

**RANCANG BANGUN APLIKASI TEXT TO SPEECH SEBAGAI ALAT
BANTU PEMBELAJARAN BAHASA INGGRIS.**

NASKAH PUBLIKASI



diajukan oleh

Joko Aris Pramono
10.01.2744

kepada
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AMIKOM
YOGYAKARTA
2013

NASKAH PUBLIKASI

**RANCANG BANGUN APLIKASI TEXT TO SPEECH SEBAGAI ALAT
BANTU PEMBELAJARAN BAHASA INGGRIS.**

disusun oleh

**Joko Aris Pramono
10.01.2744**

Dosen Pembimbing



**Bayu Setiaji, S. Kom.
NIK. 190000003**

Tanggal, 12 Februari 2013

**Ketua Jurusan
Teknik Informatika**



**Hanif Al Fatta, M. Kom.
NIK. 190302096**

**DESIGNING AND BUILDING TEXT TO SPEECH APPLICATION FOR ENGLISH
LANGUAGE LEARNING TOOLS.**

**RANCANG BANGUN APLIKASI TEXT TO SPEECH SEBAGAI ALAT BANTU
PEMBELAJARAN BAHASA INGGRIS.**

Joko Aris Pramono
Bayu Setiaji
Jurusan Teknik Informatika
STMIK AMIKOM YOGYAKARTA

ABSTRACT

Mastery of English as an international language is one of the access to success in every field. One purpose of the national curriculum in English education is to develop communication skills that include listening (listening), speaking (speaking), reading (reading), and writing (writing) in English.

Here the authors wanted to focus on making the application as a tool for learning English. In English, the pronunciation of a word is different from writing. Plus there are differences in pronunciation and writing a few words between English version with English British American version. To help overcome this, the authors want to create an application that can convert text into audio that is known as text to speech method. This application is expected to be helpful for students and the general public to be able to speak and learn English properly.

Making these applications using the Java programming language and the software used is Eclipse, and runs within the Android operating system.

Keywords: text to speech, English language learning tool, convert text to audio.

1. Pendahuluan

Penguasaan bahasa Inggris sebagai bahasa Internasional merupakan salah satu akses untuk meraih kesuksesan di segala bidang. Didalam bahasa Inggris, pengucapan dari suatu kata berbeda dengan penulisannya. Ditambah lagi terdapat perbedaan pengucapan dan penulisan beberapa kata antara bahasa Inggris versi British dengan bahasa Inggris versi American. Bahasa Inggris British cenderung mempertahankan ejaan banyak kata yang asalnya dari Perancis, sedangkan Inggris American mencoba untuk mengeja kata lebih mendekati cara mereka melafalkannya dan mereka menghilangkan huruf-huruf yang tidak diperlukan.

Untuk mengatasi permasalahan tersebut, penulis ingin membuat sebuah aplikasi *Text To Speech*, yaitu suatu aplikasi yang digunakan untuk mengkonversikan tulisan/teks kedalam bentuk ucapan dengan menggunakan pemodelan bahasa natural manusia. Dengan menggunakan aplikasi *Text To Speech* ini diharapkan pengguna bisa mempelajari cara pengucapan bahasa Inggris dengan tepat, baik versi British ataupun versi American.

Disini penulis ingin membuat aplikasi mobile *Text To Speech* yang berjalan di platform Sistem Operasi Android karena saat ini ponsel Android sedang berkembang pesat sehingga pengguna bisa menggunakan aplikasi ini di ponsel Android dengan mudah tanpa harus membeli PC/komputer untuk menjalankan aplikasinya.

2. Landasan Teori

2.1 Konsep Dasar

2.1.1 Teknologi Pemrosesan Bahasa

Bahasa dapat dibedakan menjadi 2, yaitu bahasa alami dan bahasa buatan. Bahasa alami adalah bahasa yang biasa digunakan untuk berkomunikasi antar manusia, misalnya bahasa Indonesia, Inggris, Jepang, dan sebagainya.¹ Bahasa Buatan adalah bahasa yang dibuat secara khusus untuk memenuhi kebutuhan tertentu, misalnya bahasa pemrograman komputer.²

Suatu sistem pemrosesan bahasa alami secara lisan dapat dibentuk dari tiga sub-sistem, yaitu sebagai berikut :

- a. Sub-Sistem *Natural Language Processing* (NLP), berfungsi untuk melakukan pemrosesan secara simbolik terhadap bahasa tulisan. Beberapa bentuk aplikasi

¹ Handi Dwi Rachma, Zonda Rugmiaga, Miftahul Huda, *Pembuatan Text-To-Speech Synthesis System Untuk Penutur Berbahasa Indonesia*, The 13th Industrial Electronics Seminar 2011 (IES 2011) Electronic Engineering Polytechnic Institute of Surabaya (EEPIS), Indonesia, October 26, 2011, hal 311

² *Ibid.*

sub-sistem ini adalah translator bahasa alami, sistem pemeriksaan sintaks bahasa, sistem yang dapat menyimpulkan suatu narasi, dan sebagainya.

- b. Sub-Sistem *Text-to-Speech* (TTS), berfungsi untuk mengubah teks (bahasa tulisan) menjadi ucapan (bahasa lisan).
- c. Sub-Sistem *Speech Recognition* (SR), yaitu sistem yang berfungsi untuk mengubah atau mengenali suatu ucapan (bahasa lisan) menjadi teks (bahasa tulisan).

2.1.2 Text To Speech

Pada dasarnya *Text-to-Speech* adalah suatu sistem yang dapat mengubah teks menjadi ucapan.³ Suatu pensintesa ucapan atau *Text to Speech* pada prinsipnya terdiri dari dua sub sistem, yaitu :

- a. Bagian konverter teks ke fonem (*Text to Phoneme*)
- b. Bagian konverter fonem ke ucapan (*Phoneme to Speech*)

Phonem adalah istilah linguistik dan merupakan satuan terkecil dalam sebuah bahasa yang masih bisa menunjukkan perbedaan makna.⁴ 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 *phonem*, durasi serta *pitch*-nya.

Bagian konverter *phonem* ke ucapan akan menerima masukan berupa kode-kode *phonem* serta *pitch* dan durasi yang dihasilkan oleh bagian sebelumnya. Berdasarkan kode-kode tersebut, bagian konverter *phonem* ke ucapan akan menghasilkan bunyi atau sinyal ucapan yang sesuai dengan kalimat yang ingin diucapkan.

Konversi dari teks ke *phonem* sangat dipengaruhi oleh aturan-aturan yang berlaku dalam suatu bahasa. Pada prinsipnya proses ini melakukan konversi dari simbol-simbol tekstual menjadi simbol-simbol *phonetik* yang merepresentasikan unit bunyi terkecil dalam suatu bahasa.

2.2 Pemrograman Java

2.2.1 Java

Java menurut definisi dari Sun adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer *standalone* ataupun pada lingkungan jaringan. Java2 adalah generasi kedua dari Java platform (generasi pertamanya adalah Java Development Kit).

³ *Ibid*, hal 312

⁴ *Ibid*.

2.2.2 Kategori Pemrograman Java

Platform Java memiliki tiga buah edisi yang berbeda, yaitu *J2EE (Java2 Enterprise Edition)*, *J2ME (Java2 Micro Edition)*, dan *J2SE (Java2 Standart Edition)*.

2.2.2.1 Java2 Standart Edition (J2SE)

J2SE adalah inti dari bahasa pemrograman Java. *JDK (Java Development Kit)* adalah salah satu tool dari *J2SE* untuk mengompilasi dan menjalankan program Java.

2.2.2.2 Java2 Micro Edition (J2ME)

J2ME adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak Java pada barang elektronik beserta perangkat pendukungnya. *J2ME* biasa digunakan pada telepon seluler, pager, *personal digital assistants (PDA's)* dan sejenisnya.

2.2.2.3 Java2 Enterprise Edition (J2EE)

J2EE adalah kelompok dari beberapa API dari Java dan teknologi selain Java. *J2EE* dibuat untuk membuat aplikasi yang kompleks. *J2EE* sering dianggap sebagai *middleware* atau teknologi yang berjalan di server, namun sebenarnya *J2EE* tidak hanya sebatas untuk itu.

2.3 Sistem Operasi Android

2.3.1 Sejarah Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi.⁵ Android menyediakan platform yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/*smartphone*.

2.3.2 The Dalvik Virtual Machine (DVM)

Salah satu elemen kunci dari Android adalah *Dalvik Virtual Machine (DVM)*. Android berjalan didalam *Dalvik Virtual Machine (DVM)* bukan di *Java Virtual Machine (JVM)*, sebenarnya banyak persamaannya dengan *Java Virtual Machine (JVM)* seperti *J2ME (Java Mobile Edition)*, tetapi Android menggunakan *virtual machine* sendiri yang dirancang untuk memastikan bahwa beberapa fitur-fitur berjalan lebih efisien pada perangkat mobile.

2.3.3 Android SDK (Software Development Kit)

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada platform Android menggunakan

⁵ Nazruddin Safaat H, *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*, (Bandung: Informatika, 2011), hal 1

bahasa pemrograman Java. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.

2.4 Software yang digunakan

2.4.1 Eclipse

Eclipse adalah IDE untuk pengembangan java/android yang free dan dapat didownload di <http://www.eclipse.org/download/>, versi eclipse yang ada sekarang sudah banyak seperti *Eclipse Helios* (eclipse versi 3.6), *Eclipse Indigo*, *Eclipse Galileo* (eclipse versi 3.5), dan *Eclipse Ganymede* (eclipse versi 3.4).⁶ Eclipse versi 3.4 sudah support dengan *Android Development Tool (ADT)* untuk membuat *eclipse* dapat digunakan untuk *coding* project Android. Eclipse merupakan IDE yang paling populer untuk pengembangan Android karena memiliki Android *plug-in* yang tersedia untuk memfasilitasi pengembangan Android.

ADT adalah kepanjangan dari *Android Development Tools* yang menjadi penghubung antara IDE Eclipse dengan Android SDK.⁷ Setelah selesai instalasi eclipse kita dapat melakukan instalasi Android Development Tools (ADT) dengan menggunakan koneksi internet atau menggunakan file jar untuk instalasi tanpa menggunakan koneksi internet.

2.4.2 Adobe Photoshop CS3

Adobe Photoshop CS3 merupakan salah satu software pengolah gambar untuk gambar bitmap keluaran dari Adobe System Inc. Sebagai salah satu software yang sangat handal, Photoshop telah banyak digunakan oleh para professional untuk membuat dan memanipulasi gambar, baik untuk keperluan media percetakan, internet, maupun elektronik.

2.4.3 The Unified Modeling Language

The Unified Modeling Language merupakan seperangkat aturan dan notasi untuk spesifikasi sistem perangkat lunak, dikelola dan dibuat oleh *Object Management Group* yang digunakan untuk memodelkan sebuah perangkat lunak.

UML digunakan untuk memvisualisasikan, membangun, dan mendokumentasikan rancangan sistem dari sebuah perangkat lunak. Diagram UML antara lain terdiri dari *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.

⁶ *Ibid*, hal 16

⁷ *Ibid*, hal 17

2.4.3.1 Use Case Diagram

Usecase Diagram merupakan alat komunikasi tingkat tinggi untuk mewakili persyaratan dari sebuah sistem. Diagram menunjukkan interaksi antara pengguna dan entitas eksternal lainnya dengan sistem yang sedang dikembangkan.

2.4.3.2 Activity Diagram

Activity Diagram menangkap alur dari sebuah sistem, termasuk tindakan utama dan poin keputusan. Diagram ini berguna untuk mendokumentasikan proses bisnis.

2.4.3.3 Class Diagram

Class Diagram menggambarkan struktur statis dari kelas dalam sistem yang akan dibuat dan menggambarkan atribut, operasi dan hubungan antara kelas.

2.4.3.4 Sequence Diagram.

Sequence Diagram secara khusus menjabarkan sebuah Use Case. Diagram ini menunjukkan sejumlah objek dan pesan yang melewati suatu objek.

3. Gambaran Umum

3.1 Analisis Kebutuhan Fungsional

Aplikasi *Text To Speech* ini dapat digunakan untuk membantu pelajar atau masyarakat pada umumnya untuk mempelajari dan meningkatkan kualitas berbicara menggunakan bahasa Inggris. Dalam bahasa Inggris, pengucapan bahasa berbeda dengan penulisannya. Selain itu, pengguna bisa mempelajari perbedaan pengucapan bahasa Inggris versi British dan versi American.

Cara menggunakan aplikasi ini cukup mudah, yaitu pengguna memilih versi bahasa Inggris yang akan dipelajari pengucapannya lalu memasukkan kata kedalam *EditText*, setelah itu tekan tombol *Speak* maka akan keluar suara pengucapan dari kata atau kalimat yang diketikkan di *EditText*.

Aplikasi ini juga bisa digunakan untuk membantu komunikasi bagi orang bisu. Stephen Hawking, ahli fisika teoritis menggunakan aplikasi *text to speech* setelah gejala *sklerosis lateral amiotrofik* (ALS) yang akan membuatnya kehilangan hampir seluruh kendali *neuromuskulanya* mulai muncul. Pada tahun 1985, ia terkena penyakit *pneumonia* dan harus dilakukan *trakeostomi* sehingga ia tidak dapat berbicara sama sekali. Seorang ilmuwan Cambridge membuat alat yang memperbolehkan Hawking menulis apa yang ingin ia katakan pada sebuah komputer, lalu akan dilafalkan melalui sebuah *voice synthesizer*.

3.2 Kebutuhan Perangkat Keras

Perangkat keras yang dibutuhkan sebagai sarana penunjang untuk menjalankan aplikasi *Text To Speech* berupa perangkat *smartphone* Android dengan spesifikasi Sistem Operasi Android versi 1.6 (*Donut*) karena sudah mendukung teknologi *text to*

speech, akan tetapi penulis membuat aplikasi ini dengan target Sistem Operasi Android versi 2.2 keatas.

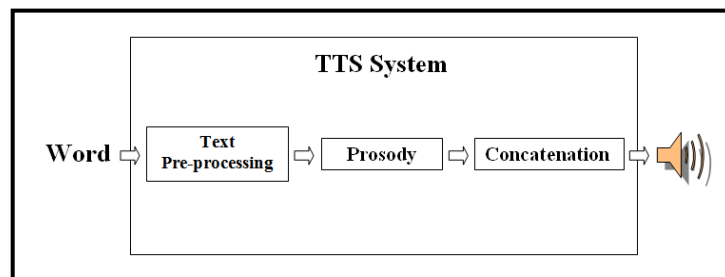
3.3 Kebutuhan Perangkat Lunak

Agar aplikasi *text to speech* dapat berjalan dengan baik, diperlukan perangkat lunak lain untuk pembuatan sistem. Dalam penelitian ini penulis menggunakan perangkat lunak diantaranya sebagai berikut:

1. *Eclipse*
2. *Java* sebagai bahasa pemrograman.

3.4 Sistem Text To Speech

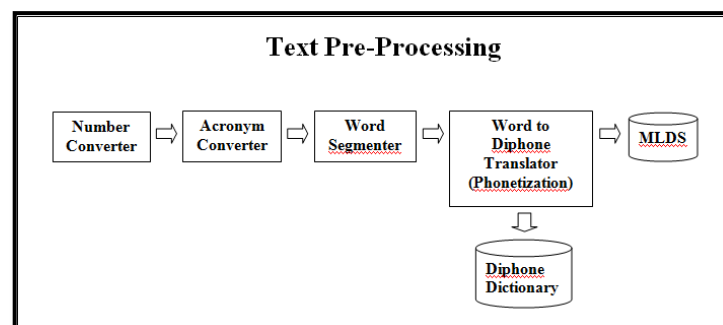
Text to Speech synthesis system terdiri dari 3 bagian, yaitu *text pre-processing*, pembangkitan *prosody* dan *concatenation*. Di bawah ini adalah diagram blok *text to speech synthesis system* :



Gambar 3.1. Blok diagram text to speech synthesis system

a. *Text pre-processing*

Yaitu pengkonversian dari input yang berupa teks menjadi *diphone* (gabungan dua buah fonem). Ketika input yang berupa teks, akronim (singkatan) ataupun angka maka bagian ini akan mengkonversikan menjadi *diphone* yang telah tersedia di database *diphone*. Diagram blok untuk proses *text pre-processing* adalah :



Gambar 3.2. Blok diagram text pre-processing

Dari blok diagram sistem dapat dijelaskan cara kerja sistem yaitu :

1. *Number Converter*

Jika input pada sistem berupa angka, maka system mengkonversikan angka ke dalam representasi diphone (gabungan dua buah fonem), misal **0.25 menjadi point two five.**

2. *Acronym Converter*

Jika input pada sistem berupa kata singkatan dalam bahasa Inggris, maka sistem mengkonversika singkatan ke dalam representasi diphone (gabungan dua buah fonem), misalnya **Mr. menjadi Mister**

3. *Word Segmenter*

Jika input pada sistem berupa kata atau kalimat maka sistem mengkonversikan kata atau kalimat ke dalam representasi diphone (gabungan dua buah fonem).

4. *Diphone Dictionary*

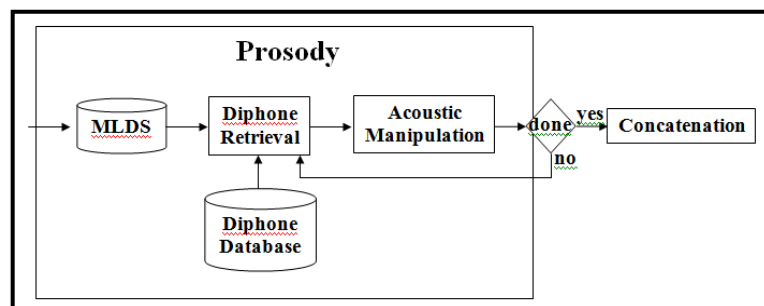
Merupakan database yang berupa kumpulan dari diphone – diphone. Pembuatan diphone dilakukan dengan melakukan pelabelan pada sinyal wicara.

5. *MLDS (Multi Level Data Structure)*

Terdiri dari semua data yang diperlukan untuk sub system berikutnya, dalam hal ini adalah proses prosody. MLDS terdiri dari representasi diphone-diphone hasil pengkonversian inputan

b. Prosody

Yaitu untuk mendapatkan ucapan yang lebih alami, ucapan yang dihasilkan harus memiliki intonasi (*prosody*). Secara kuantisasi, prosodi adalah perubahan nilai pitch (frekuensi dasar) selama pengucapan kalimat dilakukan atau pitch sebagai fungsi waktu. Prosodi bersifat sangat spesifik untuk setiap bahasa, sehingga model yang diperlukan untuk membangkitkan data-data prosodi menjadi sangat spesifik juga untuk suatu bahasa. Diagram blok untuk prosodi adalah :



Gambar 3.3. Blok diagram prosody

1. *Diphone Retrieval*

Didalamnya terdapat tiga tahapan yang terjadi, yaitu database hasil perekaman *diphone*, menyimpan bentuk gelombang *diphone* dan *Prosodic* parameter dalam variabel.

2. *Accoustic Manipulation*

Di dalamnya terdapat proses pengenalan file-file gelombang .WAV(load, play, write), vast array dari peralatan signal processing, builtin function, ease debugging , GUI-capable

c. **Concatenation**

Yaitu penggabung-gabungan segmen-segmen bunyi yang telah direkam sebelumnya. Setiap segmen berupa *diphone* (gabungan dua buah fonem). Pada perekaman suara dilakukan beberapa kali agar mendapatkan hasil yang akurat.

4. Pembahasan

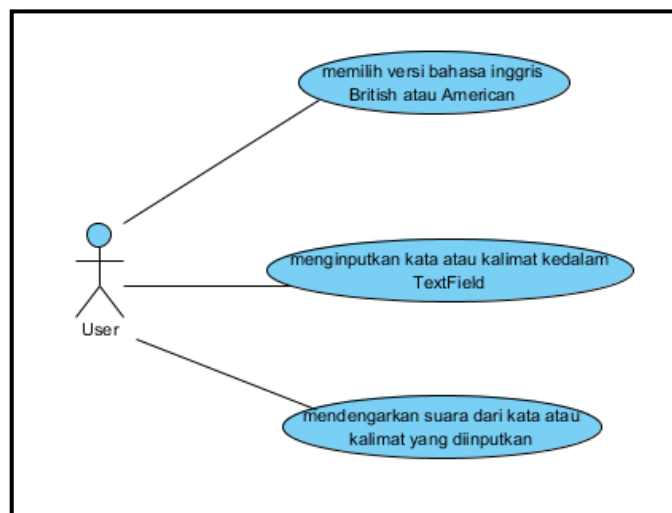
4.1 Perancangan Sistem

4.1.1 Perancangan Unified Modelling Language

Unified Modelling Language (UML) merupakan bentuk perancangan dan dokumentasi perangkat lunak berbasis pemrograman berorientasi objek. Pada aplikasi ini, bentuk perancangan UML menggunakan beberapa diagram, yaitu *Usecase Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.

4.1.1.1 *Usecase Diagram*

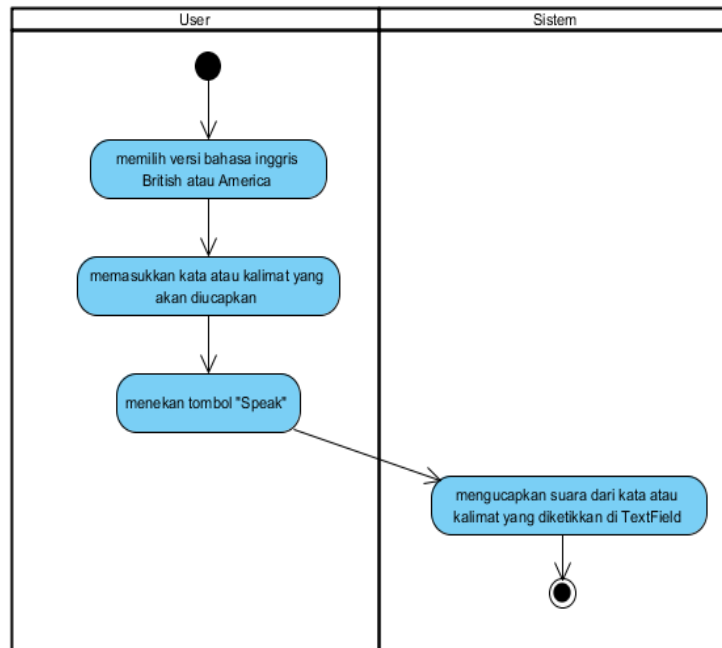
Usecase Diagram digunakan untuk menggambarkan interaksi antara user dengan perangkat lunak. Pada aplikasi *text to speech* ini, user dapat memilih versi bahasa Inggris British atau American, menginputkan kata atau kalimat kedalam TextField, mendengarkan suara dari kata atau kalimat yang diinputkan.



Gambar 4.1. *Usecase diagram text to speech*

4.1.1.2 Activity Diagram

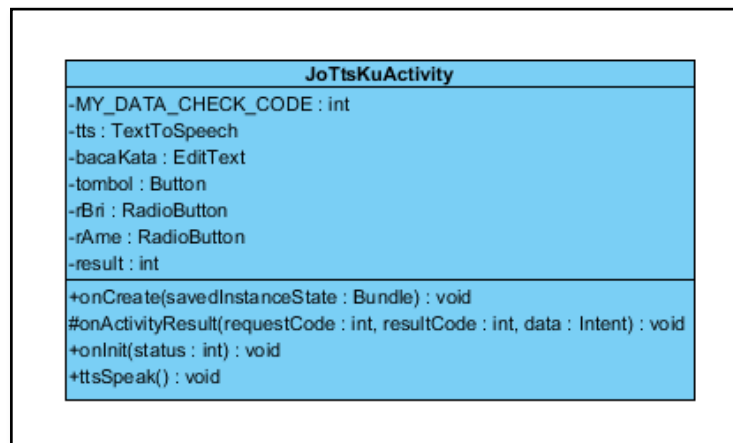
Diagram ini akan menunjukkan alur dari sistem, termasuk tindakan utama dan titik keputusan. Aktivitas sistem dari aplikasi ini didokumentasikan ke dalam *Activity Diagram* sebagai berikut:



Gambar 4.2. Activity diagram text to speech

4.1.1.3 Class Diagram

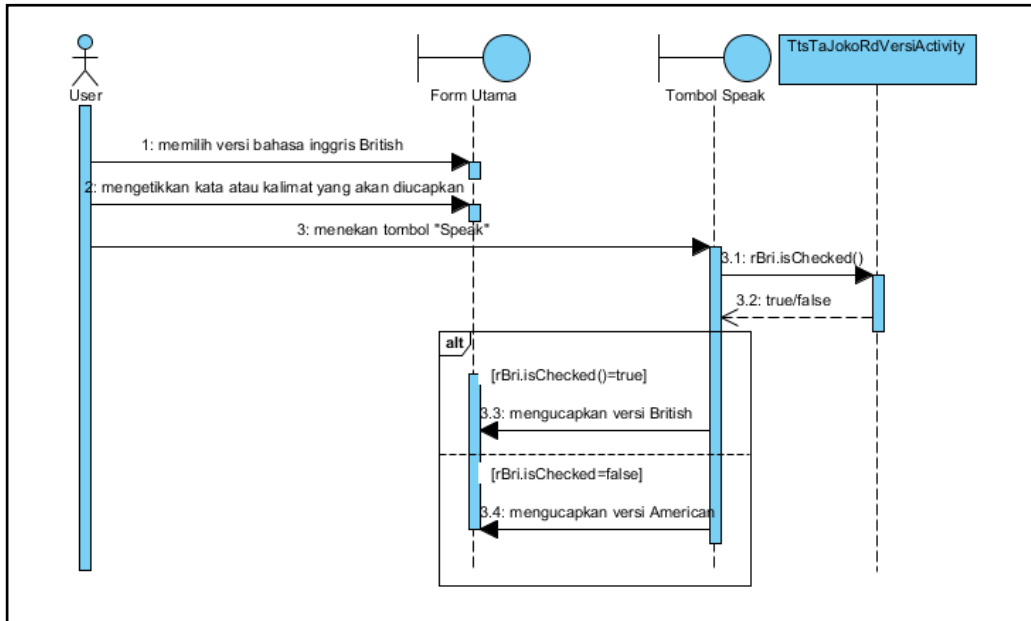
Gambaran *Class Diagram* dari aplikasi ini adalah sebagai berikut:



Gambar 4.3. Class diagram text to speech

4.1.1.4 Sequence Diagram

Sequence Diagram mendeskripsikan bagaimana entitas dalam sistem berinteraksi, termasuk pesan yang digunakan saat interaksi. Berikut adalah sequence diagram dari aplikasi ini:



Gambar 4.4. Sequence diagram text to speech

4.2 Perancangan Tampilan Aplikasi

Perancangan tampilan adalah rancangan yang menggambarkan tampilan aplikasi pada saat digunakan oleh user. Perancangan tampilan untuk aplikasi *text to speech* adalah sebagai berikut:



Gambar 4.5. Rancangan Aplikasi text to speech

4.3 Implementasi Program

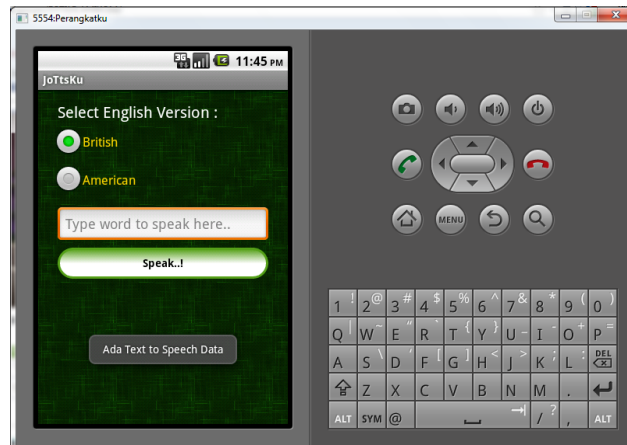
4.3.1 Kode Program

Fungsi dari metode ini adalah untuk menghasilkan suara dari masukan kata dan melakukan validasi jika masukan masih kosong atau belum mengetikkan kata/kalimat. Berikut adalah kode program untuk menghasilkan suara ketika tombol "Speak" ditekan yang memanggil metode `ttsSpeak()`.

```
public void ttsSpeak(){
    String kata = bacaKata.getText().toString();
    if(kata != null && kata.length() > 0){
        tts.speak(kata, TextToSpeech.QUEUE_FLUSH, null);
    }
    else{
        String validasi = "Type word to speak here..";
        tts.speak(validasi, TextToSpeech.QUEUE_FLUSH,
            null);
    }
}
```

4.3.2 Screenshot Program

Dibawah ini adalah tampilan dari aplikasi ketika dijalankan di emulator.



Gambar 4.6. Screenshot aplikasi di emulator

4.3.3 Cara Install Project ke Handphone Android

Cara melakukan penginstallan aplikasi dari project di Eclipse kedalam handphone Android yaitu dengan mengambil file `.apk` yang terdapat di folder `bin` kemudian diinstal kedalam handphone Android.

4.3.4 Uji Coba Program

Tabel 4.1. Hasil Pengujian Program

No	Merk dan Seri HP	Perangkat Keras	Perangkat Lunak / Versi Android	Hasil	Keterangan
1	Samsung Galaxy Young (GT-S5360)	Processor ARMv6 830 MHz, Internal memory 180 Mb, 290 Mb user available RAM, 3 inches TFT Capacitive touchscreen.	Versi 2.3.6 Gingerbread	Aplikasi berjalan lancar	-
2	Tablet TABZ Z1S	Processor ARM Cortex A8 1.5Ghz, 4GB Internal Memory, Memory 512MB RAM DDR3, 512MB ROM, 7 inches TFT WXGA Capacitive Touchscreen.	Versi 4.0 Ice Cream Sandwich	Aplikasi berjalan lancar	Terdapat pilihan menggunakan Google Text to Speech Engine atau Pico TTS
3	Sony Ericsson WT19i Live with Walkman	CPU Qualcomm MSM8255 Snapdragon1 GHz Scorpion, GPU Adreno 205, 320MB storage, 512 MB RAM, Capacitive touchscreen, Layar 3.2 inches.	Versi 2.3.4 Gingerbread	Aplikasi berjalan lancar	-
4	Samsung Galaxy Wonder (W)	CPU : 1.4 GHz Scorpion, 1.7 Gb storage, 512 RAM, 2Gb ROM,	Versi 2.3.5 Gingerbread	Aplikasi berjalan lancar	-

		TFT capacitive touchscreen 3.7 inches.			
5	Sony Ericsson Xperia Mini SP15i	CPU 1 GHz Scorpion, Memory Internal 320Mb, 512 RAM, LED-backlit LCD capacitive touchscreen 3.0 inches	Versi 4.0 Ice Cream Sandwich	Aplikasi berjalan lancar	Saat pertama kali dijalankan langsung masuk ke playstore untuk mendownload TTS Data
6	Sony Ericsson SK17i XPERIA Mini Pro	CPU 1 GHz Scorpion, Memory 400 Mb storage, 512 Mb RAM, LED-backlit LCD, capacitive touchscreen 3.0 inches	Versi 4.0 Ice Cream Sandwich	Aplikasi berjalan lancar	Saat pertama kali dijalankan langsung masuk ke playstore untuk mendownload TTS Data
7	Samsung Galaxy Ace S5830	CPU 800 MHz ARM 11, Memory 158 Mb storage, 278 Mb RAM, TFT capacitive touchscreen 3.5 inches.	versi 2.3.6 Gingerbread	Aplikasi berjalan tapi tidak mengeluarkan suara	Belum terdapat TTS Data dan tidak langsung menuju playstore

8	Samsung Galaxy Mini GT-S570	CPU 600 MHz processor, Memory Internal 160 Mb, TFT capacitive touchscreen 3.14 inches.	versi 2.3.6 Gingerbread	Aplikasi berjalan tapi tidak mengeluarkan suara	Belum terdapat TTS Data dan tidak langsung menuju playstore
---	-----------------------------	--	-------------------------	---	---

4.3.5 Kelebihan dan Kekurangan Program

4.3.5.1 Kelebihan

Kelebihan dari aplikasi ini antara lain:

- a. Menggunakan *Google/ PicoTTS Engine Text to Speech* yang menghasilkan suara dengan kualitas bagus sehingga pengucapan bahasa terdengar jelas.
- b. Terdapat pilihan versi bahasa yaitu Inggris British dan American

4.3.5.2 Kekurangan

Kekurangan dari aplikasi ini antara lain:

- a. Belum bisa menghasilkan suara di semua handphone Android.
- b. Tidak semua handphone Android memiliki TTS Data sehingga harus download TTS Data terlebih dahulu di *playstore* agar aplikasi dapat berjalan lancar.

5. Penutup

5.1 Kesimpulan

Berdasarkan hasil dari pembahasan dan uji coba program, dapat disimpulkan bahwa aplikasi *Text To Speech* telah berhasil dibuat dengan pilihan versi bahasa Inggris British dan American. Poin-poin kesimpulan secara mendetail dari hasil uji coba program adalah sebagai berikut:

1. Aplikasi text to speech ini dapat digunakan sebagai alat bantu pembelajaran bahasa Inggris khususnya belajar pengucapan, sebagaimana yang kita tahu bahwa penulisan dan pengucapan dalam bahasa Inggris itu berbeda.
2. Dengan menggunakan aplikasi ini user juga dapat mempelajari perbedaan pengucapan bahasa Inggris versi British dan American, meskipun sama-sama bahasa Inggris tapi terdapat perbedaan gaya pengucapan.
3. User juga bisa belajar pengucapan angka dan singkatan dengan benar.
4. Aplikasi ini bisa berjalan dalam mode offline sehingga user tidak perlu menggunakan koneksi internet, namun jika belum terdapat TTS Data pada perangkat *Smartphone* Android, saat pertama kali aplikasi dijalankan maka akan langsung menuju *playstore* untuk mendownload TTS Data.

5.2 Saran

Dari hasil pembahasan dan uji coba program, terdapat beberapa kekurangan yang dimasukkan ke dalam saran agar kekurangan-kekurangan tersebut dapat diperbaiki di masa yang akan datang. Poin-poin saran adalah sebagai berikut:

1. Aplikasi ini belum bisa berjalan pada semua *smartphone* Android sehingga masih diperlukan perbaikan-perbaikan di kode program agar bisa berjalan di semua *smartphone* Android.

2. Ditambah fitur untuk mengatur *pitch* dan *speedrate* agar user bisa mengatur cepat lambatnya pengucapan.
3. Untuk kedepan bisa gabung dengan aplikasi berbasis *voice recognition* dan *translator*, sehingga user dapat menginputkan teks menggunakan suara dalam bahasa Indonesia, kemudian di *translate* kedalam bahasa Inggris, selanjutnya aplikasi *text to speech* mengucapkan kata bahasa Inggrisnya.

DAFTAR PUSTAKA

Rachma, H.D, Rugmiaga, Z., Huda, M., 2011. *Pembuatan Text-To-Speech Synthesis System Untuk Penutur Berbahasa Indonesia*. The 13th Industrial Electronics Seminar 2011 (IES 2011):311-313.

Safaat H, Nazruddin. 2011. *Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika Bandung.

Shalahuddin, M dan Rosa A.S. 2008. *Pemrograman J2ME Belajar Cepat Pemrograman Perangkat Telekomunikasi Mobile*. Bandung: Informatika Bandung.

<http://android-developers.blogspot.com/2009/09/introduction-to-text-to-speech-in.html>.

Diakses pada tanggal 25 Oktober 2012.

<http://mobile.tutsplus.com/tutorials/android/android-sdk-using-the-text-to-speech-engine/>.

Diakses pada tanggal 16 Oktober 2012.

<http://swanson.github.com/blog/2012/01/30/repeating-background-textures-in-android.html>. Diakses pada tanggal 8 Desember 2012.

<http://twoh.web.id/2012/06/android-button-style-guidelines-membuat-gradient-pada-tombol/>. Diakses pada tanggal 5 Desember 2012