

# **BAB 1**

## **PENDAHULUAN**

### **1.1. Latar Belakang Masalah**

Oktaf suara merupakan bagian penting dalam menyanyikan sebuah lagu. Oktaf suara adalah nada-nada yang berada pada rentang dengan nama sama dan memiliki jarak 12 langkah nada. Satu oktaf memiliki 8 nada. Oktaf biasanya digunakan untuk menentukan nada dasar dan menyusun paduan suara berdasarkan karakteristik suara. Terdapat 5 jenis karakter suara, yaitu Bass, Baritone, Tenor, Alto dan Soprano. Karakter bass dan baritone dimiliki oleh lelaki sedangkan alto dan soprano dimiliki oleh perempuan. Karakter tenor dapat dimiliki oleh perempuan dan laki-laki [1].

Ketidaktepatan penentuan oktaf suara akan membuat penyanyi kesulitan membuat nada-nada yang berada di luar oktaf suaranya. Nada-nada yang dikeluarkan akan terdengar sumbang. Hal ini akan menurunkan performa penyanyi. Kesalahan penentuan oktaf suara akan mengakibatkan susunan penyanyi paduan suara tidak optimal. Penyanyi paduan suara akan kehilangan karakteristik suara yang dimilikinya. Hal ini akan menurunkan performa paduan suara. Ketidaksesuaian nada dasar penyanyi dan alat musik juga dapat menyebabkan lagu terdengar sumbang.

Penentuan oktaf suara biasanya dilakukan sebelum menyanyikan lagu pada saat latihan. Penentuan oktaf suara secara manual biasanya dilakukan menggunakan piano, keyboard atau gitar. Cara manual ini merupakan proses yang

cukup rumit dan sulit bagi orang awam, sehingga memerlukan seorang musisi yang cukup handal dalam bernyanyi dan menggunakan piano atau gitar. Ketiadaan musisi dan alat musik menjadi hambatan bagi seseorang yang ingin berlatih menyanyi secara mandiri. Masalah ini dapat dipecahkan dengan pembuatan aplikasi untuk menentukan oktaf suara yang dimudah dipahami dan digunakan oleh orang awam.

Aplikasi penentuan oktaf suara akan dibuat untuk perangkat dengan sistem operasi Android dengan alasan bahwa pada saat ini banyak orang menggunakan *tablet* dan *smartphone* dengan sistem operasi Android. Android adalah sistem operasi bergerak berbasis Linux yang dapat dimasukkan ke dalam berbagai perangkat keras, seperti *smartphone*, *tablet*, *e-reader*, *netbook*, *mp4 player* dan *tv*. Android merupakan sistem operasi yang gratis. Orang yang ingin menggunakan sistem operasi Android dapat mengunduh Android dari internet dengan bebas [2].

Penentuan nada-nada yang dinyanyikan oleh seseorang merupakan masalah yang kompleks sehingga diperlukan bantuan sebuah algoritma dan metode untuk memecahkan masalah tersebut. Penelitian ini akan menggunakan algoritma *Goertzel* dan *Nearest Neighbor Search*. Algoritma *Goertzel* digunakan untuk menentukan frekuensi. Algoritma *Nearest Neighbor Search* digunakan untuk menentukan oktaf suara dengan mencocokkan frekuensi suara dengan data yang ada.

*Goertzel* merupakan algoritma yang mengimplementasikan algoritma *Discrete Fourier Transform* (DFT). Algoritma *Goertzel* memiliki keunggulan perhitungan trigonometri lebih sederhana dibandingkan DFT dan perhitungan

lebih efisien dibandingkan FFT [3]. Algoritma ini dapat digunakan untuk mengenali frekuensi dari Dual-tone multi-frequency (DTMF) [4][5] dan menganalisis spektrum suara [6] pada sistem telekomunikasi. *Nearest neighbor search* (NNS) adalah algoritma untuk mencari titik terdekat. Titik terdekat ditandai dengan jarak terdekat dengan titik yang dicari. Metode ini dapat digunakan untuk pengenalan pola, klasifikasi statistik dan penilaian kesamaan skor [7]. Algoritma ini memiliki keunggulan sederhana, efisien dan efektif untuk pengenalan pola, kategorisasi teks dan pengenalan obyek [8].

Berdasarkan uraian diatas maka penulis bermaksud mengimplementasikan algoritma *Goertzel* dan *Nearest Neighbor Search* untuk mentukan oktaf suara pada sebuah aplikasi untuk perangkat dengan sistem operasi Android. Penelitian ini akan menggunakan judul "Implementasi Metode *Goertzel* Dan *Nearest Neighbor Search* Untuk Menentukan Oktaf Suara".

## 1.2. Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan diatas, maka rumusan masalah dari penelitian ini adalah bagaimana cara membangun aplikasi menentukan oktaf suara yang kemudian dirinci sebagai berikut :

1. Bagaimanakah cara membangun aplikasi untuk menentukan oktaf suara yang dapat menangkap sinyal suara, mengolah menggunakan algoritma *Goertzel* dan menentukan oktaf suara menggunakan metode *Nearest Neighbor Search*?

2. Bagaimana kinerja algoritma *Goertzel* dan *Nearest Neighbor Search* yang dapat mengenali frekuensi dan menentukan oktaf suara sehingga menghasilkan keluaran yang sesuai.

### 1.3. Tujuan Penelitian

Berdasarkan latar belakang di atas, hendaklah ada tujuan yang ingin dicapai. Beberapa tujuan yang ingin dicapai dalam penelitian ini yaitu sebagai berikut :

1. Menerapkan algoritma *Goertzel* untuk menentukan frekuensi suara.
2. Menerapkan algoritma *Nearest Neighbor Search* untuk menentukan oktaf suara.
3. Membuat aplikasi android untuk menganalisis dan mengenali oktaf dari suara yang diucapkan .
4. Menguji kemampuan mengenali oktaf suara pada aplikasi yang telah dibuat.

### 1.4. Batasan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka dibuat batasan masalah agar ruang lingkup penelitian skripsi ini jelas batasannya. Adapun batasan masalah yang dibuat sebagai berikut :

1. Input aplikasi adalah suara orang yang direkam menggunakan perangkat Android
2. Input suara direkam dalam format WAV.

3. Proses yang akan dilakukan penentuan frekuensi dan oktaf suara.
4. Output aplikasi berupa frekuensi, nama nada dan nama oktaf suara.
5. Aplikasi dibangun menggunakan bahasa pemrograman JAVA.
6. Aplikasi dibangun menggunakan Eclipse ADT.
7. Aplikasi yang dibuat adalah aplikasi jenis *prototype*.
8. Suara yang diproses adalah satu nada yang dikeluarkan oleh seseorang.
9. Aplikasi tidak memproses suara dari satu lagu penuh.

### 1.5. Metodologi Penelitian

Metodologi penelitian digunakan untuk membantu memecahkan masalah pada penelitian ini. Metodologi penelitian meliputi metode pengumpulan data, pemodelan sistem dan pengembangan sistem.

#### 1. Metode pengumpulan data

Metode yang digunakan untuk mengumpulkan data dalam penelitian ini adalah metode observasi, wawancara, kajian pustaka dan kuisioner.

- a. Metode observasi yang dilakukan dengan mengamati secara langsung aplikasi yang telah dibuat serta mencatat oktaf suara hasil deteksi aplikasi.
- b. Metode wawancara yang dilakukan dengan dialog kepada responden mengenai oktaf suara mereka.
- c. Metode kajian pustaka yang dilakukan dengan membaca buku, jurnal ilmiah, dan internet yang berkaitan dengan algoritma *Goertzel* dan *Nearest Neighbor Search*, oktaf suara, bahasa java dan android.

## 2. Metode pemodelan sistem

Pemodelan sistem dilakukan menggunakan metode *Unified Modelling Language* (UML). Model yang digunakan adalah diagram *use case*, *activity*, *sequence* dan *class*.

- a. Diagram *use case* menggambarkan rancangan kasus penggunaan pada aplikasi oleh pengguna.
- b. Diagram *activity* menggambarkan rancangan aktivitas dilakukan oleh pengguna dan aplikasi.
- c. Diagram *sequence* menggambarkan rancangan urutan aktivitas yang dilakukan oleh pengguna dan aplikasi.
- d. Diagram *class* menggambarkan rancangan kelas, metode dan properti yang akan digunakan dalam aplikasi.

## 3. Metode pengembangan sistem

Pengembangan sistem menggunakan metode *prototype*. Metode ini menghasilkan aplikasi awal dengan fungsi standar. Aplikasi ini biasanya hanya untuk didemonstrasikan kepada pengguna. Fasilitas tambahan tidak perlu ditambahkan ke dalam aplikasi [9]. Langkah metode *prototype* menurut Pressman [10] adalah sebagai berikut.

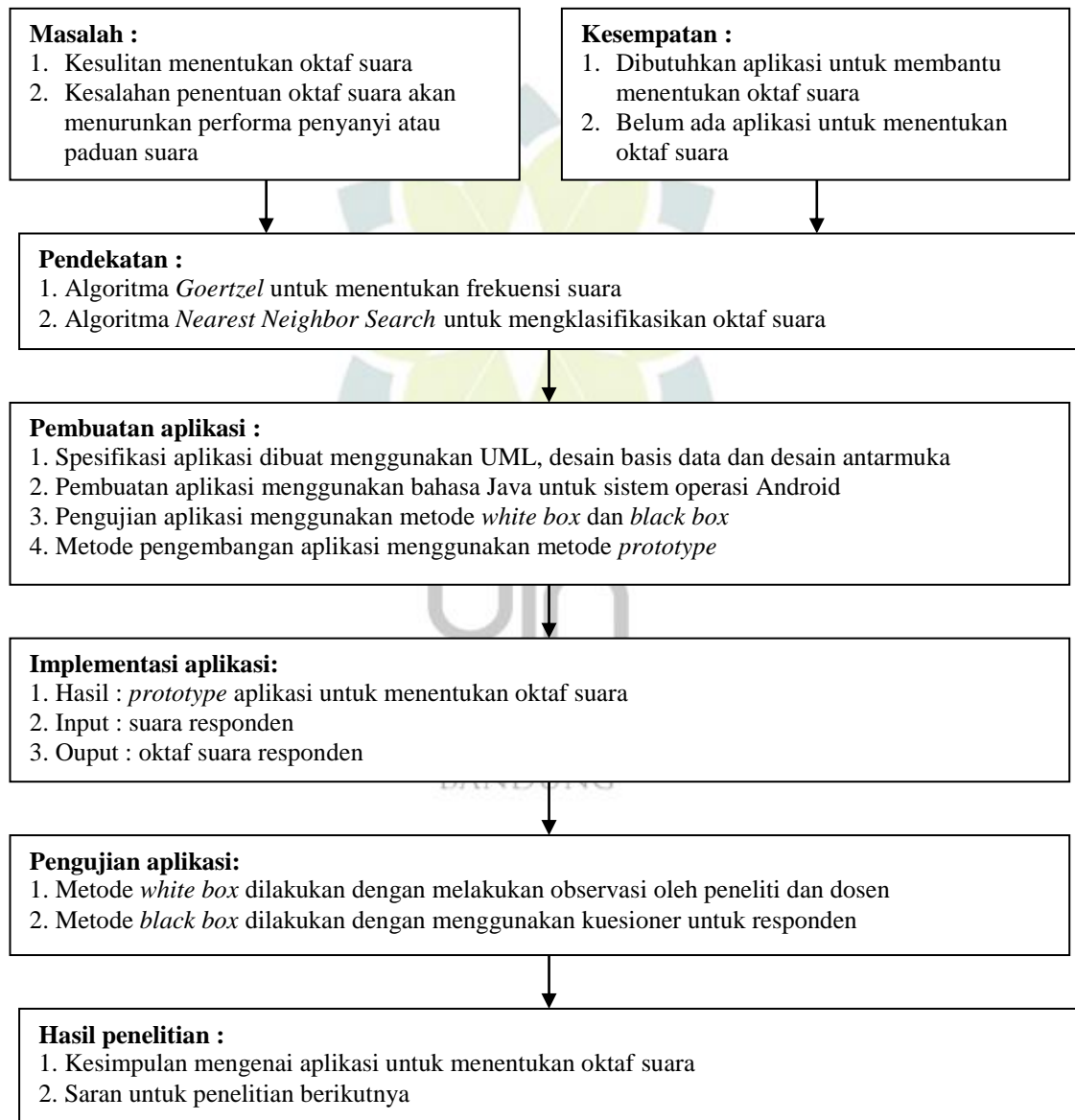
- a. Mendengar kebutuhan pengguna (analisis)
- b. Membangun atau mengembangkan aplikasi (desain dan *coding*)
- c. Pengujian oleh pengguna (*testing*)

Analisis yang akan dilakukan adalah analisis kelemahan sistem lama, kebutuhan fungsi minimal sistem baru serta kebutuhan perangkat lunak dan perangkat keras. Desain dilakukan dengan merancang antarmuka yang akan digunakan. *Coding* dilakukan dengan menulis kode program. *Testing* dilakukan untuk menguji apakah aplikasi sudah sesuai dengan kebutuhan fungsi minimal sistem dan memeriksa apakah aplikasi mengandung kesalahan sintaks dan logika. Pengujian yang dilakukan adalah uji black box.



## 1.6. Kerangka Pemikiran

Kerangka pemikiran yang digunakan terdiri atas masalah, kesempatan, pendekatan, pembuatan aplikasi, implementasi aplikasi, pengujian aplikasi dan hasil penelitian. Berikut ini adalah diagram yang menggambarkan kerangka pemikiran yang digunakan.



**Gambar 1.1** Kerangka Pemikiran Penelitian



## **1.7. Sistematika Penulisan**

Sistematika penulisan merupakan kerangka utama dalam membuat laporan tugas akhir. Sistematika penulisan bertujuan untuk mengarahkan isi laporan berdasarkan bab, judul dan sub judul yang telah ditentukan. Berikut ini adalah sistematika penulisan yang akan digunakan.

### **BAB I PENDAHULUAN**

Bab I menjelaskan masalah-masalah yang menjadi latar belakang dilakukan penelitian. Bab ini meliputi latar belakang, identifikasi masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, kerangka pemikiran dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Bab II berisi tentang teori-teori yang digunakan dalam penelitian ini, baik pada tahap analisis, perancangan, implementasi maupun pengujian.

### **BAB III ANALISIS DAN PERANCANGAN**

Bab III membahas mengenai analisis dan perancangan. Analisis dilakukan terhadap permasalahan yang ada dan kebutuhan yang diperlukan untuk mengatasi permasalahan tersebut. Perancangan mengacu pada hasil analisis. Perancangan meliputi perancangan antarmuka pengguna, desain basis data dan proses. Perancangan menjadi acuan pembuatan aplikasi atau sistem.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN**

Bab IV menjelaskan mengenai implementasi sistem dan pengujian yang telah dilakukan dalam penelitian. Bagian implementasi menguraikan tentang spesifikasi, kebutuhan dan implementasi aplikasi. Bagian pengujian

membahas hasil pengujian yang dilakukan terhadap aplikasi yang telah dibuat.

## **BAB V PENUTUP**

Bab V berisi mengenai kesimpulan dan saran. Kesimpulan merupakan jawaban dari rumusan masalah yang telah ditentukan. Saran diberikan untuk mengatasi kekurangan yang ada pada batasan masalah dan kelemahan sistem pada saat implementasi dan pengujian.

