilan dalam berbagai bidang. Kemampuan berkomunikasi meliputi mendengar atau menyimak (listening), berbicara (speaking), membaca (reading), dan menulis (writing). Kenyataan di lapangan menunjukkan bahwa sebagian besar tamatan sekolah menengah tidak dapat berkomunikasi dalam bahasa Inggris walaupun mereka telah belajar bahasa Inggris selama 6 (enam) tahun di SLTP dan SMU. Hal tersebut terjadi karena pengajaran bahasa Inggris di sekolah lebih banyak terfokus pada pengajaran tata-bahasa dan kurang memberikan kesempatan pada siswa untuk berlatih berbicara dan mendengarkan dalam bahasa Inggris. Pada umumnya siswa hanya 10% mengingat dari apa yang mereka baca, 20% dari apa yang mereka dengar, 30% dari apa yang mereka lihat, 50% dari apa yang mereka dengar dan lihat, 70% dari apa yang mereka katakan dan tulis, dan 90% dari apa yang mereka katakan seperti yang mereka lakukan. Untuk itu diperlukan media pendamping untuk melatih semua keterampilan komunikasi tersebut sehingga dapat memudahkan guru dalam memberikan pengajaran dan memudahkan siswa dalam menangkap isi pembelajaran sehingga kualitas pembelajaran menjadi lebih baik. Tujuan tersebut dapat dipenuhi dengan bantuan teknologi pengenalan suara otomatis (automatic *Speech* recognizer). Menurut Supriyono(2004), teknologi tersebut memungkinkan komputer untuk mengenali suara manusia sesuai dengan

bahasa yang digunakan sehingga komunikasi antara manusia dan komputer dapat dilakukan se-natural mungkin seperti komunikasi antara manusia dan manusia. Teknologi pengenalan suara otomatis tersebut belum dapat digunakan secara langsung dalam aplikasi, karena masih membutuhkan implementasi pada sistem operasi. Oleh Microsoft teknologi pengenalan suara otomatis digabungkan dengan teknologi sintesis suara (*Speech synthesis*), yaitu teknologi yang dapat mensintesis suara manusia menjadi suatu API (Application Programming Interface) yang disebut Microsoft *Speech API*. Untuk memudahkan penggunaannya maka aplikasi dibuat menggunakan GUI (Graphical User Interface). Dengan GUI semua pesan dari komputer ke pengguna dilakukan dengan menggunakan visualisasi gambar atau tulisan yang mudah dimengerti oleh manusia sebagai pengguna. Agar dapat digunakan secara berulang-ulang sebagai latihan maka berkas yang digunakan dapat disimpan dalam sebuah basis-data.

## **I.2 Rumusan Masalah**

Adapun rumusan masalah dari tugas akhir ini adalah dikarenakan suatu keadaan bahwa sebagian besar tamatan sekolah menengah tidak dapat berkomunikasi dalam bahasa Inggris walaupun mereka telah belajar bahasa Inggris selama 6 (enam) tahun di SLTP dan SMU, sehingga dengan memanfaatkan komponen yang terdapat pada *Microsoft Speech Application Programming Interface* memungkinkan untuk membuat suatu aplikasi pencocokan suara dengan teks (*Speech to text*) dan pencocokan teks dengan suara (*text to Speech*) dimana kemampuan mencocokkan ini dapat kita manfaatkan sebagai media untuk belajar serta berlatih mendengar, menulis dan mengucapkan bahasa inggris.

## **I.3 Batasan Masalah**

Adapun batasan masalah dalam tugas akhir ini ialah:

1. Pembelajaran bahasa inggris hanya mencakup *listening*, *writing* dan *speaking*.

2. Sistem tidak efektif menerima masukan suara jika kondisi lingkungan dalam keadaan *noise* yang tinggi.
3. Sistem tidak menangani pengukuran *noise*.
4. Pembuatan aplikasi ini tidak membahas secara mendalam tentang Microsoft *Speech* API, melainkan pemanfaatan komponen yang telah ada untuk menciptakan aplikasi belajar ini.

#### **I.4 Tujuan(bentuk point)**

Membuat suatu aplikasi yang dapat digunakan untuk belajar serta berlatih mendengar, menulis dan mengucapkan bahasa inggris dengan memanfaatkan komponen yang terdapat pada *Microsoft Speech API*.

#### **I.5 Sistematika Penulisan**

**Bab 1 Pendahuluan** yang berisi Latar Belakang, Rumusan Masalah, Batasan Masalah, Tujuan dan Sistematika Penulisan.

**Bab 2 Tinjauan Pustaka** Studi pustaka yang yang menjadi alasan tugas akhir ini dan pustaka yang menunjang pembuatan sistem.

**Bab 3 Analisis** yang berisi spesifikasi sistem, Use Case.

**Bab 4 Perancangan** yang berisi Sequence Diagram, Class Diagram dan dokumen perancangan lainnya.

**Bab 5 Implementasi dan Pengujian** yang berisi hasil implementasi dan pembahasannya serta hasil pengujian serta pembahasannya.

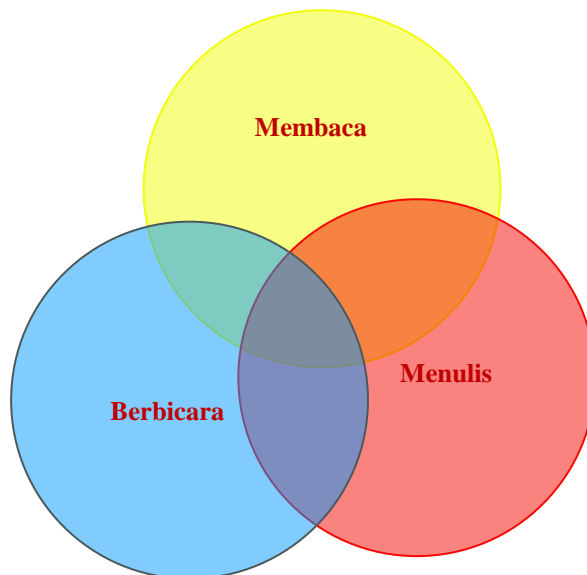
**Bab 6 Kesimpulan dan Saran** yang berisi tentang Kesimpulan dari hasil pembangunan aplikasi dan Saran untuk perbaikan dan pengembangan.

## Bab II Tinjauan Pustaka

Pada bab ini akan di jelaskan teori teori dasar pendukung dalam tugas akhir ini seperti: Perspektif keterampilan berbahasa, bahas inggris, *text to Speech*, *Speech recognition*, algoritma yang digunakan oleh *library*, bahasa pemograman yang digunakan dan beberapa dasar teori pendukung lainnya.

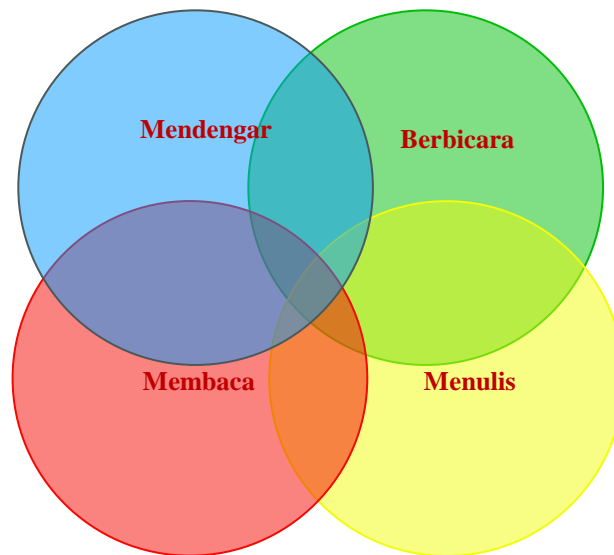
### II.1 Perspektif Keterampilan Berbahasa

Salah satu lingkup keterampilan berbahasa, yaitu mendengarkan, berbicara, membaca, dan menulis. Keterampilan mendengarkan, berbicara, membaca dan menulis tidak diposisikan secara linier, berjajar, melainkan terpadu sebagai struktur yang mewarnai rancangan proses belajar dan mengajar. Hal itu dapat dipresentasikan dalam tiga lingkaran berbicara (talking), membaca (reading), dan menulis (writing) seperti pada Gambar 1.



Gambar 1 Diagram Kern

Dalam konteks pengajaran bahasa asing, seringkali kegiatan mendengarkan dilakukan terpisah dari kegiatan berbicara karena kegiatan tersebut difokuskan kepada latihan mengucapkan atau menirukan bunyi-bunyi, kata dan sebagainya. Oleh karenanya model Gambar 2 di atas dapat dimodifikasi menjadi empat lingkaran seperti pada Gambar 2



**Gambar 2 Modifikasi Diagram Kern**

## **II.2 Bahasa Inggris**

Bahasa Inggris adalah sebuah bahasa yang berasal dari Inggris, merupakan bahasa utama di Britania Raya (termasuk Inggris), Amerika Serikat, serta banyak negara lainnya, dan termasuk rumpun bahasa Jermanik Barat. Bahasa ini berawal dari kombinasi antara beberapa bahasa lokal yang dipakai oleh orang-orang Norwegia, Denmark, dan Anglo-Saxon dari abad ke-6 sampai 10. Lalu pada tahun 1066 dengan ditaklukkan Inggris oleh William the Conqueror, sang penakluk dari Normandia, Perancis Utara, maka bahasa Inggris dengan sangat intensif mulai dipengaruhi bahasa Latin dan bahasa Perancis. Dari seluruh kosakata bahasa Inggris modern, diperkirakan  $\pm 50\%$  berasal dari bahasa Perancis dan Latin.

Bahasa merupakan bahasa pertama di banyak negara-negara maju seperti Amerika Serikat, Australia dan Selandia Baru. Selain itu bahasa Inggris juga merupakan salah satu bahasa resmi di organisasi internasional seperti Perserikatan Bangsa-

Bangsa dan Komite Olimpiade Internasional. Di dunia, bahasa Inggris merupakan bahasa kedua pertama yang dipelajari. Bahasa Inggris bisa menyebar karena pengaruh politik dan imperialisme Inggris dan selanjutnya Britania Raya di dunia. Salah satu pepatah Inggris zaman dahulu mengenai kerajaan Inggris yang disebut Imperium Britania (British Empire) adalah tempat “Matahari yang tidak pernah terbenam” (“where the sun never sets”).

Sangat penting bagi kita warga negara Indonesia untuk bisa berbahasa Inggris. Pertama karena Indonesia dikelilingi oleh negara-negara yang kebanyakan penduduknya menggunakan bahasa Inggris sebagai bahasa pertama atau kedua. Negara-negara tersebut adalah Singapura, Malaysia, Filipina dan Papua Nugini. Hal ini juga terjadi di negara Belanda. Di sana, murid-murid pada tingkat SMA memang dianjurkan mempelajari dan menguasai bahasa asing mengingat bahasa Belanda tidak dipakai oleh negara di sekelilingnya. Jerman memakai bahasanya sendiri. Belgia memakai Perancis.

Alasan kedua dan paling umum, bahasa Inggris perlu dipelajari karena penggunaan luasnya sebagai bahasa komunikasi Internasional. Agar dapat melakukan komunikasi dengan orang-orang yang berbeda latar belakang budaya dan kenegaraan, bahasa Inggris menjadi pilihan utama yang sering dipakai dalam melakukan komunikasi. Contoh yang mudah dilihat ada di dunia pariwisata. Para wisatawan yang melakukan perjalanan di negara asing lazim menggunakan bahasa Inggris untuk dapat berkomunikasi dengan warga negara asli yang dikunjunginya. Bukan hanya penutur jati bahasa Inggris, wisatawan yang tidak menggunakan bahasa Inggris sebagai bahasa ibu juga memilih bahasa Inggris sebagai *lingua franca*-nya. Orang Jepang yang melancong ke Indonesia, menggunakan bahasa Inggris apabila dia hendak menanyakan sesuatu pada orang pertama yang ditemuinya di jalan. Wisatawan Indonesia yang berjalan-jalan di Paris akan sangat senang sekali apabila bertemu dengan penduduk setempat yang menguasai bahasa Inggris untuk dimintai bantuannya. Juga kecil kemungkinannya ada orang Italia

yang berani berwisata ke India tanpa memiliki bekal bahasa Inggris yang memadai. Bahasa Inggris juga menjadi bahasa pengantar resmi dalam dunia transportasi udara dan laut. Pilot pesawat, apapun kewarganegaraanya, dilatih untuk menguasai bahasa Inggris agar dapat berkomunikasi dengan pihak menara pengawas bandara yang menjadi tujuan pesawat yang diterbangkannya. Apakah dia menerbangkan pesawat di Asia atau di Afrika, dia harus berkomunikasi dalam bahasa Inggris. Begitu pula pihak menara pengawas bandara pun harus mahir berbicara dalam bahasa Inggris, karena pesawat yang mendarat di bandara tidak hanya datang dari satu negara tapi juga manca negara. Tidak bisa dibayangkan rupanya apabila para pilot dan petugas menara pengendali harus menguasai seluruh bahasa di dunia ini.<sup>1</sup>

### **II.3 Microsoft *Speech* API**

*Speech Application Programming Interface (Speech API)* adalah sebuah API yang dikembangkan oleh Microsoft yang digunakan sebagai pengenalan suara didalam lingkungan pemrograman aplikasi Windows. Sampai saat ini *Speech API* dikemas baik berupa SDK (*System Development Kit*) maupun disertakan dalam sistem operasi Windows itu sendiri. Salah satu aplikasi yang telah menggunakan *SPEECH API* antara lain *Microsoft Office*.

Komponen utama di dalam *Speech API* yang digunakan dalam pembuatan tugas akhir ini adalah sebagai berikut:

1. *Voice Command*, sebuah obyek level tinggi untuk perintah dan kontrol menggunakan pengenalan suara.
2. *Voice Dictation*, sebuah obyek level tinggi untuk continous dictation *Speech* recognition.
3. *Voice Talk*, sebuah obyek level tinggi untuk *Speech synthesis*.

---

<sup>1</sup> Andreanna Achmad *WHAT TO SAY*, Surabaya , *Arkola*,1996,h.3

4. *Direct Speech Recognition*, sebuah obyek sebagai mesin untuk mengontrol pengenalan suara (direct control of recognition engine).
5. *Direct Text to Speech*, sebuah obyek sebagai mesin yang mengontrol *synthesis*.
6. *Audio Object*, untuk membaca dari audio device atau sebuah file audio.

## II.4 Text to Speech

### II.4.1 Definisi Text to Speech

*Text-To-Speech System* adalah sebuah konverter tulisan menjadi sebuah ucapan atau audio (*spoken language*) yang bisa di dengar oleh *user*. Konverter ini banyak digunakan untuk membantu mempermudah manusia yang memiliki kekurangan seperti tuna wicara atau tunanetra, bahkan konverter ini juga bisa digunakan untuk mempermudah memberi pengajaran kepada anak-anak atau orang-orang yang masih belajar tulis menulis dan membaca. Namun dalam perkembangannya, konverter ini juga banyak diintegrasikan pada aplikasi-aplikasi lain. Seperti pada handphone yang memiliki OS symbian. Konverter ini juga digunakan sebagai *message reader*. Sistem *text to Speech* meliputi : proses *text pre-processing*, pembangkitan *prosody* (irama, penekanan dan intonasi) dan proses *diphone concatenation* atau unit pemroses *diphone*.

## II.5 Speech Recognition

### II.5.1 Definisi Speech Recognition

*Speech recognition* adalah proses identifikasi suara berdasarkan kata yang diucapkan. Parameter yang dibandingkan ialah tingkat penekanan suara yang kemudian akan dicocokkan dengan template database yang tersedia. Algoritma yang digunakan oleh *Microsoft Speech API* dalam proses *Speech recognition* ini adalah algoritma FFT (*Fast fourier transform*), yaitu algoritma yang cukup efisien dalam pemrosesan sinyal digital (dalam hal ini suara) dalam bentuk diskrit. Konsep utama algoritma ini adalah mengubah sinyal suara yang berbasis waktu

menjadi berbasis frekuensi dengan membagi masalah menjadi beberapa masalah yang lebih kecil. Kemudian, setiap masalah diselesaikan dengan cara melakukan pencocokan pola digital suara.<sup>2</sup>

## **II.6 Algoritma FFT (Fast Fourier Transform)**

Proses penting dalam Digital Signal Processing (DSP) adalah menganalisis suatu sinyal input maupun output untuk mengetahui karakteristik tertentu. Proses analisis dan sintesis dalam domain waktu memerlukan analisis cukup panjang dengan melibatkan turunan dari fungsi, yang dapat menimbulkan ketidakteelitian hasil analisis. Analisis dan sintesis sinyal akan lebih mudah dilakukan pada domain frekuensi, karena besaran yang paling menentukan suatu sinyal adalah frekuensi. Oleh karena itu, untuk dapat bekerja pada domain frekuensi dibutuhkan suatu formulasi yang tepat sehingga proses manipulasi sinyal sesuai dengan kenyataan. Salah satu formulasi yang ampuh untuk proses pengolahan sinyal adalah menggunakan Discrete Fourier Transform (DFT). Prinsip DFT adalah mentransformasikan (alih bentuk) sinyal yang semula analog menjadi diskret dalam domain waktu, dan kemudian diubah ke dalam domain frekuensi. Hal ini dilakukan dengan mengalikan sinyal diskret dengan suatu fungsi kernel. Algoritma lain yang lebih cepat adalah Fast Fourier Transform (FFT). Prinsip kerja FFT adalah membagi sinyal hasil penyamplingan menjadi beberapa bagian yang kemudian masing-masing bagian diselesaikan dengan algoritma yang sama dan hasilnya dikumpulkan kembali.

## **II.7 Konsep Digitalisasi Audio**

### **II.7.1 Suara**

Suara atau bunyi biasanya merambat melalui udara dan akan memantul apabila ada dinding, suara atau bunyi tidak dapat merambat melalui ruang hampa. Suara

---

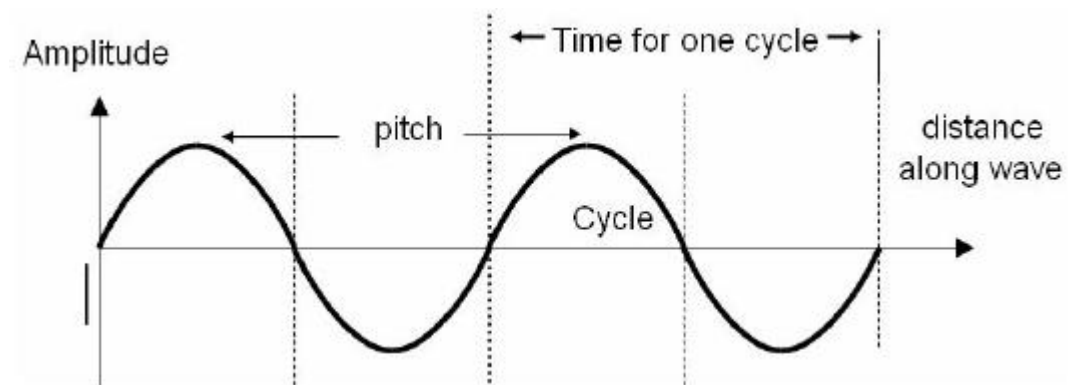
<sup>2</sup> [http://en.wikipedia.org/wiki/Speaker\\_recognition](http://en.wikipedia.org/wiki/Speaker_recognition). Diakses pada tanggal 10 September 2011 pukul 20.10

dihasilkan oleh getaran suatu benda. Selama bergetar, perbedaan tekanan terjadi di udara sekitarnya. Pola osilasi yang terjadi disebut sebagai gelombang, gelombang mempunyai pola sama dan berulang pada interval tertentu, yang disebut periode.

Suara dapat dibagi menjadi dua yaitu :

- Frekuensi adalah merupakan standar pengukuran suara jumlah satu siklus per detik (Jumlah pitch atau amplitudo terbesar yang terdengar oleh telinga).
- Amplitudo merupakan intensitas atau keras lemahnya suara, Semakin keras suara semakin besar amplitudonya.

Dimana semua bunyi mempunyai jangka waktu dan irama



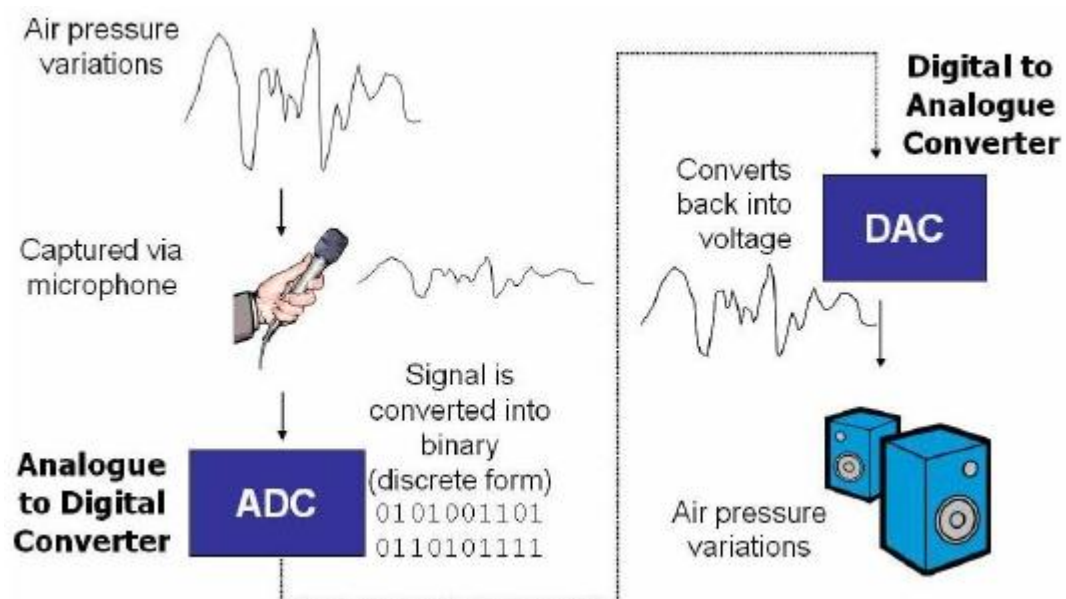
**Gambar 3 Karakteristik Gelombang Suara**

### **II.7.2 Analog to Digital**

Saat ini perkembangan audio digital sudah pesat, walaupun file audio digital banyak jenisnya tetapi teori dasar tentang proses bekerjanya adalah sama. Dimana file diproses dengan pengklasifikasian sinyal analog (dalam bentuk voltase) lalu disandakan, diproses, disimpan, dan akhirnya di produksi berulang-ulang kali kedalam suatu sistim bilangan biner. Pada perangkat digital sistem bilangan yang digunakan sistem bilangan binary (basis 2) dalam bekerja, dengan sistem bilangan ini perangkat tersebut dapat bekerja secara cepat dan lebih efisien. Sedangkan pada dua puluh enam (26) bilangan alphabet yang kita kenal, menggunakan sistem

bilangan desimal (basis 10). Oleh karena itu bilangan alphabet harus di ubah dahulu (diterjemahkan) kedalam sistem bilangan yang dapat dimengerti oleh perangkat digital, yakni bilangan biner dan data tersebut dapat di *encoded* kan pada beberapa bentuk media seperti:

- Logika 1 atau 0
- On atau Off
- Voltage atau no voltage



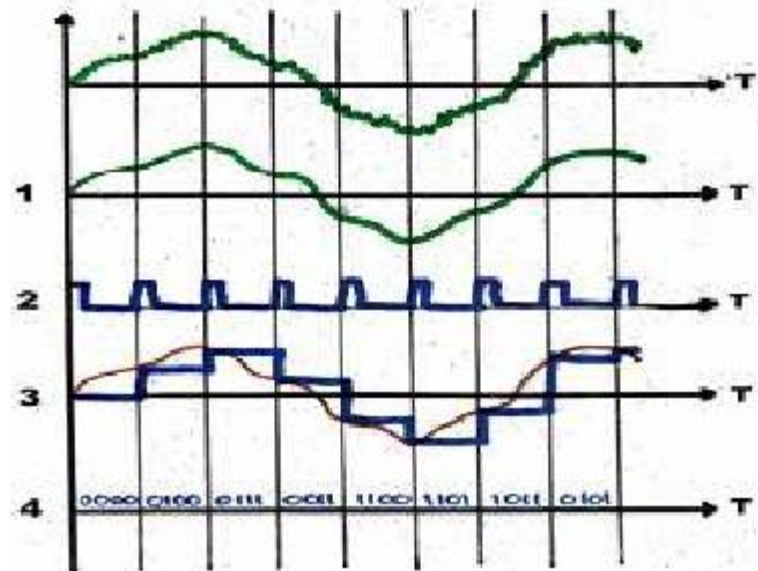
Gambar 4 Proses Digital Audio

Analog to digital conversion (ADC) adalah proses mengubah amplitude gelombang bunyi ke dalam waktu interval tertentu (disebut juga sampling), sehingga menghasilkan representasi digital dari suara, dimana sampling rate adalah beberapa gelombang yang diambil dalam satu detik (Standar sampling rate untuk berbicara 11,025 kHz)

Berikut proses Analog to digital conversion (ADC) :

1. Membuang frekuensi tinggi dari source signal.
2. Mengambil sample dengan interval waktu tertentu (sampling).

3. Menyimpan amplitude sample dan mengubahnya ke dalam bentuk diskrit (kuantisasi).
4. Merubah bentuk menjadi nilai biner.



Gambar 5 Proses Analog to Digital Conversion (ADC)

## II.8 Pemrograman C#

C# (dibaca: C sharp) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka NET Framework.

Standar European Computer Manufacturer Association (ECMA) mendaftarkan beberapa tujuan desain dari bahasa pemrograman C#, sebagai berikut:

- Bahasa pemrograman C# dibuat sebagai bahasa pemrograman yang bersifat general-purpose (untuk tujuan jamak), berorientasi objek, modern, dan sederhana.
- Bahasa pemrograman C# ditujukan untuk digunakan dalam mengembangkan komponen perangkat lunak yang mampu mengambil keuntungan dari lingkungan terdistribusi.

- Portabilitas programmer sangatlah penting, khususnya bagi programmer yang telah lama menggunakan bahasa pemrograman C dan C++.

Dukungan untuk internasionalisasi (multi-language) juga sangat penting. C# ditujukan agar cocok digunakan untuk menulis program aplikasi baik

## II.9 Proses Utama Sistem

### II.9.1 Proses *Speech Recognition* (Pengenalan Suara)

Semua metode dasar proses pengenalan suara terdiri dari dua fase operasi, pertama adalah proses training, pada proses ini sistem belajar dari referensi pola yang berupa perbedaan pola sinyal suara misal frase, kata, fonem yang akan mengisi vocabulari dari sistem. Setiap referensi di pelajari dari kata yang dikatakan yang kemudian disimpan dalam template dan telah mengalami metode untuk meratakan dan karakteristik statistik dan parameter statistik. Yang kedua adalah proses recognition yaitu pada proses ini sistem akan diberikan inputan yang belum diketahui dan akan diidentifikasi berdasarkan pola template yang telah didapatkan pada proses training.



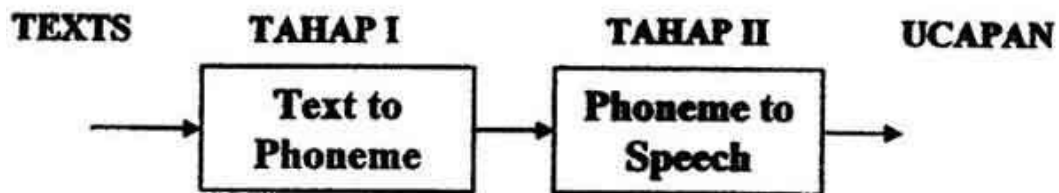
**Gambar 6 Spektrum Suara**

Secara umum, *Speech* recognizer memproses sinyal suara yang masuk dan menyimpannya dalam bentuk digital. Hasil proses digitalisasi tersebut kemudian dikonversi dalam bentuk spektrum (serangkaian gelombang-gelombang) suara yang akan dianalisa dengan membandingkannya dengan template suara pada database sistem. Sebelumnya, data suara masukan dipilah-pilah dan diproses satu per satu berdasarkan urutannya.

## II.9.2 Proses Text to Speech

Sistem Text to *Speech* pada prinsipnya terdiri dari dua sub sistem, yaitu :

- 1) bagian Konverter Teks ke Fonem (*Text to Phoneme*).
- 2) bagian Konverter Fonem to Ucapan (*Phoneme to Speech*).



Gambar 7 Proses text to *Speech*

Bagian konverter teks ke fonem berfungsi untuk mengubah kalimat masukan dalam suatu bahasa tertentu yang berbentuk teks menjadi rangkaian kode-kode bunyi yang biasanya direpresentasikan dengan kode fonem (satuan bunyi terkecil yang dapat membedakan makna), durasi serta *pitch*-nya. Untuk suatu bahasa baru, bagian ini harus dikembangkan secara lengkap khusus untuk bahasa tersebut. Bagian konverter fonem ke ucapan akan menerima masukan berupa kode-kode fonem serta *pitch* dan durasi yang dihasilkan oleh bagian sebelumnya. Berdasarkan kode-kode tersebut, bagian konverter fonem ke ucapan akan menghasilkan bunyi atau sinyal ucapan yang sesuai dengan kalimat yang ingin diucapkan.

Ada beberapa alternatif teknik yang dapat digunakan untuk implementasi bagian ini. Dua teknik yang banyak digunakan adalah formant *Synthesizer*, serta *diphone concatenation*. Formant *Synthesizer* bekerja berdasarkan suatu model matematis yang akan melakukan komputasi untuk menghasilkan sinyal ucapan yang diinginkan. *Synthesizer* jenis ini telah lama digunakan pada berbagai aplikasi. Walaupun dapat menghasilkan ucapan dengan tingkat kemudahan interpretasi yang baik, *Synthesizer* ini tidak dapat menghasilkan ucapan dengan tingkat

kealamian yang tinggi. *Synthesizer* yang menggunakan teknik *diphone concatenation* bekerja dengan cara menggabung-gabungkan segmen-segmen bunyi yang telah direkam sebelumnya. Setiap segmen berupa *diphone* (gabungan dua buah fonem). *Synthesizer* jenis ini dapat menghasilkan bunyi ucapan dengan tingkat kealamian (naturalness) yang tinggi. Konversi dari teks ke fonem sangat dipengaruhi oleh aturan-aturan yang berlaku dalam suatu bahasa. Pada prinsipnya proses ini melakukan konversi dari simbol-simbol tekstual menjadi simbol-simbol fonetik yang merepresentasikan unit bunyi terkecil dalam suatu bahasa. Setiap bahasa memiliki aturan cara pembacaan dan cara pengucapan teks yang sangat spesifik. Hal ini menyebabkan implementasi unit konverter teks ke fonem menjadi sangat spesifik terhadap suatu bahasa. Untuk mendapatkan ucapan yang lebih alami, ucapan yang dihasilkan harus memiliki intonasi. Secara kuantisasi, prosodi adalah perubahan nilai pitch (frekuensi dasar) selama pengucapan kalimat dilakukan atau pitch sebagai fungsi waktu. Pada prakteknya, informasi pembentuk prosodi berupa data-data pitch serta durasi pengucapannya untuk setiap fonem yang dibangkitkan. Nilai-nilai yang dihasilkan diperoleh dari suatu model prosodi. Prosodi bersifat sangat spesifik untuk setiap bahasa, sehingga model yang diperlukan untuk membangkitkan data-data prosodi menjadi sangat spesifik juga untuk suatu bahasa.

## **II.10 Microfone**

Mikrofon (bahasa Inggris: *microphone*) adalah suatu jenis transduser yang mengubah energi-energi akustik (gelombang suara) menjadi sinyal listrik. Mikrofon merupakan salah satu alat untuk membantu komunikasi manusia. Mikrofon dipakai pada banyak alat seperti telepon, alat perekam, alat bantu dengar, dan pengudaraan radio serta televisi.

Secara umum ada tiga jenis mikrofon dengan sifat yang berbeda-beda.

**1) Ribbon Microphone/Mike Carbon**

Jenis ini kualitasnya tidak/kurang bagus, karena daya serapnya sangat rendah, sehingga penggunaannya harus memerlukan tambahan kekuatan.

**2) Dynamic Microphone/Mike dinamik**

Mikrofon jenis ini sering juga disebut ball microphone/mike ball. Daya serap jenis mikrofon ini sangat tinggi. Artinya, sangat sensitif karena mampu menyerap seluruh suara.

**3) Condensor Microphone/Mike condensor**

Seperti halnya jenis mike ball, mike condensor juga mampu menerima suara dari segala arah, meskipun masih harus memerlukan tambahan kekuatan. *Microphone* jenis inilah yang di digunakan dalam mengerjakan soal-soal *speaking*.

## Bab III Analisis

Pada bab analisis ini akan dijelaskan mengenai Deskripsi Umum Sistem, Karakteristik Pengguna, Fitur Utama Perangkat Lunak, Kebutuhan Fungsional, Kebutuhan Non Fungsional, Diagram Use Case dan Analisis Kelas.

### III.1 Deskripsi Umum Sistem



Gambar 8 Deskripsi Umum Sistem

Keterangan:

1. Pengguna membaca cara penggunaan aplikasi pada tampilan pertama.
2. Pengguna memilih kategori soal yang telah disediakan pada tampilan utama yaitu listening dan speaking.
3. System akan menampilkan topik-topik berdasarkan kategori yang dipilih pengguna pada menu utama.
4. Pengguna memilih topik yang diinginkan dan mengikuti test.
5. Bila pengguna memilih kategori listening akan muncul wacana yang berisi cerita untuk di pahami dan merupakan dasar pengguna untuk menjawab pertanyaa. Disini Pengguna memilih tombol listen untuk mendengarkan pertanyaan dan pengguna memberikan masukan berupa data teks melalui *keyboard*, lalu bila pengguna yakin dengan jawaban maka pilih ok sedangkan bila belum yakin dengan jawabannya pengguna memilih tombol skip untuk melewati pertanyaan tersebut.

6. Bila pengguna memilih kategori *Speaking* akan muncul wacana yang berisi cerita untuk di pahami dan merupakan dasar pengguna untuk menjawab pertanyaan. Disini pengguna membaca pertanyaan dalam *form* kategori *speaking*. Pengguna memberikan masukan berupa data suara melalui *microphone* untuk menjawab soal. Sebelumnya pengguna memilih tombol *speak* kemudian menjawab pertanyaan dengan menahan tombol tersebut, jawaban akan muncul di list jawaban, lalu bila pengguna yakin dengan jawaban maka pilih *ok* sedangkan bila belum yakin dengan jawabannya pengguna memilih tombol *skip* untuk melewati pertanyaan tersebut.
7. *Microphone* digunakan sebagai perantara masukan suara yang diberikan oleh pengguna untuk proses pengecekan suara dengan soal.
8. Masukan jawaban berupa teks melalui *keyboard* akan langsung dicocokkan dengan jawaban yang ada di *database*.
9. Sistem akan memberikan hasil pengecekan dan menampilkannya dalam bentuk pesan dan jumlah jawaban yang benar.
10. Pengguna dapat menggunakan tombol *story* untuk membaca wacana dengan kondisi waktu tetap berjalan.
11. Waktu berjalan ketika pengguna telah membaca wacana topik dan bila waktu habis maka sistem hanya akan membaca jawaban yang benar, lalu sistem akan kembali ke menu utama.

### **III.2 Karakteristik Pengguna**

Karakteristik pengguna dalam tugas akhir ini adalah semua pengguna yang bertugas untuk menyelesaikan semua soal-soal yang diberikan oleh aplikasi sebelum waktu berakhir.

### **III.3 Kebutuhan Fungsional (seuaikan dengan tujuan)**

Kebutuhan fungsional diawali huruf F diikuti dengan nomor urut dan fungsionalnya. Dibawah ini merupakan penjelasan kebutuhan fungsional:

1. F-001 Sistem mampu menampilkan topik sesuai kategori yang ditentukan pengguna.
2. F-002 Sistem menerima masukan suara atau teks dari pengguna sebagai jawaban atas soal yang diberikan.
3. F-003 Sistem mampu mengenali dan mengecek suara atau teks masukan dengan jawaban yang diharapkan.
4. F-004 Sistem mampu menampilkan hasil penilaian secara langsung dari hasil keseluruhan untuk semua soal tiap kategori dan hanya menampilkan perhitungan benar dan salah saja tidak meliputi penilaian berupa skor.

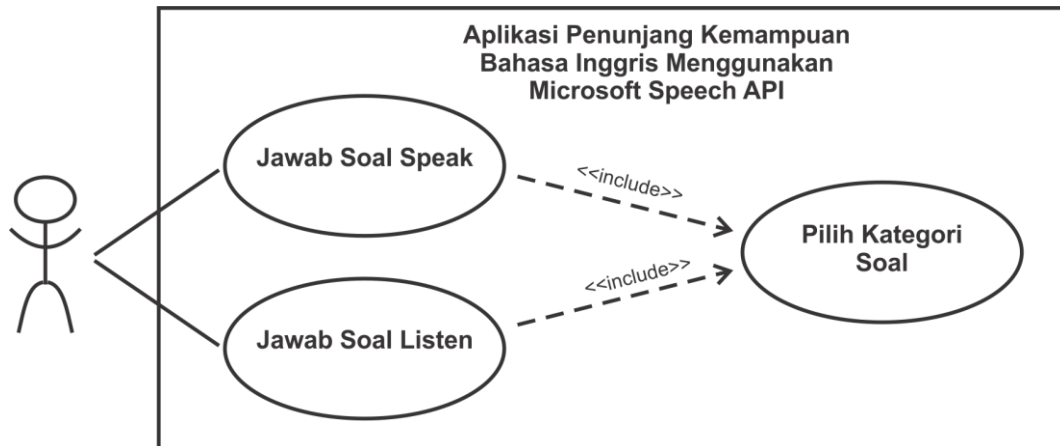
### **III.4 Kebutuhan Non Fungsional**

Kebutuhan fungsional diawali huruf NF (Nilai Fungsional) yang diikuti dengan nomor dan fungsionalnya. Dibawah ini merupakan penjelasan kebutuhan non fungsional:

1. NF-001 Sistem pembelajaran bahasa Inggris menggunakan sistem operasi Windows.
2. NF-002 Sistem menggunakan *microphone* standar untuk bisa menangkap masukan suara.
3. NF-003 Sistem memerlukan kondisi lingkungan yang tenang dan tidak ribut untuk dapat menangkap dan mengenali suara dengan baik.

## III.5 Use Case

### III.5.1 Diagram Use Case



Gambar 9 Diagram Use Case

### III.5.2 Skenario Use Case

Skenario pada Aplikasi Penunjang Kemampuan Bahasa Inggris Menggunakan Microsoft *Speech* API adalah pengguna memilih kategori yang tersedia, kemudian memilih soal yang terlampir dan dilanjutkan dengan menyelesaikan soal yang diberikan dengan melafalkan atau menuliskan jawaban sesuai soal. Setelah itu sistem akan melakukan pengecekan terhadap masukan suara atau teks yang diberikan dan menampilkan hasilnya.

### III.5.3 Use Case Pilih Kategori Soal

Aktor : Pengguna

Kondisi Awal :

- Jendela kategori telah aktif dan menampilkan beberapa kategori soal.
- Pengguna memilih salah satu kategori soal.
- Pengguna memilih salah satu soal yang terlampir.

Kondisi Akhir : Tampil jendela selanjutnya yang berisi wacana soal

dan kemudian jendela soal sesuai kategori pilihan pengguna.

- Skenario :
- Pengguna memilih kategori soal yang tersedia.
  - Pengguna memilih salah satu soal yang terlampir.
  - Tampil jendela selanjutnya yang berisi soal sesuai kategori pilihan pengguna.

#### **III.5.4 Use Case Jawab Soal Speak**

Aktor : Pengguna

- Kondisi Awal :
- *Micophone*, *keyboard* dan sistem pembelajaran bahasa inggris telah aktif.
  - Pengguna telah memilih kategori dan soal yang diinginkan.
  - Pengguna menjawab soal yang diberikan dengan melafalkannya.

Kondisi Akhir : Jawaban yang diberikan pengguna telah masuk kedalam sistem untuk dilakukan pengecekan dan kemudian ditampilkan hasil pengecekan.

- Skenario :
- Pengguna menjawab soal dengan melafalkan jawabannya kearah *microphone* dengan jelas. Setelah masukan suara pengguna diterima sistem maka akan dilakukan pengecekan terhadap *database* jawaban.

#### **III.5.5 Use Case Jawab Soal Listen**

Aktor : Pengguna

- Kondisi Awal :
- *Keyboard* dan sistem pembelajaran bahasa inggris telah aktif.

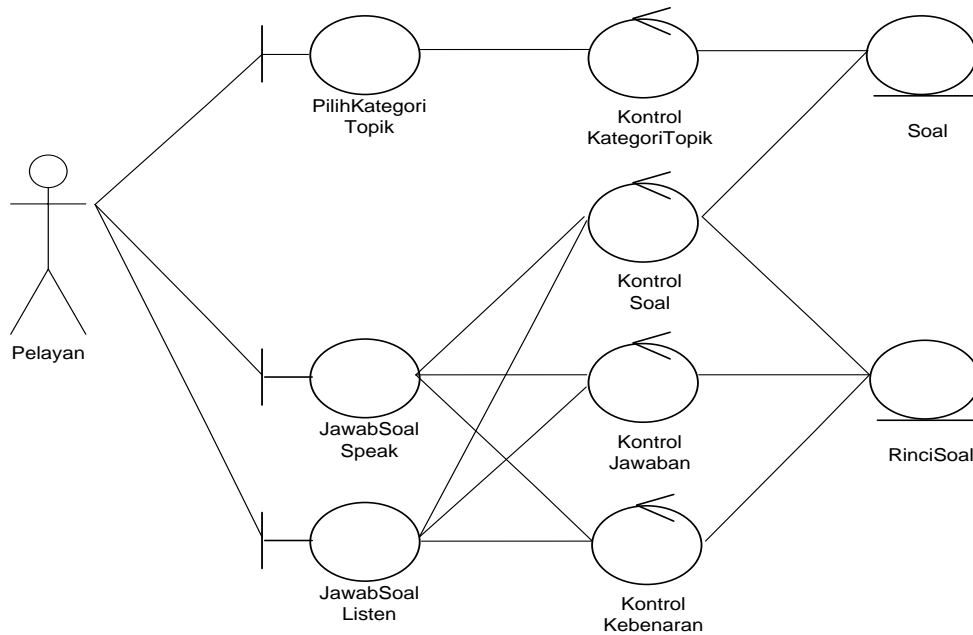
- Pengguna telah memilih kategori yang diinginkan.
- Pengguna memilih soal yang terlampir.
- Pengguna menjawab soal yang diberikan.

Kondisi Akhir : Jawaban yang diberikan pengguna telah masuk kedalam sistem untuk dilakukan pengecekan dan kemudian ditampilkan hasil pengecekan.

Skenario : - Pengguna menjawab soal dengan mengetikkan jawabannya pada *field* yang tersedia. Setelah masukan teks pengguna diterima sistem maka akan dilakukan pengecekan terhadap *database* jawaban.

### III.6 Analisis Kelas

Dalam menentukan kelas-kelas yang akan digunakan dalam aplikasi pembelajaran bahasa inggris ini, analisa yang terdapat di dalamnya yaitu PilihKategoriSoal, JawabSoalSpeak dan JawabSoalListen. Berikut merupakan gambar analisis kelas :



Gambar 10 Analisis Kelas

Gambar 10 menjelaskan tentang analisis kelas pada aplikasi ini. Adapun rincian kelas – kelas pada gambar tersebut dijelaskan pada tabel 1 sebagai berikut :

<b>Jenis Kelas</b>	<b>Nama Kelas</b>	<b>Deskripsi</b>
Kelas Boundary	PilihKategoriSoal	Kelas ini berhubungan dengan use case Pilih KategoriSoal
	JawabSoalSpeak	Kelas ini berhubungan dengan use case Jawab Soal Speak
	JawabSoalListen	Kelas ini berhubungan dengan use case Jawab Soal Listen
Kelas Control	KontrolKategoriTopik	Mengatur kategori dan topik soal
	KontrolSoal	Mengatur penampilan soal pada sistem
	KontrolJawaban	Mengatur jawaban tiap soal dan pencocokan antara soal dengan jawaban
	KontrolKebenaran	Menampilkan hasil kebenaran dari jawaban soal
Kelas Entity	Soal	Menyimpan informasi berupa kategori dan wacana
	RinciSoal	Menyimpan Informasi berupa soal dan jawaban

**Tabel 1 Spesifikasi Analisis Kelas**

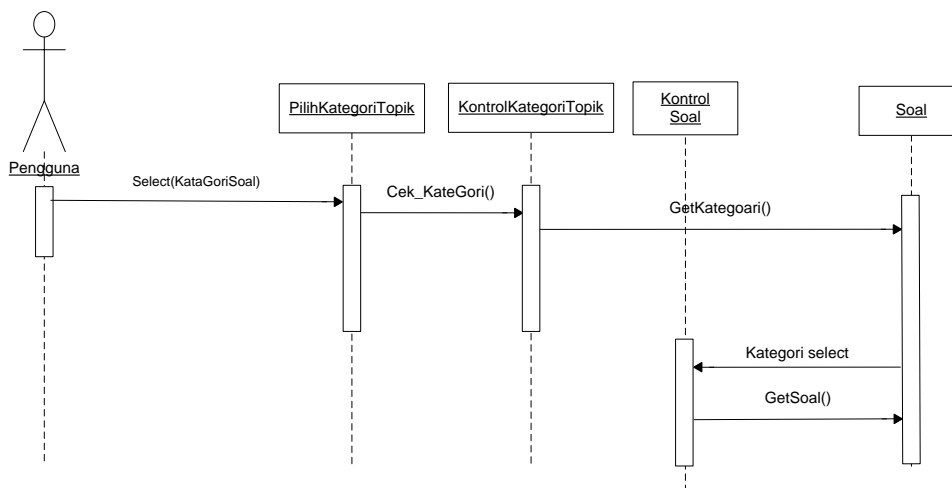
## Bab IV Perancangan

Pada bab ini akan dijelaskan tentang sequence diagram, diagram class, dan perancangan antarmuka.

### IV.1 Sequence Diagram

Sequence diagram merupakan gambaran dari skenario use case pada bab sebelumnya yang merupakan respon dari sebuah *event* untuk menghasilkan *output* tertentu. Detil sequence diagram dari sistem adalah sebagai berikut:

#### IV.1.1 Sequence Diagram PilihKategoriSoal

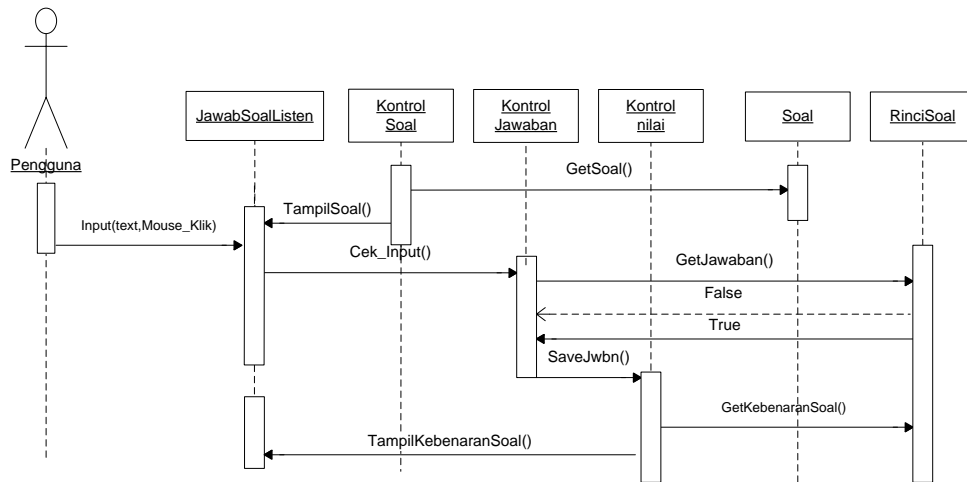


Gambar 11 Sequence diagram PilihKategoriSoal

Pada gambar sequence diagram Pilih Kategori Soal ini pengguna memilih kategori yang di sediakan yaitu kategori *Listening* dan *Speaking* yang berisi beberapa topik pembelajaran bahasa inggris. Kemudian setelah memilih kategori dan topik yang akan di gunakan untuk belajar maka sistem akan mengambil soal dari *database* sesuai kategori dan topik yang di pilih oleh pengguna.



### IV.1.3 Sequence Diagram JawabSoalListen



Gambar 13 Sequence diagram JawabSoalListen

Pada sequence diagram JawabSoalListen ini melanjutkan sequence diagram PilihKategoriSoal yaitu setelah sistem mengambil soal dari *database* sesuai kategori dan topik yang di pilih oleh pengguna yaitu berupa kategori *Listening* dan di tampilkan di layar. Kemudian pengguna mengerjakan soal yang telah di sediakan berupa suara dari sistem dan pengguna harus mendengarkannya dengan baik kemudian jabannya di ketik di tempat yang telah di sediakan lalu sistem akan mengecek kebenaran dari jawaban yang di ucapkan pengguna. Setelah soal terjawab semua atau waktu pengerjaan soal berakhir maka sistem akan memberikan informasi tentang berapa soal yang di jawab pengguna dengan benar.

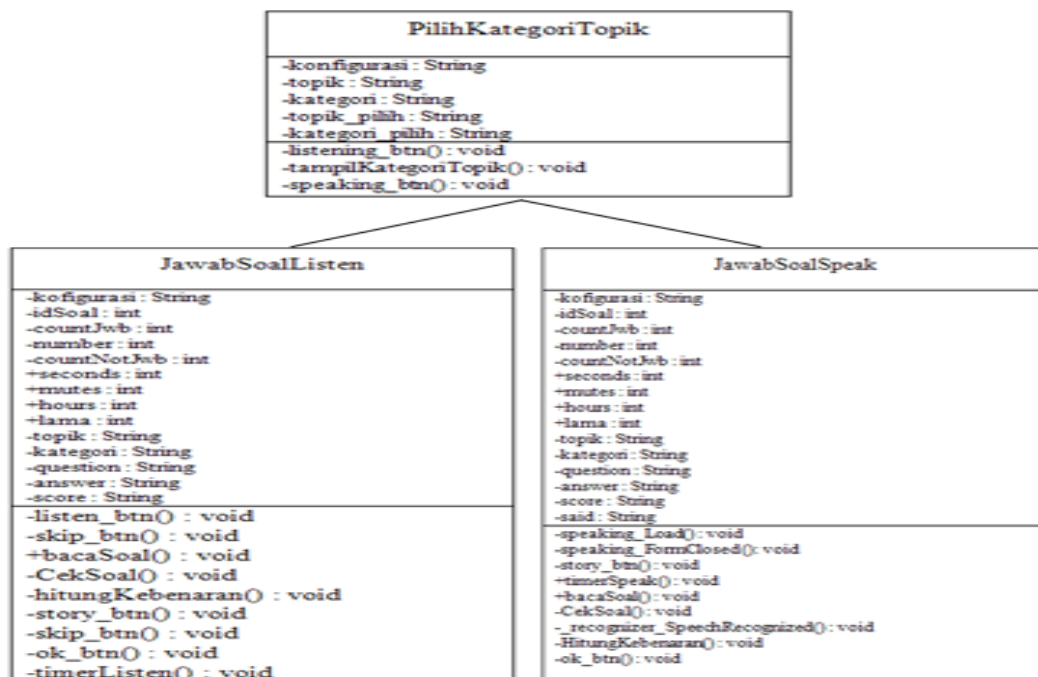
## IV.2 Diagram Kelas

Diagram kelas adalah diagram yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu dengan yang lain, serta dimasukkan pula atribut dan operasi. Untuk mempermudah pengelolaan hubungan antar kelas ada beberapa kelas yang digabungkan dan juga penambahan kelas kontrol. keterangan diagram fix kelas sistem ini adalah sebagai berikut:

Kelas Fix	Jenis	Nama Kelas Sebelumnya	Jenis	Keterangan
PilihKategoriTopik	Boundry	PilihKategoriTopik	Boundary	Digabungkan
		Kontrol KategoriTopik	Control	
		KontrolSoal	Control	
		Soal	Entity	
JawabSoalSpeak	Boundry	JawabSoalSpeak	Boundry	Digabungkan
		KontrolJawaban	Control	
		KontrolKebenaran	Control	
		RinciSoal	Entity	
JawabSoalListen	Boundry	JawabSoalListen	Boundry	Digabungkan
		Kontrol Jawaban	Control	
		KontrolKebenaran	Control	
		RinciSoal	Entity	

Tabel 2 keterangan Kelas Fix

Gambar diagram kelas untuk sistem pembelajaran bahasa inggris dapat dilihat seperti pada Gambar Diagram Kelas berikut:



Gambar 14 Diagram Interaksi Kelas

## IV.3 Rancangan Kelas Rinci

### IV.3.1 Kelas PilihKategoriTopik

#### IV.3.1.1 Rinci Kelas PilihKategoriTopik

PilihKategoriTopik
-konfigurasi : String -topik : String -kategori : String -topik_pilih : String -kategori_pilih : String
-listening_btn() : void -tampilKategoriTopik() : void -speaking_btn() : void

#### IV.3.1.2 Algoritma Kelas PilihKategoriTopik

Bagian ini hanya diisi untuk kerangka algoritma untuk operasi/method yang dianggap cukup penting.

Nama Kelas : *PilihKategoriTopik*

Algoritma : *(Algo-001)*

```
//memilih kategori
sql = "select * from soal where kategori='listening'";
atau
sql = "select * from soal where kategori='speaking'";
//menampilkan soal yang tersedia pada richTextBox
view.Items.Add(kategori)
//memilih soal yang akan dikerjakan
sql = "select * from soal where topik='"+topik_pilih+"' and
kategori='"+kategori_pilih+"'";
Wacana bdForm = new Wacana(topik,kategori);
bdForm.Show();
this.Dispose();
```

### IV.3.2 Kelas JawabSoalListen

#### IV.3.2.1 Rinci Kelas JawabSoalListen

Nama Kelas : JawabSoalListen

JawabSoalListen
-kofigurasi : String -idSoal : int -countJwb : int -number : int -countNotJwb : int +seconds : int +mutes : int +hours : int +lama : int -topik : String -kategori : String -question : String -answer : String -score : String
-listen_btn() : void -skip_btn() : void +bacaSoal() : void -CekSoal() : void -hitungKebenaran() : void -story_btn() : void -skip_btn() : void -ok_btn() : void -timerListen() : void

#### IV.3.2.2 Algoritma Kelas JawabSoalListen

Bagian ini hanya diisi untuk kerangka algoritma untuk operasi/method yang dianggap cukup penting.

Nama Kelas : JawabSoalListen

Algoritma : (Algo-002)

```
readonly SpeechSynthesizer _synthesizer = new SpeechSynthesizer();  
String konfigurasi  
int idSoal;
```

```

int countJwb = 0,number=0,countNotJwb=10;
public int seconds, minutes, hours, lama = 2 ;
string topik, kategori,question,answer,score;
//memanggil soal
sql = "select * from rincisoal where kategori='" + _kategori + "' and topik= '" +
_topik + "' and score='salah' ORDER BY RAND() LIMIT 10";
//mendengarkan soal
_synthesizer.Volume = 100;
_synthesizer.Rate = -4;
_synthesizer.Speak(pertanyaan);
//membaca wacana
Wacana bdForm = new Wacana(topik, kategori);
bdForm.Show();
getSelectedTopik();
//menghitung kebenaran
sql1 = "select count(idSoal) from rincisoal where kategori='" + _kategori + "'
and topik= '" + _topik + "' and score='benar'";

```

### IV.3.3 Kelas JawabSoalSpeak

#### IV.3.3.1 Rinci Kelas JawabSoalSpeak

Nama Kelas : JawabSoalSpeak

JawabSoalSpeak
-kofigurasi : String -idSoal : int -countJwb : int -number : int -countNotJwb : int +seconds : int +minutes : int +hours : int +lama : int

-topik : String -kategori : String -question : String -answer : String -score : String -said : String
-speaking_Load() : void -speaking_FormClosed () : void -story_btn() : void +timerSpeak() : void +bacaSoal() : void -CekSoal() : void -_recognizer_SpeechRecognized() : void -HitungKebenaran() : void -ok_btn() : void

#### IV.3.3.2 Algoritma Kelas JawabSoalSpeak

Bagian ini hanya diisi untuk kerangka algoritma untuk operasi/method yang dianggap cukup penting.

Nama Kelas : JawabSoalSpeak

Algoritma : (Algo-003)

```

private SpeechRecognitionEngine _recognizer = new SpeechRecognitionEngine();
readonly SpeechSynthesizer _synthesizer = new SpeechSynthesizer();
String konfigurasi
int idSoal;
int countJwb = 0, number = 0, countNotJwb = 10;
public int seconds, minutes, hours, lama = 5;
string topik, kategori, question, answer, score,said;

_recognizer.SetInputToDefaultAudioDevice();
_recognizer.LoadGrammar(new DictationGrammar());

//membaca wacana
Wacana bdForm = new Wacana(topik, kategori);
bdForm.Show();

```

```

getSelectedTopik();
//melafalkan jawaban
_recognizer.RecognizeAsync(RecognizeMode.Multiple); //mulai menangkap suara
_recognizer.RecognizeAsyncStop(); //berhenti menangkap suara
//membaca wacana
Wacana bdForm = new Wacana(topik, kategori);
bdForm.Show();
getSelectedTopik();
//menghitung kebenaran
sql1 = "select count(idSoal) from rincisoal where kategori='" + _kategori + "'
and topik= '" + _topik + "' and score='benar'";
//menghitung kebenaran
sql1 = "select count(idSoal) from rincisoal where kategori='" + _kategori + "'
and topik= '" + _topik + "' and score='benar'";

```

## IV.4 Perancangan Antarmuka

Bagian ini diisi rancangan antarmuka untuk kelas-kelas yang diidentifikasi sebagai objek antar muka.

### IV.4.1 Inteface welcome



Gambar 15 interface welcome

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>terima</i>	<i>Button</i>	<i>accept</i>	<i>Masuk ke menu utama</i>
<i>richTextBox1</i>	<i>richTextBox</i>	<i>Welcome</i>	<i>Menampilkan cara pemakaian aplikasi</i>
<i>Title1</i>	<i>Lable1</i>	<i>Title</i>	<i>Menampilkan judul dan ajakan belajar</i>

Tabel 3 Deskripsi Tampilan interface welcome

#### IV.4.2 Interface Pilih Kategori dan Topik Soal

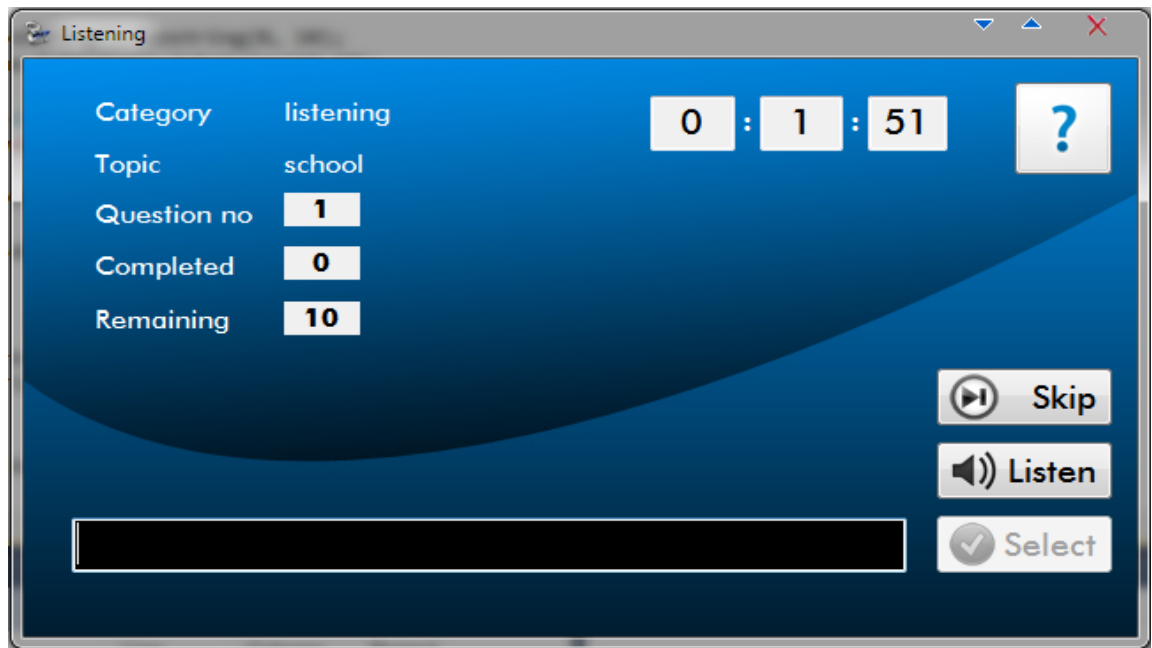


Gambar 16 Tampilan Menu Kategori Soal

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>ViewTopik</i>	<i>RichTextbox</i>	<i>view</i>	<i>Melihat topik yang telah di pilih berdasarkan kategori</i>
<i>Listen</i>	<i>Button</i>	<i>listening</i>	<i>Menampilkan list topik listening</i>
<i>speak</i>	<i>Button</i>	<i>speaking</i>	<i>Menampilkan list topik speaking</i>

Tabel 4 Deskripsi Tampilan Pilih Kategori Soal

#### IV.4.3 Interface Jawab Soal Listen



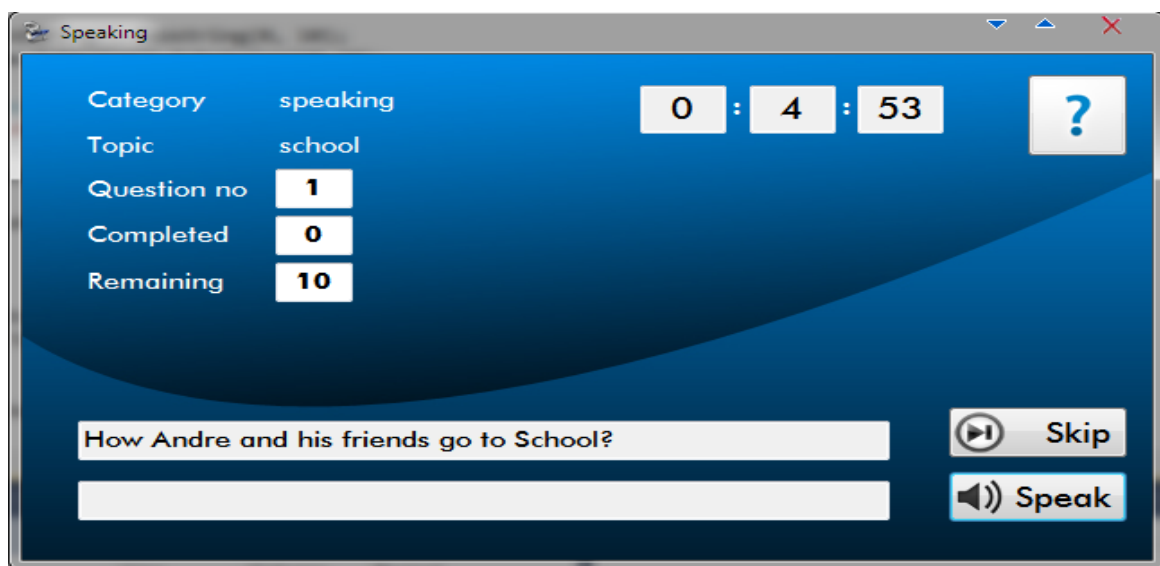
Gambar 17 Tampilan Jawab Soal Listen

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>LblKategori</i>	<i>Label</i>	<i>View kategori</i>	<i>Menampilkan kategori soal yang sedang dikerjakan</i>
<i>LblTopik</i>	<i>Label</i>	<i>Category</i>	<i>Menampilkan topic soal yang sedang dikerjakan</i>
<i>Lblhr</i>	<i>TextBox</i>	<i>Jam</i>	<i>Waktu (jam)</i>
<i>Lblmin</i>	<i>TextBox</i>	<i>Menit</i>	<i>Waktu (menit)</i>
<i>Lblsec</i>	<i>TextBox</i>	<i>Detik</i>	<i>Waktu (detik)</i>
<i>noText</i>	<i>TextBox</i>	<i>Question No</i>	<i>Menampilkan nomor soal</i>
<i>AnswerTxt</i>	<i>TextBox</i>	<i>Answered</i>	<i>Menampilkan jumlah soal yang telah terjawab</i>
<i>NotAnswerTxt</i>	<i>TextBox</i>	<i>NotYet Answered</i>	<i>Menampilkan jumlah soal yang belum terjawab</i>
<i>BtnHelp</i>	<i>Button</i>	<i>Help</i>	<i>Menampilkan wacana</i>
<i>BtnSkip</i>	<i>Button</i>	<i>Skip</i>	<i>Melewatkan soal ke soal yang selanjutnya</i>

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>BtnOk</i>	<i>Button</i>	<i>ok</i>	<i>Memilih soal yang telah dijawab</i>
<i>BtnLstn</i>	<i>Button</i>	<i>listening</i>	<i>Untuk mendengarkan soal</i>
<i>AnswerInput</i>	<i>TextBox</i>		<i>Tempat masukkan jawaban</i>

**Tabel 5 Deskripsi Tampilan Jawab Soal Listening**

#### **IV.4.4 Interface Jawab Soal Speak**



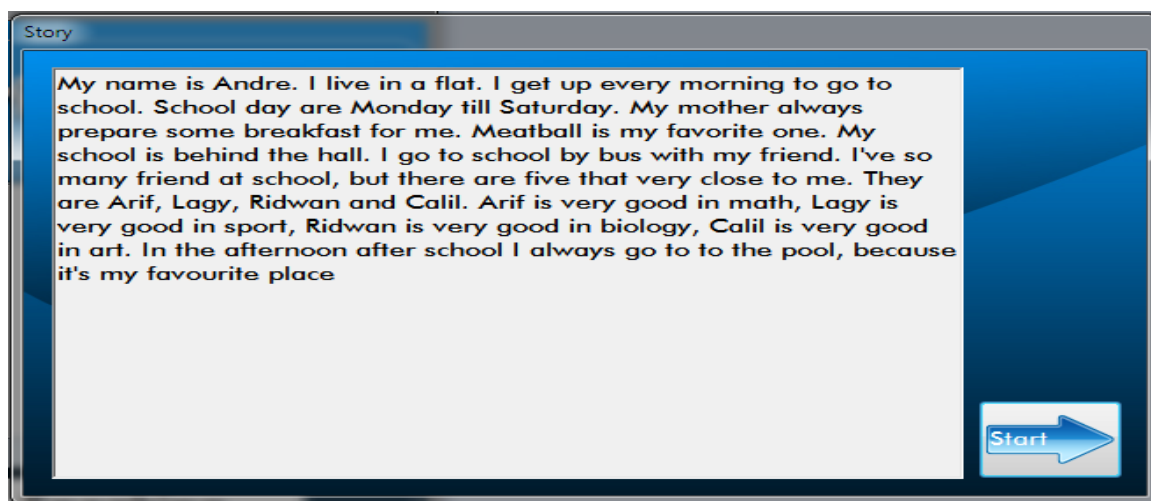
**Gambar 18 Tampilan Jawab Soal Speak**

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>LblKategori</i>	<i>Label</i>	<i>Category</i>	<i>Menampilkan kategori soal yang sedang dikerjakan</i>
<i>LblTopik</i>	<i>Label</i>	<i>Topic</i>	<i>Menampilkan topic soal yang sedang dikerjakan</i>
<i>AnswerInputTtTt</i>	<i>TextBox</i>	<i>Category</i>	<i>Menampilkan suara yang di tangkap</i>
<i>Lblhr</i>	<i>TextBox</i>	<i>Jam</i>	<i>Waktu (jam)</i>
<i>Lblmin</i>	<i>TextBox</i>	<i>Menit</i>	<i>Waktu (menit)</i>
<i>Lblsec</i>	<i>TextBox</i>	<i>Detik</i>	<i>Waktu (detik)</i>
<i>noText</i>	<i>TextBox</i>	<i>Question No</i>	<i>Menampilkan nomor soal</i>

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>AnswerTxt</i>	<i>TextBox</i>	<i>Answered</i>	<i>Menampilkan jumlah soal yang telah terjawab</i>
<i>NotAnswerTxt</i>	<i>TextBox</i>	<i>NotYet Answered</i>	<i>Menampilkan jumlah soal yang belum terjawab</i>
<i>Button4</i>	<i>Button</i>	<i>Help</i>	<i>Menampilkan wacana</i>
<i>Button3</i>	<i>Button</i>	<i>Skip</i>	<i>Melewatkan soal ke soal yang selanjutnya</i>
<i>QuestionText</i>	<i>TextBox</i>		<i>Untuk menampilkan pertanyaan</i>
<i>listBox1</i>	<i>TextBox</i>		<i>Untuk menampilkan list yang di ucapkan</i>

**Tabel 6** Diskripsi tampilan interface jawab soal speak

#### IV.4.5 Interface wacana



**Gambar 19** interface wacana

<b>Id_Objek</b>	<b>Jenis</b>	<b>Nama</b>	<b>Keterangan</b>
<i>goBt</i>	<i>button</i>	<i>accept</i>	<i>Lanjut masuk tes dengan kategori dan topik yang sudah di pilih</i>
<i>closeHelp</i>	<i>button</i>	<i>Close help</i>	<i>Menutup wacana</i>
<i>ReadingTopik</i>	<i>richTextBox</i>	<i>reading</i>	<i>Menampilkan cerita soal</i>

**Tabel 7** Deskripsi Tampilan interface wacana

## Bab V Implementasi dan Pengujian

Setelah dilakukan tahap perancangan maka tahap selanjutnya adalah implementasi dan pengujian pada perangkat lunak. Implementasi akan menghasilkan aplikasi yang dapat dijalankan di lingkungan operasional. Untuk mengetahui apakah aplikasi tersebut dapat melakukan fungsi sesuai deskripsi perencanaan maka perlu dilakukan pengujian. Adapun fungsi yang akan diuji pada Aplikasi Penunjang Kemampuan Bahasa Inggris Menggunakan Microsoft Speech API sesuai dengan analisis kebutuhan fungsional adalah sebagai berikut:

1. Sistem menampilkan pilihan kategori dan topik soal.
2. Sistem menerima masukan data suara atau teks dari pengguna sebagai jawaban atas soal yang diberikan.
3. Sistem mampu mengenali dan mengecek data suara atau teks masukan dengan jawaban yang diharapkan.
4. Sistem menampilkan hasil penilaian secara langsung dari hasil keseluruhan untuk semua soal tiap kategori dan hanya menampilkan perhitungan benar dan salah saja tidak meliputi penilaian berupa skor.

### V.1 Implementasi Kelas

Bagian ini diisi dengan daftar kelas yang telah diimplementasikan, dapat disajikan dengan tabel berikut.

No	Nama Kelas	Nama File Fisik	Nama File Executable
1	KontrolKatagori Topik	KontrolKatagoriTopik.cs	-
2	JawabSoalListen	JawabSoalListen.cs	-
3	JawabSoalSpeak	JawabSoalSpeak.cs	-

Tabel 8 Implementasi Kelas

## V.2 Pengujian Menampilkan Soal Sesuai Kategori dan Topik

### V.2.1 Skenario Pengujian

Kelas yang di uji:

- PilihKategoriTopik

Pada tahap pertama skenario pengujian pencocokan teks dengan suara ialah :

1. Pengguna memilih kategori yang tersedia pada *list* kategori
2. Kemudian akan tampil pilihan soal sesuai yang telah ditentukan pengguna

### V.2.2 Hasil Pengujian

Kelas	Fungsi	Skenario	Hasil Pengujian	Keterangan
PilihKategori Topik	Pilih soal per kategori	Pengguna memilih kateogri yang tersedia pada list kategori	Tampil pilihan soal sesuai kategori yang telah ditentukan	Sebelum tombol untuk mengikuti tes ditekan harus dipilih salah satu kategori

Tabel 9 Pengamatan I

## V.3 Pengujian Pencocokan Suara dengan Teks

### V.3.1 Skenario Pengujian

Kelas yang di uji:

- JawabSoalSpeak

Pada tahap pertama skenario pengujian pencocokan suara dengan teks ialah :

1. Menampilkan soal berupa teks.
2. Mengucapkan soal yang telah ditampilkan, pengucapan di lakukan oleh beberapa user yang berbeda yaitu laki-laki dan perempuan.

### V.3.2 Hasil Pengujian

Hasil pengujian pencocokan suara dengan teks pada alat masukan berupa dua jenis mic yang berbeda, di mana pada pengujian ini jawaban yang di ucapkan yaitu

jawaban yang sesuai dengan jawaban soal, jawaban yang mirip dan jawaban soal yang salah atau tidak mirip sama sekali, seperti dijelaskan pada tabel pengamatan II dibawah ini:

Pegguna	Alat Masukan		Jawaban Soal	Jawaban Yang di ucapkan	Yang di Harapkan	Hasil Pengujian
	Mic1	Mic2				
Syahril Idwar			No	No	(√)	Sesuai
					(√)	Sesuai
				Now	(X)	Sesuai
					(X)	Sesuai
				Null	(X)	Sesuai
					(X)	Sesuai
				how	(X)	Sesuai
					(X)	Sesuai
Pegguna	Alat Masukan		Jawaban Soal	Jawaban Yang di ucapkan	Yang di Harapkan	Hasil Pengujian
Mic1	Mic2					
Arif Nu'man			Pool	Pool	(√)	Sesuai
					(√)	Sesuai
				Cool	(X)	Sesuai
					(X)	Sesuai

				Paul	( X )	Sesuai
					( X )	Sesuai
				Spool	( X )	Sesuai
					( X )	Sesuai
Pengguna	Alat Masukan		Jawaban Soal	Jawaban Yang di ucapkan	Yang di Harapkan	Hasil Pengujian
	Mic1	Mic2				
Lagiyanto			Five	Five	( √ )	Sesuai
					( √ )	Sesuai
				Wife	( X )	Sesuai
					( X )	Sesuai
				Live	( X )	Sesuai
					( X )	Sesuai
				Wave	( X )	Sesuai
					( X )	Sesuai

Tabel 10 Pengamatan II

Keterangan :

X = Tidak cocok      Mic1 = Laptop *Built-in Microphone*

√ = Cocok              Mic2 = *Microphone Headset*

## **Analisa**

Dari tiga pengguna yang masing-masing melakukan empat percobaan menjawab satu soal dengan jawaban yang berbeda, yaitu satu jawaban yang benar, dua jawaban yang mirip dengan jawaban soal dan satu jawaban yang salah atau tidak mirip sama sekali dengan menggunakan dua jenis *microphone* yang berbeda untuk setiap percobaannya dan hasilnya sesuai dengan yang di harapkan seperti yang tampak di tabel pengamatan II.

## **V.4 Pengujian Pencocokan Teks dengan Suara**

### **V.4.1 Skenario Pengujian**

Kelas yang di uji:

- JawabSoalListen

Pada tahap pertama skenario pengujian pencocokan teks dengan suara ialah :

1. Sistem menyediakan soal berupa suara.
2. Menuliskan soal yang di sebutkan oleh sistem, menulis soal akan di lakukan dengan 5 cara yaitu : menulis dengan huruf kecil semua, menulis dengan huruf besar semua, menulis dengan huruf besar di awal kata, menulis dengan huruf kecil dan besar secara acak dan menulis dengan kata yang salah.

### **V.4.2 Hasil Pengujian**

Hasil pengujian pencocokan teks dengan suara pada 5 jenis penulis dapat dilihat pada tabel pengamatan III sebagai berikut:

<b>Jawaban (Soal Listening)</b>	<b>Inputan Jawaban Jenis Penulisan xxx</b>	<b>Hasil Yang Di Harapkan Tidak Cocok (X)</b>	<b>Hasil Pengujian</b>
Less than one pound	Less thanone pound	X	Sesuai
	Les then one pount	X	Sesuai
Some land	Someland	X	Sesuai
	Same lend	X	Sesuai
Apartment	Apart ment	X	Sesuai
	Apartmen	X	Sesuai
<b>Soal Suara (Soal Listening)</b>	<b>Inputan Jawaban Jenis Penulisan name</b>	<b>Hasil Yang Di Harapkan Cocok (√)</b>	<b>Hasil</b>
Less than one pound	less than one pound	√	Sesuai
Some land	some land	√	Sesuai
Apartment	apartment	√	Sesuai
<b>Soal Suara (Soal Listening)</b>	<b>Inputan Jawaban Jenis Penulisan NAME</b>	<b>Hasil Yang Di Harapkan Cocok (√)</b>	<b>Hasil</b>
Less than one pound	LESS THAN ONE POUND	√	Sesuai
Some land	SOME LAND	√	Sesuai
Apartment	APARTMENT	√	Sesuai

<b>Soal Suara (Soal Listening)</b>	<b>Inputan Jawaban Jenis Penulisan Name</b>	<b>Hasil Yang Di Harapkan Cocok (√)</b>	<b>Hasil</b>
Less than one pound	Less Than One Pound	√	Sesuai
Some land	Some Land	√	Sesuai
Apartment	Apartment	√	Sesuai
<b>Soal Suara (Soal Listening)</b>	<b>Inputan Jawaban Jenis Penulisan nAmE</b>	<b>Hasil Yang Di Harapkan Cocok (√)</b>	<b>Hasil</b>
Less than one pound	LeSs tHan One poUNd	√	Sesuai
Some land	SoME laND	√	Sesuai
Apartment	APaRtMenT	√	Sesuai

**Tabel 11 Pengamatan III**

Keterangan :

X = Tidak cocok

√ = Cocok

**xxx** = menulis jawaban yang salah

**Name** = menulis dengan huruf besar di awal kata

**name** = menulis dengan huruf kecil semua

**nAmE** = menulis dengan huruf besar dan kecil secara acak

**NAME** = menulis dengan huruf besar semua

## **Analisa**

Dari 3 soal yang di tampilkan sistem dalam pengujian di lakukan 5 jenis penulisan seperti di jelaskan di atas. Semua hasil pengujian sesuai dengan yang di harapkan mulai dari jawaban yang salah sistem bisa membedakanya walau penulisannya sama hanya kurang sepasi sistem menganggap salah jawaban tersebut ini sesuai dengan yang di harapkan yaitu tidak cocok. Kemudian untuk empat jenis penulisan selanjutnya semuanya bisa di kenali oleh sistem dengan baik sehingga sesuai dengan hasil yang di harapkan yaitu cocok walaupun penulisannya berbeda pada huruf besar dan kecil tulisan tersebut.

## **Bab VI      Kesimpulan dan Saran**

Bab ini memuat simpulan-simpulan yang merupakan rangkuman dari hasil analisis kinerja pada bagian sebelumnya dan bab ini berisi saran-saran pengembangan dari penelitian yang dibuat dan aspek yang belum terselesaikan.

### **VI.1 Kesimpulan**

Dari bab sebelumnya tentang implementasi dan pengujian aplikasi dapat diambil beberapa kesimpulan yaitu:

1. Para pengguna dapat langsung praktik mendengar, menulis dan melafalkan kalimat bahasa inggris.
2. Pada saat menjawab soal kategori *speaking* hasilnya kurang maksimal bila lingkungan dalam keadaan *noise* yang tinggi.
3. Hasil akhir dapat menampilkan jumlah jawaban yang benar untuk setiap kategori.

### **VI.2 Saran**

Untuk perbaikan aplikasi pembelajaran bahasa inggris ini kedepannya, bagi pengembang ada beberapa saran sebagai berikut :

1. Ruang lingkup pembelajaran sebaiknya ditambah, seperti pembelajaran mengenai grammar, penguasaan kosakata dan tebak gambar.
2. Aplikasi ini akan lebih baik jika menerapkan client dan server.
3. Untuk pengembangan selanjutnya diharapkan dapat menangani noise agar hasilnya dapat maksimal.

## DAFTAR PUSTAKA

- [1]. Andreanna Achmad *WHAT TO SAY*, Surabaya , Arkola,1996,h.3
- [2]. BALITBANG DEPDIKNAS: *Panduan Materi SMP dan MTs*, Jakarta, Januari 2008
- [3]. Betts, M., *The Almanac: Hot Tech*, ComputerWorld 52, 17 November 2003.
- [4]. A.W.M.,Van Den En Den and N.A.M.,Verhoeckx, *Discrete-Time Signal Processing*, An Introduction, Prentice-Hall
- [5]. C.Ludeman,Lonie *Fundamentals of Digital Signal Processing*
- [6]. Hiroshi Kanai, *Delayed Block Transfer Function in the Frequency Domain*, IEEE Transactions on Signal Processing, Vol.42, July 1994.
- [7]. Purba Tambunan, 1984 , *Teknik Audio*,Yayasan Binatronika,
- [8]. R.E.,Bogner, *Introduction to Digital Filtering*, John Wiley & Son Inc.
- [9]. V.Oppenheim,Allen and W. Schafer,Roland ,1975, *Digital Signal Processing*, Prentice-Hall
- [10]. [http://agusza.its-sby.edu/kuliah/citra/bab4\\_detail.html](http://agusza.its-sby.edu/kuliah/citra/bab4_detail.html). Diakses pada tanggal 19 September 2011, pukul 15.05
- [11]. <http://www.dspguru.com/info/faqs/fftfaq.htm>. Diakses pada tanggal 20 Oktober 2011, pukul 14.00
- [12]. <http://mathworld.wolfram.com/FastFourierTransform.html>. Diakses pada tanggal 19 Oktober 2011, pukul 15.30
- [13]. [http://en.wikipedia.org/wiki/Spectrum\\_analyzer](http://en.wikipedia.org/wiki/Spectrum_analyzer). Diakses pada tanggal 20 Oktober 2011 pukul 12.15
- [14]. [http://en.wikipedia.org/wiki/Speaker\\_recognition](http://en.wikipedia.org/wiki/Speaker_recognition). Diakses pada tanggal 15 Oktober 2011 pukul 20.10
- [15]. <http://www.relisoft.com/Science/Physics/fft.html>. Diakses pada tanggal 20 November 2011, pukul 14.30