

TUGAS AKHIR

APLIKASI PIANO DIGITAL MENGGUNAKAN SENSOR LEAP MOTION

*Sebagai salah satu syarat untuk menyelesaikan
Program Studi Diploma III Jurusan Teknik Informatika*



Oleh :

MUHAMMAD HAFIZH ALFALAH
6103181302

JURUSAN TEKNIK INFORMATIKA
PROGRAM STUDI D-III TEKNIK INFORMATIKA
POLITEKNIK NEGERI BENGKALIS

2021

PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa Tugas Akhir ini adalah asli hasil karya saya dan tidak terdapat karya dilakukan untuk memperoleh gelar Ahli Madya di perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau dipublikasikan oleh orang lain, kecuali yang secara tertulis disebutkan sumbernya dalam naskah dalam daftar pustaka.

Bengkalis, 1 September 2021



Muhammad Hafizh Alfalah

LEMBAR PENGESAHAN

APLIKASI PIANO DIGITAL MENGGUNAKAN SENSOR LEAP MOTION

*Sebagai salah satu syarat untuk menyelesaikan
Program Studi Diploma III Jurusan Teknik Informatika*

Oleh :

Muhammad Hafizh Alfalah
6103181302

*Disetujui oleh Tim Penguji Tugas Akhir : Tanggal Ujian : 1 September 2021
Priode Wisuda : XVIII*



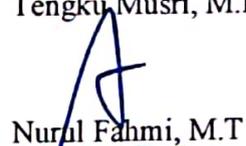
Supria, M.Kom

(Pembimbing)



Tengku Musri, M.Kom

(Penguji 1)



Nurul Fahmi, M.T

(Penguji 2)



Eko Prayitno, M.Kom

(Penguji 3)

Bengkalis, 1 September 2021

Ketua Jurusan Teknik Informatika



Danuri, M.Cs

NIP. 198508122014041001

LEMBAR PENGESAHAN

kami dengan sebenarnya menyatakan bahwa, kami telah membaca keseluruhan dari Tugas Akhir ini, dan kami berpendapat bahwa Tugas Akhir ini layak dan memenuhi syarat untuk memperoleh gelar Ahli Madya.

Tanda Tangan : 
Nama Penguji I : Tengku Musri, M.Kom
Tanggal Pengujian : 1 September 2021

Tanda Tangan : 
Nama Penguji II : Nurul Fahmi, M.T
Tanggal Pengujian : 1 September 2021

Tanda Tangan : 
Nama Penguji III : Eko Prayitno, M.Kom
Tanggal Pengujian : 1 September 2021

**PERNYATAAN PERSETUJUAN
PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN
AKADEMIS**

Sebagai civitas akademik Politeknik Negeri Bengkalis, Saya yang bertanda tangan di bawah ini :

Nama : Muhammad Hafizh Alfalah
NIM : 6103181302
Program Studi : D-III Teknik Informatika
Jenis Karya : Tugas Akhir

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Program Studi D-III Teknik Informatika Politeknik Negeri Bengkalis **Hak Bebas Royalti Noneksklusif** atas karya ilmiah saya yang berjudul :

**APLIKASI PIANO DIGITAL MENGGUNAKAN SENSOR
LEAP MOTION**

Beserta perangkat yang ada. Dengan Hak Bebas Royalti Noneksklusif ini Program Studi Teknik Informatika Politeknik Negeri Bengkalis berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasi Tugas Akhir Saya selama tetap mencantumkan nama Saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Dibuat di : Bengkalis

Pada Tanggal : 1 September 2021

Yang menyatakan



(Muhammad Hafizh Alfalah)

NIM. 6103181302

PERSEMBAHAN

Alhamdulillah kupersembahkan kepada Allah SWT, atas segala rahmat dan juga kesempatan dalam menyelesaikan tugas akhir skripsi saya dengan segala kekurangannya. Segala syukur kuucapkan kepadaMu Ya Rabb, karena sudah menghadirkan orang-orang berarti disekeliling saya. Yang selalu memberi semangat dan doa, sehingga skripsi saya ini dapat diselesaikan dengan baik. Untuk karya yang sederhana ini, maka saya persembahkan untuk ...

Ayahanda dan Ibunda tercinta dan tersayang

Apa yang saya dapatkan hari ini, belum mampu membayar semua kebaikan, keringat, dan juga air mata bagi saya. Terima kasih atas segala dukungan kalian, baik dalam bentuk materi maupun moril. Karya ini saya persembahkan untuk kalian, sebagai wujud rasa terima kasih atas pengorbanan dan jerih payah kalian sehingga saya dapat menggapai cita-cita. Kelak cita-cita saya ini akan menjadi persembahan yang paling mulia untuk Ayah dan Ibu, dan semoga dapat membahagiakan kalian.

Kakak dan Adik tercinta

Untuk Kakakku Indah dan adikku Aufa, tiada waktu yang paling berharga dalam hidup selain menghabiskan waktu dengan kalian. Walaupun saat dekat kita sering bertengkar, tapi saat jauh kita saling merindukan. Terima kasih untuk bantuan dan semangat dari kalian, semoga awal dari kesuksesan saya ini dapat membanggakan kalian.

Dosen Pembimbing

Kepada pak Supria selaku dosen pembimbing saya yang paling baik dan bijaksana, terima kasih karena sudah menjadi pembimbingku. Terima kasih atas bantuannya, nasehatnya, dan ilmunya yang selama ini dilimpahkan pada saya dengan rasa tulus dan ikhlas.

Sahabat dan seluruh teman di kampus tercinta

Tugas Akhir ini kupersembahkan untuk teman-temanku. Ucapan terima kasih kepada teman-temanku, Bagus, Herdi, Hakim, Aidil, Fakhmi, Anggi, Hafiz, Fikar, Via, Kiki, dan Fera, yang telah menemani selama hampir tiga tahun dan senantiasa memberikan motivasi untuk menjadi lebih baik. Walaupun mulut kalian toxicnya luar biasa.

APLIKASI PIANO DIGITAL MENGGUNAKAN SENSOR LEAP MOTION

Nama : Muhammad Hafizh Alfalah
NIM : 6103181302
Dosen Pembimbing : Supria, M.Kom

Abstrak

Piano adalah alat musik tuts yang diklasifikasikan sebagai instrumen dawai dan perkusi yang dimainkan dengan menekan tuts pada papan piano. Alat musik piano pada saat ini tidak murah, membutuhkan biaya yang besar untuk dapat memilikinya. Dari permasalahan diatas maka di usulkan Aplikasi Piano Digital Menggunakan Sensor *Leap Motion*. Sensor *Leap Motion* yang mengusung metode baru untuk berinteraksi dengan komputer dapat diterapkan dalam pembuatan aplikasi, salah satunya yaitu pembuatan Aplikasi Piano Digital Dengan Menggunakan Sensor *Leap Motion* sebagai alat untuk pengoperasiannya. Perancangan sistem menggunakan *mapping area*, hal ini menentukan *Threshold* (batasan) untuk menghasilkan suara yang berbeda dengan jarak yang diinginkan, dari setiap area akan menghasilkan 8 suara dari aplikasi piano digital tersebut melalui titik koordinat yang telah ditentukan. Aplikasi ini memiliki 7 tingkatan suara, dimulai dari *octaf 1* hingga *octaf 7*. Aplikasi ini dirancang menggunakan *software Netbeans* menggunakan bahasa pemrograman *java*. Uji coba dilakukan kepada 7 orang, hasil dari uji coba menunjukkan bahwa aplikasi berjalan dengan baik, aplikasi dapat digunakan oleh siapa saja di setiap perangkat komputer.

Kata Kunci : Aplikasi, *Leap Motion*, *mapping area*, *Netbeans*, *Java*, Piano

DIGITAL PIANO APPLICATION USING LEAP MOTION SENSOR

Name : Muhammad Hafizh Alfalah
ID Student Number : 6103181302
Supervisor : Supria, M.Kom

Abstract

The piano is a keyed musical instrument classified as a string and percussion instrument that is played by pressing the keys on the piano board. Piano musical instruments at this time are not cheap, it costs a lot to be able to own one. From the above problems, a Digital Piano Application using a Leap Motion Sensor is proposed. The Leap Motion sensor which carries a new method for interacting with the computer can be applied in making applications, one of which is the creation of a Digital Piano Application Using the Leap Motion Sensor as a tool for its operation. The system design uses area mapping, this determines the Threshold (limit) to produce a different sound with the desired distance, from each area will produce 8 sounds from the digital piano application through a predetermined coordinate point. This application has 7 sound levels, starting from octave 1 to octave 7. This application is designed using Netbeans software using the java programming language. The trial was carried out on 7 people, the results of the trial showed that the application was running well, the application can be used by anyone on any computer device.

Keywords: Application, Leap Motion, mapping area, Netbeans, Java, Piano

KATA PENGANTAR

Puji dan Syukur penulis ucapkan kehadiran Tuhan Yang Maha Esa, karena dengan rahmat dan hidayah-Nyalah sehingga penulis dapat menyusun dan menyelesaikan Laporan Tugas Akhir dengan judul “Aplikasi Piano Digital Dengan Menggunakan Sensor *Leap Motion*”. Adapun maksud dan tujuan penulisan laporan ini adalah salah satu syarat bagi setiap mahasiswa untuk menyelesaikan jenjang Diploma III di Politeknik Negeri Bengkalis.

Dalam kesempatan ini, saya mengucapkan ribuan terima kasih kepada berbagai pihak yang telah membantu secara aktif maupun pasif memberikan bimbingan serta pengetahuan sehingga menjadi dasar dan mempermudah penulis dalam menyelesaikan Tugas Akhir. Laporan Tugas Akhir ini tidak lepas dari bantuan beberapa pihak. Oleh sebab itu, saya ucapkan terima kasih kepada :

1. Bapak Johny Custer, ST., MT., selaku Direktur Politeknik Negeri Bengkalis
2. Bapak Danuri, M.Cs., selaku Ketua Jurusan Teknik Informatika Politeknik Negeri Bengkalis.
3. Bapak Muhammad Nasir, M.Kom., selaku Ketua Prodi D-III Teknik Informatika Politeknik Negeri Bengkalis.
4. Bapak Supria, M.Kom., selaku Dosen Pembimbing Tugas Akhir Jurusan Teknik Informatika di Politeknik Negeri Bengkalis.
5. Bapak Tengku Musri., M.Kom., selaku Dosen Penguji I Tugas Akhir Jurusan Teknik Informatika di Politeknik Negeri Bengkalis.
6. Bapak Nurul Fahmi, M.T selaku Dosen Penguji II Tugas Akhir Jurusan Teknik Informatika di Politeknik Negeri Bengkalis.
7. Bapak Eko Prayitno, M.Kom selaku Dosen Penguji III Tugas Akhir Jurusan Teknik Informatika di Politeknik Negeri Bengkalis.
8. Kedua orang tua beserta seluruh keluarga atas segala doa serta dukungan materi maupun moral yang telah diberikan kepada peneliti.

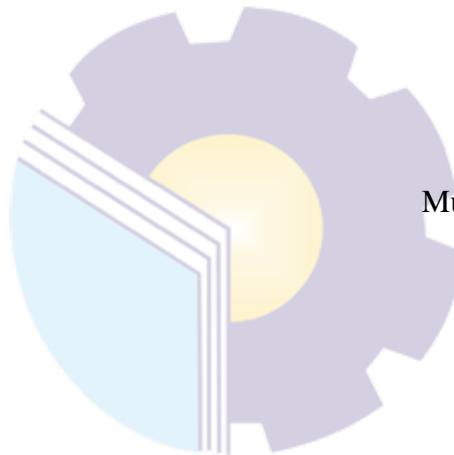
Penulis menyadari bahwa tidak ada yang sempurna di dunia ini, termasuk salah satu diantaranya yaitu dalam penulisan, penyusunan ataupun bahasa pada

Laporan Tugas Akhir ini yang masih jauh dari kata sempurna. Oleh karena itu, penulis dengan senang hati menerima kritik dan saran dari pembaca yang sifatnya membangun.

Semoga Laporan Tugas Akhir ini dapat memberikan manfaat bagi penulis dan juga bagi seluruh pembaca untuk lebih baik di masa yang akan datang. Atas semua perhatian dari seluruh pihak yang telah membantu penulis dalam menyusun Laporan Tugas Akhir ini, Penulis ucapkan Terima kasih.

Bengkalis, 1 September 2021

Penulis



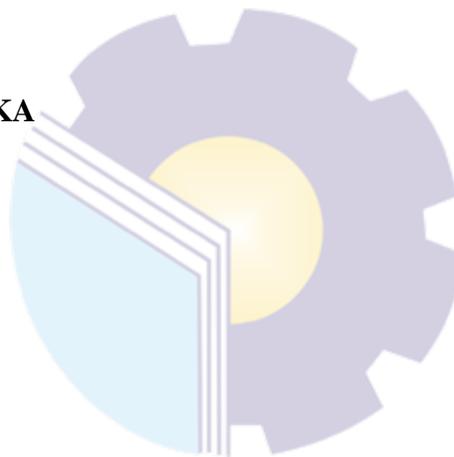
Muhammad Hafizh Alfalah

DAFTAR ISI

	Halaman
HALAMAN SAMPUL DEPAN	
HALAMAN JUDUL	
HALAMAN KEASLIAN	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN	iv
HALAMAN PERSEMBAHAN	v
ABSRTAK	vi
ABSTRACT	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Manfaat	2
1.6 Metode Penyelesaian	3
BAB II TINJAUAN PUSTAKA	5
2.1 Kajian Terdahulu	5
2.2 Landasan Teori	6
2.2.1 Leap Motion	6
2.2.2 Java	6
2.2.3 Piano	7
2.2.4 NetBeans	8
2.2.5 Aplikasi	9

BAB III PERANCANGAN	10
3.1 Bahan dan Alat Penelitian	10
3.1.1 Bahan Penelitian	10
3.1.2 Alat Penelitian	10
3.2 Prosedur Penelitian	11
3.2.1 Identifikasi Masalah	12
3.2.2 Pengumpulan Data	12
3.2.3 Analisa Sistem	12
3.2.4 Perancangan Aplikasi	12
3.2.5 Pembuatan Aplikasi	12
3.2.6 Pengujian Aplikasi	12
3.2.7 Pembuatan Laporan Akhir	13
3.3 Perancangan Aplikasi	13
3.3.1 Analisa Cara Bermain Piano	13
3.3.2 Analisa Aplikasi Piano Yang Diusulkan	13
3.4 Perancangan Sistem	14
3.4.1 Rancangan Diagram Alir Sistem Secara Umum	17
3.4.2 Rancangan Diagram Output Suara	18
3.4.3 Perancang Antarmuka	23
BAB IV HASIL DAN PENGUJIAN	26
4.1 Hasil Implementasi	26
4.1.1 Visualizer Leap Motion	26
4.1.2 Tampilan Icon Software Netbeans	27
4.1.3 SDK Leap Motion LeapDeveloperKit_3.2.1+45911_win	27
4.1.4 Tampilan Dari Software Netbeans	28
4.1.4.1 Class Leap Motion	28
4.1.4.2 Class Main	37
4.1.5. Tampilan Menghubungkan Leap Motion	39
4.1.6 Tampilan Awal Mulai	40

4.1.7 Tampilan Petunjuk	40
4.1.8 Tampilan Main Piano Digital	41
4.1.9 Tampilan Main Saat Dijalankan	42
4.1.10 Tampilan Titik Koordinat Jari Telunjuk	46
4.2 Pengujian Sistem	47
4.2.1 Pengujian Aplikasi	47
4.2.2 Pengujian Suara	48
BAB V PENUTUP	49
5.1 Kesimpulan	49
5.2 Saran	49
DAFTAR PUSTAKA	51



DAFTAR TABEL

	Halaman
Tabel 4.1 Pengujian Aplikasi	47
Tabel 4.2 Pengujian Suara	48



DAFTAR GAMBAR

	Halaman
Gambar 2.1 Leap Motion	6
Gambar 2.2 Java	7
Gambar 2.3 Piano	7
Gambar 2.4 NetBeans IDE	8
Gambar 3.1 Prosedur Penelitian	11
Gambar 3.2 Perancangan Sistem	14
Gambar 3.3 <i>Potional Tracking</i>	15
Gambar 3.4 Penentuan Area	16
Gambar 3.5 Rancangan Diagram Alir Sistem Secara Umum	18
Gambar 3.6 Diagram output suara	20
Gambar 3.7 Perancangan Antarmuka	24
Gambar 4.1 Visualizer Leap Motion	26
Gambar 4.2 Netbeansn IDE 8.0.2	27
Gambar 4.3 Tampilan Dari Netbeans	28
Gambar 4.4 Source Code Leap Motion	31
Gambar 4.5 Sorce Code Suara	34
Gambar 4.6 Source Code Mapping Area	37
Gambar 4.7 Source Code Class Main	38
Gambar 4.8 Tampilan menghubungkan Leap Motion	39
Gambar 4.9 Tampilan Awal	40
Gambar 4.10 Tampilan Petunjuk	41
Gambar 4.11 Tampilan Main	42
Gambar 4.12 Tampilan area satu terbaca	42
Gambar 4.13 Tampilan area dua terbaca	43
Gambar 4.14 Tampilan area tiga terbaca	43

Gambar 4.15 Tampilan area empat terbaca	44
Gambar 4.16 Tampilan area lima terbaca	44
Gambar 4.17 Tampilan area enam terbaca	45
Gambar 4.18 Tampilan area tujuh terbaca	45
Gambar 4.19 Tampilan area delapan terbaca	46
Gambar 4.20 Tampilan Koordinat Jari Telunjuk	46



BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam 10 tahun terakhir, berbagai proyek menggambarkan dampak antar muka baru dalam membangun DMI (*Digital Musical Instrument*). *Leap motion* adalah salah satunya kemajuan teknologi terkini yang mungkin berdampak penciptaan DMI. *Leap Motion* bekerja dengan dua kamera inframerah (IR) dan tiga LED IR sebagai sensor kedalaman dalam bidang pandang terbatas, Menggunakan *stereoscopy* dari kedua kamera, perangkat dapat meminimalkan kesalahan dari alat, dari dan fitur tangan dan dibangun di atas model matematika yang unik, untuk memaksimalkan kecepatan dan presisi. Saat perangkat mendeteksi fitur-fitur ini, perangkat menyediakan pembaruan dalam ingkai data. Setiap bingkai memiliki daftar data pelacakan seperti tangan, jari, alat, gerakan dikenali dan faktor yang menggambarkan gerakan keseluruhan dari adegan itu. (Eduardo dkk. 2013).

Musik merupakan salah satu seni yang menarik banyak perhatian manusia, terlebih lagi alat musik piano. Piano juga dikenal sebagai raja alat musik, karna mudah di mengerti, memiliki wilayah nada yang luas, dapat mewakili semua instrumen musik. Piano juga memiliki suara yang khas, Teknik yang menarik, dan memiliki “kelas” tersendiri bagi manusia. Hal ini yang membuat orang mudah tertarik pada alat musik ini, mulai dari sekedar menjadi penikmat, sebagai sebuah hobi, hingga bagi yang ingin mendalami secara professional. Begitu banyak dengan mudah dijumpai pada masyarakat orang-orang yang dapat memainkan piano, dan dengan bangga menyebut dirinya sebagai pianis. Namun banyak orang tidak mengerti bahwa menikmati, menjadikan hobi, apalagi menjadikan pianis sebagai profesi hidup memerlukan pengorbanan yang sangat tidak mudah. (Genta, 2018).

Leap Motion yang mengusung metode baru untuk berinteraksi dengan komputer dapat diterapkan dalam pembuatan aplikasi, salah satunya yaitu pembuatan Aplikasi Piano Digital Dengan Menggunakan Sensor *Leap Motion* sebagai alat untuk pengoprasiannya. Tujuan dari pembuatan Aplikasi Digital

Menggunakan Sensor *Leap Motion* ini adalah sebagai sarana bermain alat music yang lebih sederhana dan menghemat biaya, karna untuk memiliki ala musik piano butuh banyak biaya yang harus dikeluarkan, dan juga aplikasi Piano Digital ini dapat mempermudah pengguna untuk memainkannya. Pengguna aplikasi ini lasung bisa beinteraksi secara *real-time* seolah-olah pengguna benar-benar memainkan alat music piano secara nyata.

1.2 Rumusan Masalah

Berdasarkan latar belakang di tersebut, rumusan masalah pada penelitian ini adalah “Bagaimana membangun aplikasi piano digital menggunakan sensor *Leap Motion*”.

1.3 Batasan Masalah

Agar tujuan utama tercapai dan pembahasan tidak meluas, adapun Batasan masalah yang diperoleh adalah sebagai berikut :

1. Aplikasi yang dirancang berupa piano digital.
2. Aplikasi yang dibuat merupakan aplikasi desktop.
3. Menggunakan sensor *Leap Motion*.
4. Aplikasi ini hanya menampilkan 8 tuts putih saja yaitu, Do, Re, Mi, Fa, Sol, La, Si, Do

1.4 Tujuan

Tujuan dari peneitian ini adalah membuat aplikasi piano digital yang dapat menghasilkan output berupa suara yang mirip dengan suara aslinya, dengan menggunakan sensor *Leap Motion* dan menggunakan bahasa pemograman *java*.

1.5 Manfaat

Manfaat yang diharapkan dari pengembangan aplikasi ini adalah terciptanya sebuah aplikasi alat musik piano yang menjadi sebagai sarana bermain alat musik yang lebih sederhana dan menghemat biaya.

1.6 Metode Penyelesaian Masalah

Metode yang digunakan dengan melakukan tahapan-tahapan untuk memastikan upaya penelitian dan perancangan aplikasi akan mencapai hasil yang maksimal.

1. Identifikasi Masalah

Identifikasi Masalah dilakukan pertamakali untuk mengetahui permasalahan yang terjadi saat ini.

2. Metode Pengumpulan Data

Prosedur penelitian dalam metodologi penelitian data yang dilakukan dengan studi literatur yaitu mencari informasi dan referensi tentang sistem absensi yang sudah pernah diterapkan berupa Buku, Jurnal, Maupun Internet

3. Perancangan Aplikasi

Aplikasi yang dibuat memiliki sensor gerak terhadap benda dan tangan manusia menggunakan *Leap Motion Controller*. Sensor *Leap Motion* Sebagai alat sensor untuk permainan alat musik piano digital dengan menggunakan perangkat computer yang terhubung, *Leap Motion* ini terus membaca gerak tangan secara terus-menerus (*real time*) dengan aplikasi yang terhubung saat dijalankan maka harus menentukan delay agar tidak terus-menerus membaca gerakan tangan dan benda yang ditangkap oleh sensor *Leap Motion* tersebut

4. Pembuatan Aplikasi

Pembuatan aplikasi menggunakan Bahasa pemrograman *Java*, dan menggunakan aplikasi *NetBeans* untuk mendesain gambar dan menginputkan suara serta codingan.

5. Pengujian Aplikasi

Pengujian aplikasi memerlukan objek atau tangan seseorang di atas sensor *Leap Motion* untuk membaca data-data yang telah di inputkan sebelumnya, pengujian bisa dilakukan lebih dari 1 orang karna aplikasi ini dapat dimainkan oleh siapa saja. Saat tangan di letakkan diatas sensor *Leap Motion* alat ini akan

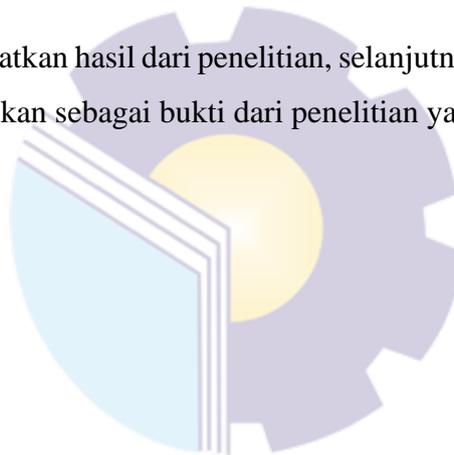
mendeteksi dari gerakan tangan yang di hasilkan area mana yang akan ditangkap dari posisi tangan tersebut apa bila tepat berada pada posisi yang di tetapkan maka aplikasi akan berjalan dan menghasilkan suara yang telah di inputkan, akan tetapi apabila posisi tangan tidak berada pada posisi yang di tetapkan aplikasi tidak akan merespon dan aplikasi tidak akan berbunyi.

6. Hasil Penelitian

Hasil dari penelitian ini dapat berupa sebuah aplikasi piano digital yang dimainkan menggunakan sensor *Leap Motion*.

7. Pembuatan Laporan Penelitian

Setelah mendapatkan hasil dari penelitian, selanjutnya pembuatan laporan yang nantinya akan dijadikan sebagai bukti dari penelitian yang telah dilakukan.



BAB II

TINJAUAN PUSTAKA

2.1 Kajian Terdahulu

Dalam penelitian yang berjudul Evaluasi Awal dari Sensor *Leap Motion* sebagai Pengontrol Alat Musik Digital Baru oleh Eduardo S. Silva dkk (2013). Artikel ini membahas mengenai studi pendahuluan dan evaluasi baru pada sensor untuk membangun DMI (*Digital Musical Instrument*), disini mencantumkan serangkaian gerakan yang dikenali oleh perangkat yang secara teoritis dapat digunakan untuk memainkan sejumlah besar alat musik, kemudian menyajikan analisis presisi dan latensi gerakan.

Dalam penelitian yang berjudul Sensor Gerak Dengan *Leap Motion* Untuk Membantu Komunikasi Tuna Rungu/Wicara oleh Achmad Basuki dkk (2016). Penelitian ini mengenalkan sebuah alat bantu untuk menterjemahkan Bahasa isyarat SIBI menjadi teks atau suara untuk memudahkan komunikasi antara orang tuna rungu/wicara dan orang normal. Alat bantu ini berupa perangkat lunak yang dilengkapi dengan perangkat *Leap Motion* sebagai sensor gerak.

Dalam penelitian yang berjudul Aplikasi Marawis Digital Menggunakan Sensor *Leap Motion* Redondo Fernando dkk (2017). Membahas tentang Penggunaan alat sensor menggunakan *Leap Motion* dan *virtual reality* untuk menggerakkan objek yang ada di dalam komputer yang berbentuk gambar 3D. Tujuan dari penelitian ini adalah untuk menyajikan pendekatan berbasis gesture untuk mengendalikan dan memanipulasi benda-benda dari jarak jauh *virtual* dalam lingkungan realitas maya melalui sensor *Leap Motion*. Aplikasi ini dapat menjadi alternatif untuk memainkan alat musik marawis. Dari hasil uji coba yang dilakukan menunjukkan bahwa aplikasi ini dapat digunakan untuk memainkan marawis secara digital menggunakan Leap Motion.

2.2 Landasan Teori

Teori pendukung pada penelitian yang akan dilakukan diantaranya tentang pembuatan aplikasi piano digital menggunakan sensor Leap Motion yaitu *Leap Motion, Java, Piano, NetBeans, Aplikasi*.

2.2.1 Leap Motion



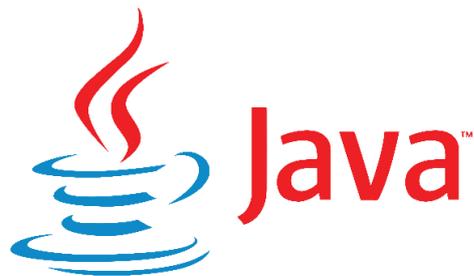
Gambar 2.1 Leap Motion
(Sumber: ultraleap.com)

Menurut Redondo Fernando dkk (2017) Leap Motion adalah sebuah perangkat yang memungkinkan manusia untuk melakukan masukan sentuhan dengan menggunakan tangan dan jari atau menggunakan benda yang menyerupai jari misalnya pensil atau pulpen. *Leap Motion* dapat memberikan alternative dalam pemanfaatan gerakan tangan sebagai interaksi yang bersifat alami antara manusia dan komputer. *Leap Motion* merupakan istilah untuk perekaman gerakan tangan yang digunakan menjadi model digital, yang dapat digunakan untuk menggantikan fungsi mouse dan keyboard. *Leap Motion SDK* adalah sebuah pustaka (*library*) yang dibuat oleh *Leap Motion Inc.* untuk pengembangan aplikasi perangkat lunak yang menggunakan *Leap Motion* sebagai alat masukan utamanya. *Leap Motion SDK* ini ditulis dalam banyak bahasa yaitu *Python, C#, C++, Java Script, ObjectiveC* dan *Java*.

2.2.2 Java

Java adalah bahasa pemrograman yang multi platform dan multi device. Sekali anda menuliskan sebuah program dengan menggunakan *Java*, anda dapat menjalankan hampir di semua computer dan perangkat lain yang mensupport java, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya.

Aplikasi dengan berbasis *Java* ini dikompilasikan ke dalam p-code dan bisa dijalankan dengan *java virtual machine*. Fungsionalitas dari java ini dapat berjalan dengan platform system operasi yang berbeda karena sifatnya yang umum dan non-spesifik (Maria dkk 2015).



Gambar 2.2 Java

(Sumber : logos-download.com)

2.2.3 Piano

Piano adalah alat musik tuts sebagai instrumen dawai dan perkusi yang dimainkan dengan menekan tuts-tuts pada papan piano. Setiap tuts terhubung ke palu yang ada didalam piano dan menekan senar didalamnya sehingga menghasilkan bunyi. Setiap senar memiliki panjang yang berbeda oleh sebab itu piano menghasilkan suara yang berbeda.



Gambar 2.3 Piano

(Sumber: [Wikipedia](https://en.wikipedia.org/wiki/Piano))

Menurut Hirza Herna (2018) piano dikatakan juga alat musik serba bisa, dikatakan serba bisa karena piano mampu memainkan melodi-melodi yang sangat bervariasi dan beragam, piano juga bisa digunakan sebagai pengiring lagu. Piano bisa bermain secara solo (tunggal) dan juga bisa bermain secara kelompok. Piano

juga digunakan dalam suatu orkestra. Piano mampu menghasilkan suara yang rendah (*bass*), dan juga mampu menghasilkan suara yang sangat tinggi, hingga terkesan melengking tinggi.

2.2.4 NetBeans

NetBeans adalah *software* pengembangan perangkat lunak yang ditulis dalam bahasa pemrograman *Java*. Pada *NetBeans*, pengembangan suatu aplikasi dapat dilakukan dimulai dari setelan perangkat lunak modular bernama *modules*. Semula, aplikasi *NetBeans IDE* ini diperuntukkan bagi pengembangan dalam *Java*.



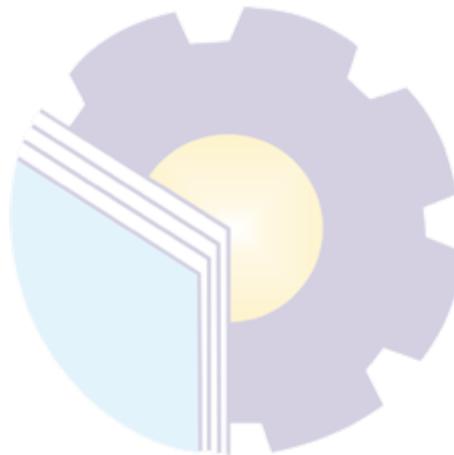
Gambar 2.4 NetBeans IDE

(Sumber:)

Netbeans adalah sebuah aplikasi *Integrated Development Environment* (IDE) yang berbasis *java* dari *sun microsystem* yang berjalan di atas *swing*. *Swing* merupakan teknologi *java* untuk pengembangan aplikasi dekstop yang dapat berjalan pada berbagai macam platform seperti *windows*, *linux*, *Mac OS X*, dan *solaris*. Sebuah IDE merupakan lingkup pemrograman yang diintegrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan *Graphic User Interface* (GUI), suatu editor atau teks, suatu compiler dan suatu *debugger*. *Netbeans* juga digunakan programmer untuk menulis, *mengcompile*, mencari kesalahan, dan menyebarkan program *netbeans* yang ditulis dalam bahasa pemrograman *java*. *Netbeans* merupakan sebuah proyek kode terbuka yang sukses. *Sun mycrosystem* mendirikan proyek kode terbuka *netbeans* pada bulan Juni 2000 dan terus menjadi sponsor utama. Dan saat ini pun *netbeans* menjadi dua produk yaitu platform *NetBeans* dan *NetBeans IDE*. (Rohensih dan Suwarni, 2015).

2.2.5 Aplikasi

Aplikasi adalah Program siap pakai yang dapat digunakan untuk untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembutan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yng diharapkan.



BAB III

PERANCANGAN

3.1 Bahan dan Alat Penelitian

Dalam penelitian ini membutuhkan beberapa bahan dan alat yang digunakan sebagai alat pendukung pembuatan aplikasi.

3.1.1 Bahan Penelitian

Pada aplikasi ini, bahan yang dibutuhkan pada penelitian yang akan dilakukan antara lain :

1. Suara dari setiap tuts piano
2. Gambar yang berkaitan dengan piano

3.1.2 Alat Penelitian

Adapun alat yang digunakan untuk penelitian ini terdiri dari Perangkat Keras dan Perangkat Lunak.

2.1.1.1. Perangkat keras

1. Laptop

Digunakan sebagai alat untuk membuat aplikasi dan program serta melakukan pengujian terhadap hasil yang dibuat. Adapun spesifikasi laptop yang digunakan yaitu :

- *RAM* : 8GB
- *Processor* : Core i5-9300H 2.40Ghz
- *Sistem Operasi* : Windows 10 64-bit

2. Leap Motion

Sebagai Sensor untuk mendeteksi posisi titik koordinat tulang tangan dalam ruang 3 dimensi (X, Y, Z).

3. Kabel *USB*

Sebagai alat untuk menghubungkan antara Laptop ke alat sensor *Leap motion*.

3.1.2.2. Perangkat Lunak

1. *NetBeans*

NetBeans merupakan software pembuatan aplikasi yang menggunakan bahasa pemrograman java.

2. *Driver dan Software Leap Motion*

Driver dan Software dibutuhkan untuk menjalankan *Leap Motion* agar terhubung ke perangkat komputer dan data dapat diterima saat data diinputkan melalui hasil gerakan tangan.

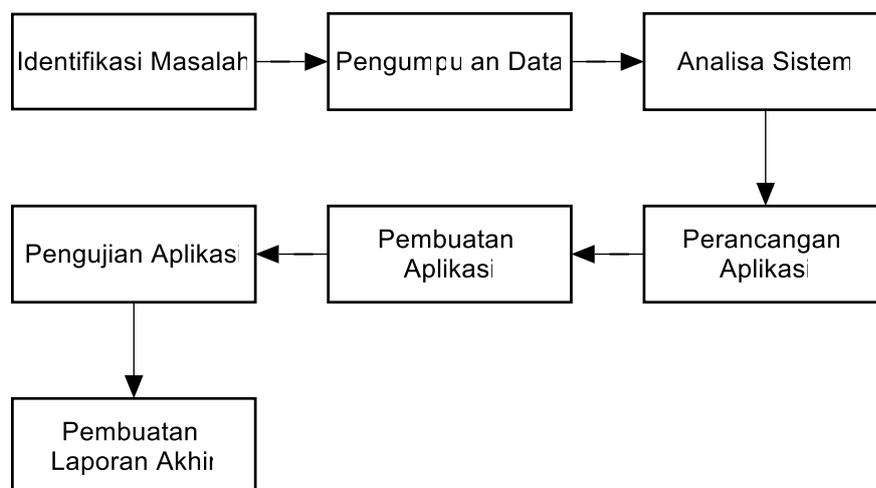
3. *Windows 10*

Windows 10 merupakan system yang digunakan dalam perangkat komputer.

3.2 Prosedur Penelitian

Dalam prosedur penelitian akan dibahas mengenai langkah-langkah penelitian yang akan dilakukan. Adapun langkah-langkah yang dilakukan yaitu identifikasi masalah, pengumpulan data, analisa sistem, perancangan, pembuatan aplikasi, pengujian aplikasi, dan pembuatan laporan akhir. Ditunjukkan pada gambar

3.1



Gambar 3.1 Prosedur Penelitian

(Sumber : data olahan)

3.2.1 Identifikasi Masalah

Identifikasi masalah adalah dengan menentukan masalah yang terjadi dan akan diteliti pada penelitian ini. Setelah mengidentifikasi selanjutnya dilakukan perumusan masalah dan menyelesaikannya.

3.2.2 Pengumpulan Data

Metode pengumpulan data ini yaitu dengan melakukan kajian pada penelitian terdahulu yaitu berupa jurnal yang digunakan untuk membandingkan penelitian terdahulu dengan penelitian yang akan dilakukan.

3.2.3 Analisa Sistem

Analisa sistem dilakukan untuk mengetahui seperti apa gambaran sistem yang akan dibuat pada penelitian yang akan dilakukan yaitu berupa aplikasi piano digital menggunakan sensor *Leap Motion*.

3.2.4 Perancangan Aplikasi

Perancangan aplikasi bertujuan untuk merancang bentuk aplikasi piano digital menggunakan sensor *Leap Motion* akan diterapkan pada penelitian yang akan dilakukan.

3.2.5 Pembuatan Aplikasi

Pada tahap ini dilakukan pembuatan aplikasi menggunakan *NetBeans* sebagai *software* yang digunakan, dan *Leap Motion* sdk sebagai aplikasi bantuan untuk menghubungkan antara *Leap Motion* dengan perangkat komputer.

3.2.6 Pengujian Aplikasi

Setelah aplikasi selesai dibuat, langkah selanjutnya dilakukan pengujian pada beberapa orang karna setiap orang memiliki bentuk tangan yang berbeda. Target pengujian dilakukan minimal 5 pengguna, dan akan menjadi evaluasi apabila ada kekurangan pada aplikasi tersebut.

3.2.7 Pembuatan Laporan Akhir

Tahap terakhir dalam penelitian ini yaitu membuat laporan akhir yang didapat dari hasil pengujian. Laporan akhir akan ditulis sesuai dengan apa yang terjadi pada penelitian tersebut. Adapun format dalam penulisan laporan penelitian tersebut, penulis mengacu pada Panduan Penyusunan Tugas Akhir dan Skripsi Program Diploma Tiga dan Sarjana Terapan Kementerian Riset, Teknologi Dan Pendidikan Tinggi Politeknik Negeri Bengkalis tahun 2017.

3.3 Perancangan Aplikasi

3.3.1 Analisa Cara Bermain Piano

Dalam tahap ini akan menjelaskan mengenai bagaimana system saat ini yang sedang berjalan dalam memainkan alat musik piano. Dalam system yang sedang berjalan saat ini pengguna memainkan alat musik piano harus menekan tuts agar mengeluarkan suara, pada setiap tuts tersebut mengeluarkan suara yang berbeda-beda, nada suara yang dihasilkan berupa tangga nada.

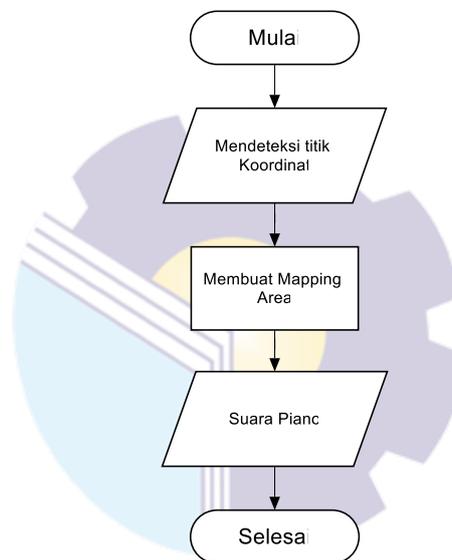
3.3.2 Analisa Aplikasi Piano Yang Diusulkan

Pada tahap ini akan dijelaskan tentang perancangan aplikasi piano digital menggunakan sensor *Leap Motion*. Merancang aplikasi ini sangat diperlukan untuk dapat memahami bagaimana bentuk kerja dari aplikasi tersebut. Pada perancangan aplikasi pengguna dapat memainkan marawis hanya dengan laptop ataupun komputer yang sudah memiliki aplikasi Piano digital sensor *Leap Motion*. Yang akan memudahkan pengguna dalam memainkan alat musik Piano, dimana pengguna hanya perlu menggunakan sensor *Leap Motion* yang terhubung ke komputer. Untuk menghasilkan suara dilakukan dengan menentukan titik gerakan tangan dan posisi tangan berada pada sumbu tertentu sehingga menghasilkan suara yang berbeda-beda yang akan direspon oleh *Leap Motion* secara *real-time*.

3.4 Perancangan Sistem

Perancangan sistem pada penelitian ini sangat diperlukan untuk memudahkan pemahaman cara penggunaan aplikasi yang diusulkan menggunakan rancangan dengan susunan sistem yang diusulkan.

Perancangan Sistem adalah merancang suatu sistem yang baik, yang isinya adalah langkah-langkah dalam proses pembuatan aplikasi yang diusulkan. Untuk dapat menjelaskan rancangan dalam pembuatan Aplikasi Piano Digital Menggunakan Sensor *Leap Motion* dapat dilihat pada gambaran sistem pada Gambar 3.3.



Gambar 3.2 Perancangan Sistem

(Sumber: Data Olahan)

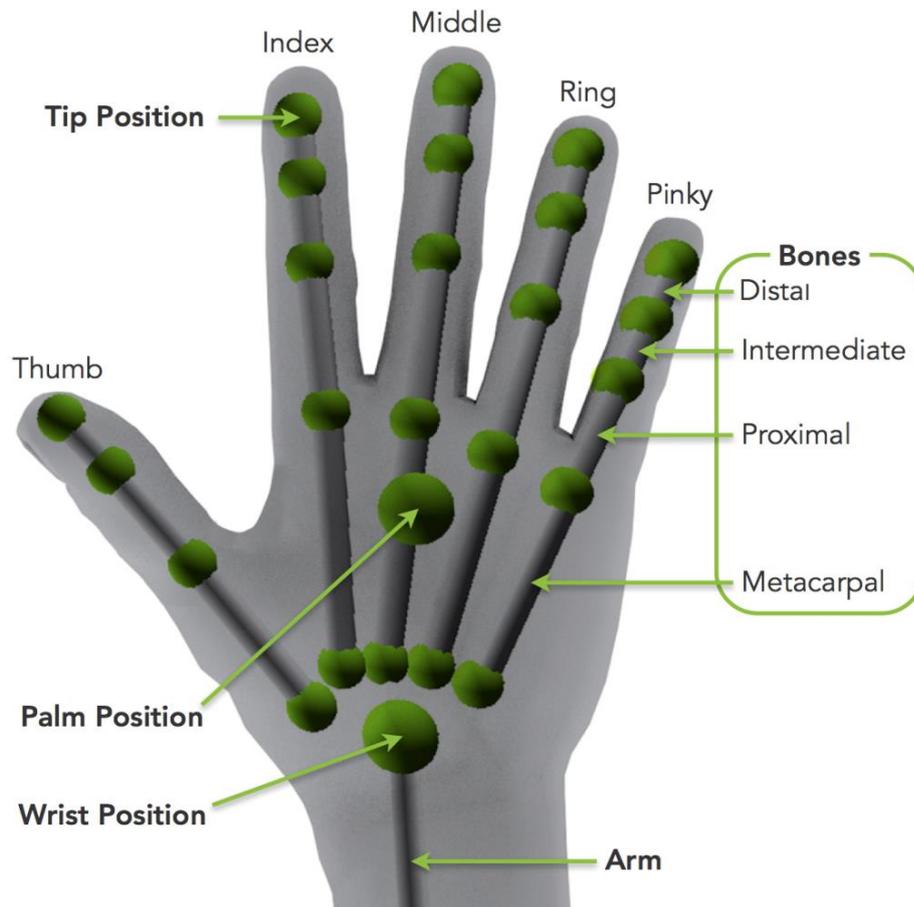
Adapun penjelasan dari perancangan sistem tersebut sebagai berikut:

1. Mulai

Pada tahap awal ini pengguna dapat membuka aplikasi piano digital yang telah dipersiapkan untuk mulai memainkan alat musik piano.

2. Deteksi Titik Koordinat Tangan

Setelah aplikasi dijalankan akan ada penentuan titik koordinat tangan, jari dan gerakan tangan menggunakan sensor *Leap Motion* untuk memainkan piano digital tersebut.



Gambar 3.3 Potional Tracking

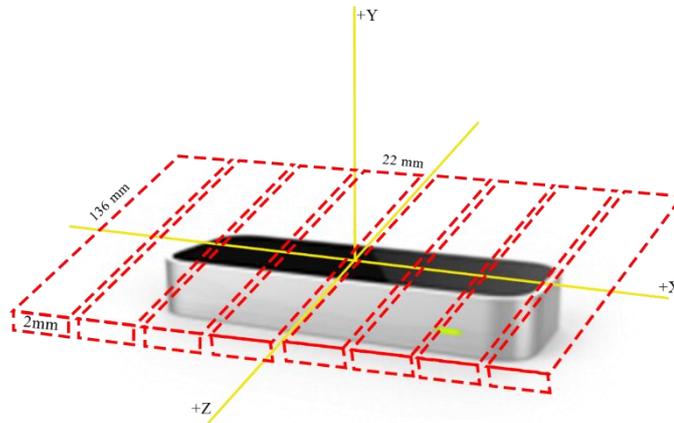
(Sumber: <https://blog.leapmotion.com>)

Pada gambar 3.4 merupakan titik-titik koordinat yang mana ketika tangan digerakan titik-titik inilah yang akan dibaca oleh sensor *Leap Motion* secara real time. Hal ini memungkinkan *Leap Motion* mengenali bentuk tangan seperti telapak tangan, panjang jari, lebar jari, dan juga mengenali yang bukan tangan, sekaligus untuk mendapatkan sumbu X,Y dan Z pada tangan.

3. Mapping Area

Proses mapping area ini menentukan *Threshold* (batasan) untuk menghasilkan suara yang berbeda dengan jarak yang diinginkan, dari setiap

area akan menghasilkan 8 suara dari aplikasi piano digital tersebut melalui titik koordinat yang telah ditentukan.



Gambar 3.4 Penentuan Area

(Sumber : www.leapmotion.com)

Pada gambar 3.5 menjelaskan mengenai 8 area yang berbeda yang memiliki koordinat yang berbeda-beda, seperti:

Area 1 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq -73 \ \& \ X \geq -95 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 2 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq -49 \ \& \ X \geq -71 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 3 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq -25 \ \& \ X \geq -47 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 4 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq -2 \ \& \ X \geq -23 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 5 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq 2 \ \& \ X \geq 23 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 6 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq 25 \ \& \ X \geq 47 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 7 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq 49 \ \& \ X \geq 71 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Area 8 : $(Y \geq 200 \ \& \ Y \leq 210 \ \& \ X \leq 73 \ \& \ X \geq 95 \ \& \ Z \leq 68 \ \& \ Z \geq -68)$

Apabila posisi tangan tidak tepat berada di area yang ditentukan maka aplikasi tidak akan merespon apapun karna suara yang dihasilkan sudah ditentukan posisi dan jarak pada sensor *Leap Motion*.

4. Suara

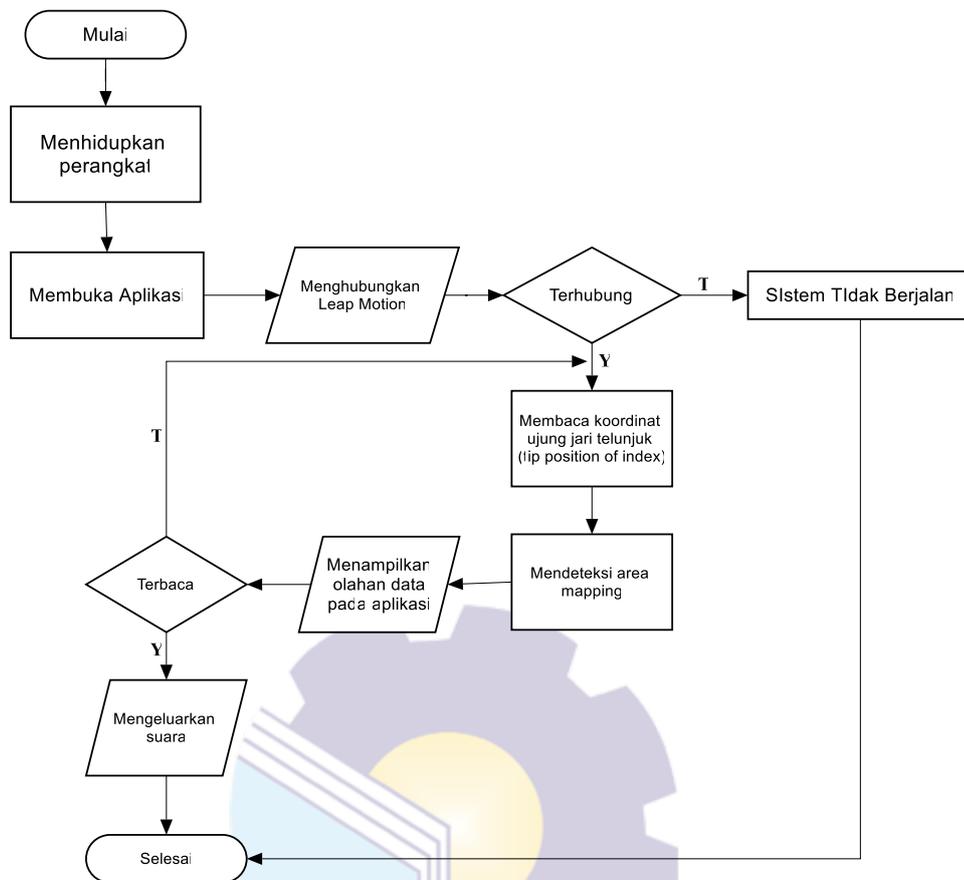
Suara yang dihasilkan merupakan inputan dari aplikasi tersebut, suara akan muncul apabila mapping area yang ditetapkan tersentuh dan dapat dibaca oleh sensor *Leap Motion*, dan pada setiap area memiliki suara yang berbeda-beda yaitu berupa tangga nada.

5. Selesai

Setelah semua berjalan dengan baik maka aplikasi dapat digunakan dan dapat keluar dari aplikasi setelah memainkannya.

3.4.1 Rancangan Diagram Alir Sistem Secara Umum

Pada gambar 3.6 adalah gambar alir sistem penggunaan aplikasi piano digital secara umum. Pada diagram ini menjelaskan mengenai proses sistem yang berjalan dari awal hingga akhir. Diagram ini diawali dari menghidupkan perangkat komputer, lalu membuka aplikasi piano, lalu dihubungkan ke sensor Leap Motion dengan menggunakan kabel USB, secara otomatis *Leap Motion* dapat digunakan melalui perangkat komputer. Kemudian setelah itu *Leap Motion* akan membaca gerakan tangan secara real time dan mendeteksi mapping area lalu menampilkan olahan data melalui aplikasi. Ketika proses pengolahan data berhasil maka aplikasi akan mengeluarkan suara sesuai area yang telah ditentukan dan aplikasi berjalan semestinya.

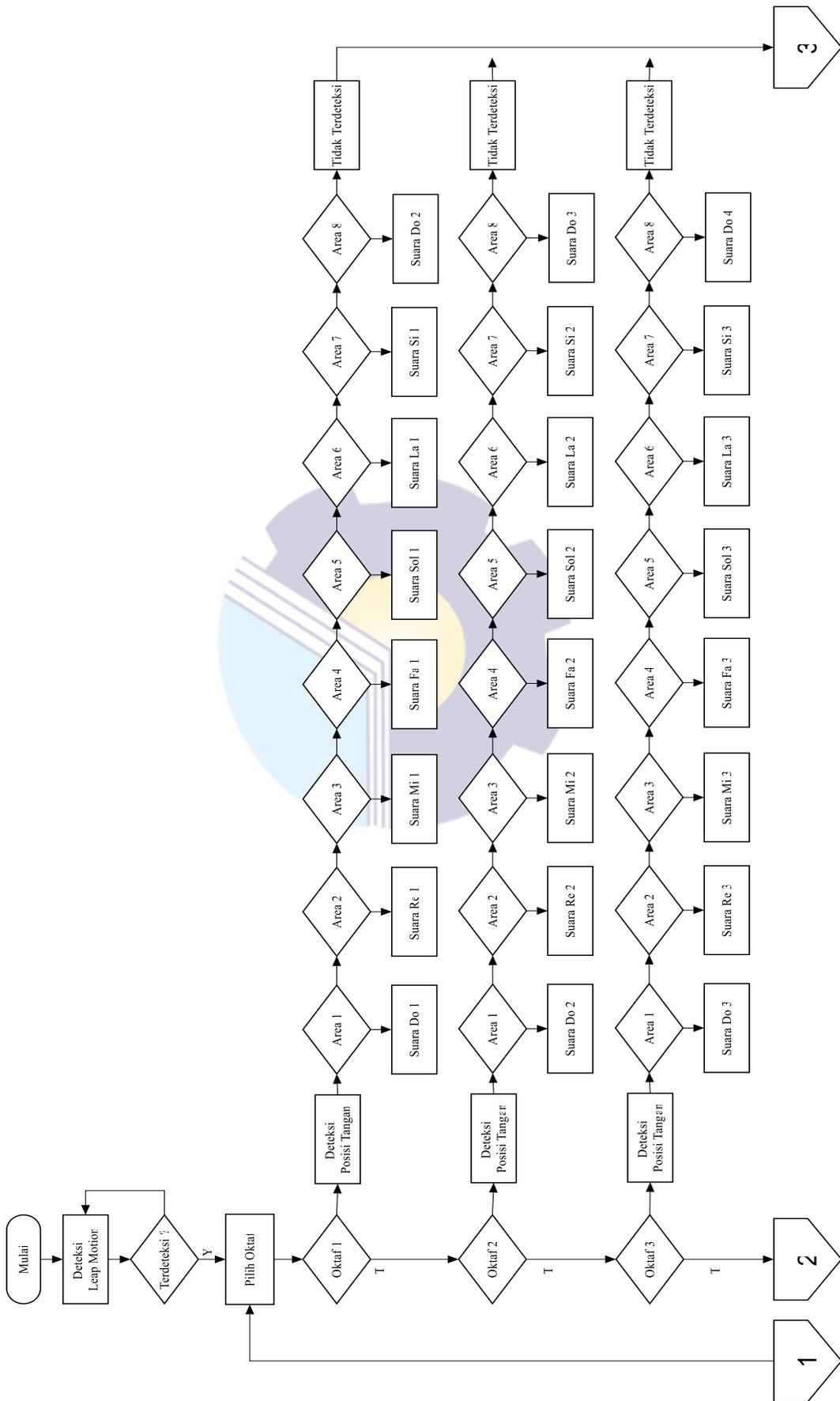


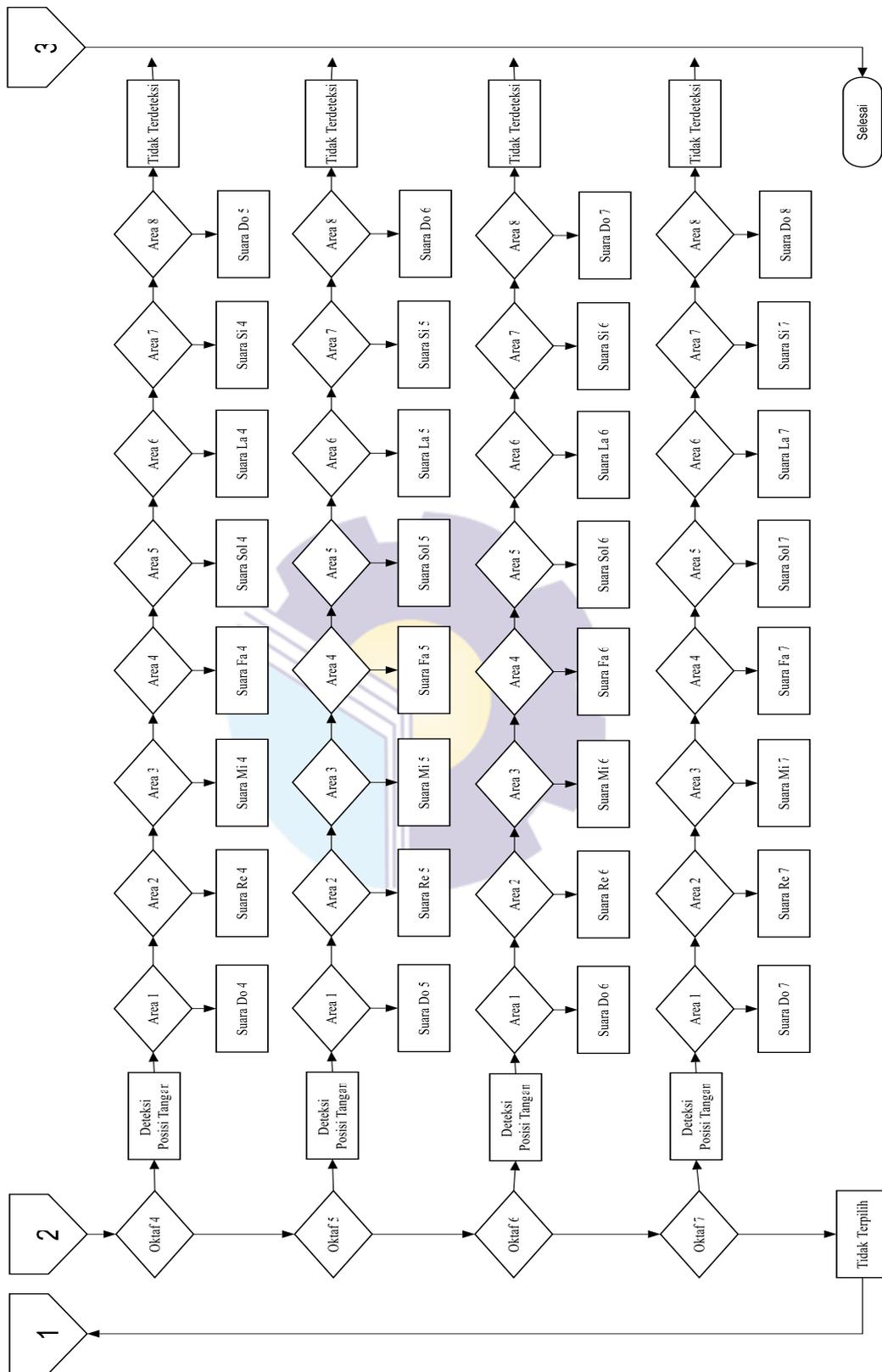
Gambar 3.5 Rancangan Diagram Alir Sistem Secara Umum

(Sumber : data olahan)

3.4.2 Rancangan Diagram Output Suara

Pada gambar 3.6 merupakan diagram untuk output suara yang dihasilkan aplikasi piano digital. Pada diagram ini menjelaskan mengenai proses sistem *output* suara dari awal hingga akhir.





Gambar 3.6 Rancangan output suara
(sumber : data olahan)

Proses perancangan output suara dimulai dari mendeteksi *Leap Motion*, jika *Leap Motion* terdeteksi maka sistem akan lanjut ke pemilihan oktaf, jika tidak maka sistem akan kembali mendeteksi *Leap Motion*. Pada pemilihan oktaf terdapat 7 pilihan oktaf yang disediakan yaitu oktaf 1 hingga oktaf 7, untuk pemilihan oktaf akan dijelaskan sebagai berikut :

1. Jika terpilih oktaf 1, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do 1), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 1), apabila tangan berada di posisi area 3 maka akan mengeluarkan suara 3 (Mi 1), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 1), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 1), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 1), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 1), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 2).
2. Jika terpilih oktaf 2, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do 2), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 2), apabila tangan berada di posisi area 3 maka akan mengeluarkan suara 3 (Mi 2), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 2), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 2), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 2), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 2), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 3).
3. Jika terpilih oktaf 3, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do 3), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 3), apabila tangan berada di posisi area 3 maka akan mengeluarkan

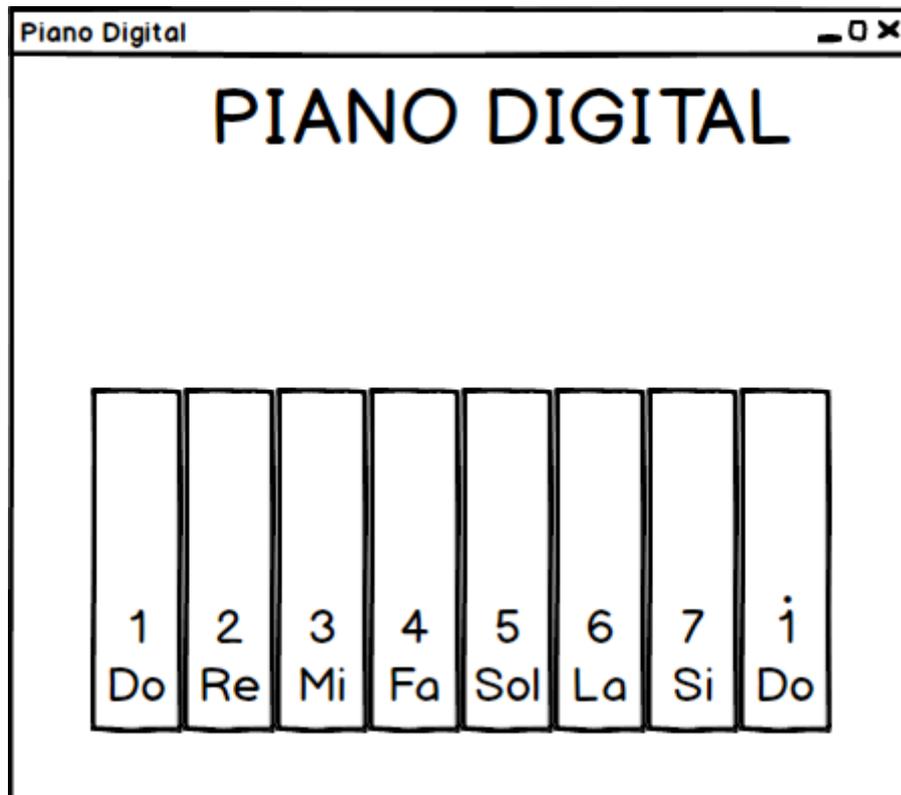
suara 3 (Mi 3), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 3), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 3), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 3), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 3), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 4).

4. Jika terpilih oktaf 4, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do 4), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 4), apabila tangan berada di posisi area 3 maka akan mengeluarkan suara 3 (Mi 4), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 4), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 4), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 4), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 4), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 5).
5. Jika terpilih oktaf 5, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do 5), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 5), apabila tangan berada di posisi area 3 maka akan mengeluarkan suara 3 (Mi 5), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 5), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 5), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 5), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 5), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 6).
6. Jika terpilih oktaf 6, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do

- 6), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 6), apabila tangan berada di posisi area 3 maka akan mengeluarkan suara 3 (Mi 6), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 6), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 6), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 6), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 6), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 7).
7. Jika terpilih oktaf 7, sensor Leap Motion akan mendeteksi posisi tangan , apabila tangan berada di posisi area 1 maka akan mengeluarkan suara 1 (Do 7), apabila tangan berada di posisi area 2 maka akan mengeluarkan suara 2 (Re 7), apabila tangan berada di posisi area 3 maka akan mengeluarkan suara 3 (Mi 7), apabila tangan berada di posisi area 4 maka akan mengeluarkan suara 4 (Fa 7), apabila tangan berada di posisi area 5 maka akan mengeluarkan suara 5 (Sol 7), apabila tangan berada di posisi area 6 maka akan mengeluarkan suara 6 (La 7), apabila tangan berada di posisi area 7 maka akan mengeluarkan suara 7 (Si 7), apabila tangan berada di posisi area 8 maka akan mengeluarkan suara 8 (Do 8).

3.4.3 Rancangan Antar Muka

Pada tampilan antarmuka aplikasi piano digital cukup sederhana, karna pada aplikasi ini hanya menampilkan betuk dari piano tersebut karna pada pembuatan hanya mengutamakan fungsi dan kegunaan dari sensor *Leap Motion*.



Gambar 3.7 Perancangan Antarmuka

(sumber : data olahan)

Pada gambar 3.7 tuts piano akan berubah warna apabila sensor mendeteksi keberadaan jari pada area-area yang telah ditentukan. jadi bentuk proses dari aplikasi tersebut sebagai berikut :

1. Apabila area 1 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 1 berbunyi pada saat yang bersamaan.
2. Apabila area 2 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 2 berbunyi pada saat yang bersamaan,
3. Apabila area 3 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 3 berbunyi pada saat yang bersamaan,
4. Apabila area 4 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 4 berbunyi pada saat yang bersamaan,
5. Apabila area 5 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 5 berbunyi pada saat yang bersamaan,

6. Apabila area 6 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 6 berbunyi pada saat yang bersamaan,
7. Apabila area 7 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 7 berbunyi pada saat yang bersamaan,
8. Apabila area 8 berubah warna berarti sensor mendeteksi keberadaan tangan lalu suara 8 berbunyi pada saat yang bersamaan.



BAB IV

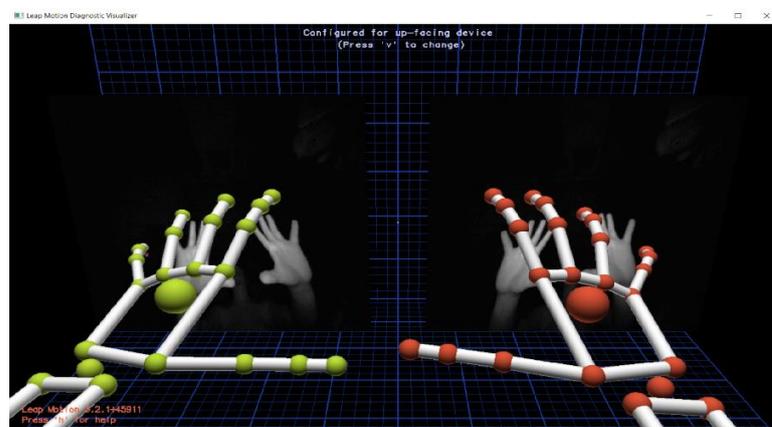
HASIL DAN PENGUJIAN

4.1. Hasil Implementasi

Hasil dari implementasi ini menghasilkan aplikasi Piano Digital dengan menggunakan Sensor *Leap Motion*. Sensor *Leap Motion* dapat memudahkan untuk bermain Piano Digital, karena memainkannya hanya menggerakkan jari pada posisi yang telah ditentukan diatas sensor *Leap Motion*. Aplikasi ini dibuat menggunakan *software NetBeans* untuk mendesain alat musik piano.

4.1.1. Visualizer Leap Motion

Visualizer Leap Motion menampilkan berbagai data pelacakan yang disediakan oleh *Leap Motion API* dan merupakan cara yang baik untuk merasakan data yang dihasilkan oleh *Leap*. Mulai *Visualizer* dari halaman *Troubleshooting* pada dialog *Leap Motion Settings* dengan mengklik tombol *Diagnostic Visualizer*. Setiap gerakan akan terus di tampilkan secara *real time*, dari sinilah data yang akan diperoleh saat menggunakan sensor *Leap Motion* data-data tersebut akan di ambil melalui titik-titik jari pada manusia. Berikut ini adalah tampilan gambar dari *Visualizer Leap Motion*.



Gambar 4.1 Visualizer Leap Motion

(Sumber : Data Olahan)

4.1.2. Tampilan Icon Software NetBeans

Pada bagaian ini adalah permulaan untuk menjalan aplikasi piano digital karna aplikasi piano di buat menggunakan *software NetBeans*. Berikut tampilan gambar awal untuk membuka *software NetBeans*.



Gambar 4.2 Netbeasn IDE 8.0.2

(Sumber : Data Olahan)

Software NetBeans yang digunakan pada aplikasi ini adalah NetBeans.IDE Versi 8.0.2. *Software NetBeans* ini dapat berjalan pada berbagai macam platform seperti *windows, linux, Mac OS X* dan *Solaris*. Selain menggunakan *NetBeans* masih ada aplikasi yang dapat digunakan untuk membuat aplikasi marawis digital yaitu *VisualBasic* hanya saja menggunakan bahasa C.

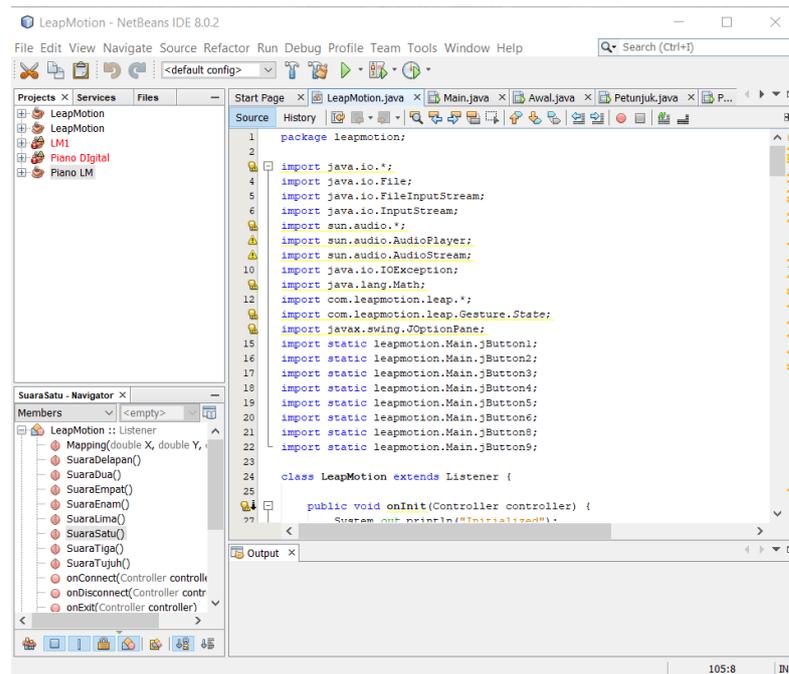
4.1.3. SDK Leap Motion LeapDeveloperKit_3.2.1+45911_win

Untuk menghubungkan antara *Leap Motion* dan perangkat komputer yang menggunakan aplikasi pemrograman *Java* yaitu NetBeans IDE, di dalam komputer atau laptop harus di install *Driver Leap Motion* untuk membaca sensor *Leap Motion* ke dalam komputer atau laptop yang dihubungkan melalui kabel *USB*, sensor ini akan membaca dan mengambil data pada gerakan tangan secara *real time*.

Selain itu penambahan libraries juga harus dilakukan agar perangkat lunak atau software seperti *NetBeans* dapat digunakan pada sensor *Leap Motion* yang terhubung di dalam komputer atau laptop, masukkan *SDK Leap Motion* di dalam libraries *NetBeans* untuk penghubung dari project ke sensor *Leap Motion*.

4.1.4. Tampilan Dari *Software Netbeans*

Setelah *NetBeans* terbuka aplikasi yang sudah dirancang dan di program di *software Netbeans* sudah bisa langsung dijalankan di dalam *software NetBeans*. Berikut gambar tampilan dari *software NetBeans* untuk menjalankan aplikasi yang telah di buat.



Gambar 4.3 Tampilan Dari Netbeans

(Sumber : Data Olahan)

Dalam pembuatan aplikasi piano, aplikasi menggunakan package dan beberapa class yang dibutuhkan yang saling berhubungan satu sama lain, berikut class yang saling terhubung :

4.1.4.1. *Class Leap Motion*

Class Leap Motion disini berperan sebagai program utama untuk menggunakan sensor leap motion. Dalam class ini semua program Leap Motion dibuat, mulai dari penetapan titik jari tangan, suara dari aplikasi, mapping area dan posisi X, Y dan Z pada sensor *Leap Motion*. Berikut ini merupakan *source code* dari *class leap motion*.

```
package leapmotion;

import java.io.*;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import sun.audio.*;
import sun.audio.AudioPlayer;
import sun.audio.AudioStream;
import java.io.IOException;
import java.lang.Math;
import com.leapmotion.leap.*;
import com.leapmotion.leap.Gesture.State;
import javax.swing.JOptionPane;
import static leapmotion.Main.jButton1;
import static leapmotion.Main.jButton2;
import static leapmotion.Main.jButton3;
import static leapmotion.Main.jButton4;
import static leapmotion.Main.jButton5;
import static leapmotion.Main.jButton6;
import static leapmotion.Main.jButton8;
import static leapmotion.Main.jButton9;

class LeapMotion extends Listener {

    public void onInit(Controller controller) {
        System.out.println("Initialized");
    }
    double fingerPointX, fingerPointY, fingerPointZ, fingerPoint;

    public void onConnect(Controller controller) {
```

```

System.out.println("Connected");
controller.enableGesture(Gesture.Type.TYPE_SWIPE);
controller.enableGesture(Gesture.Type.TYPE_CIRCLE);
controller.enableGesture(Gesture.Type.TYPE_SCREEN_TAP);
controller.enableGesture(Gesture.Type.TYPE_KEY_TAP);
}

public void onDisconnect(Controller controller) {
    //Note: not dispatched when running in a debugger.
    System.out.println("Disconnected");
}

public void onExit(Controller controller) {
    System.out.println("Exited");
}

public void onFrame(Controller controller) {
    // Get the most recent frame and report some basic information
    Frame frame = controller.frame();

    for (Hand hand : frame.hands()) {

        // Get the hand's normal vector and direction
        // Get finge
        for (Finger finger : hand.fingers()) {
            String handType = hand.isLeft() ? "Left hand" : "Right hand";
            Finger.Type fingerType = finger.type().TYPE_INDEX;
            System.out.println("coba: " + finger.type().TYPE_INDEX);

            //Get Bones
            for (Bone.Type boneType : Bone.Type.values()) {

```

```

        Bone bone = finger.bone(boneType);
        System.out.println("    " + bone.type().TYPE_DISTAL + ",
end: " + bone.nextJoint());

        Vector boneEnd = bone.nextJoint();
        Vector AmbilUjung = bone.nextJoint();

        fingerPointX = AmbilUjung.getX();
        fingerPointY = AmbilUjung.getY();
        fingerPointZ = AmbilUjung.getZ();

    }
}
Mapping(fingerPointX, fingerPointY, fingerPointZ);

System.out.println("Nilai X :" + fingerPointX);
System.out.println("Nilai Y :" + fingerPointY);
System.out.println("Nilai Z :" + fingerPointZ);
System.out.println("\n\n");
}

```

Gambar 4.4 *Source Code Leap Motion*

(Sumber : Data Olahan)

Source code diatas merupakan source code untuk Pengkoneksian Leap Motion dengan aplikasi yang dibuat, dan untuk penentuan posisi X, Y, Dan Z.

```

public static void SuaraSatu() {
    InputStream iAudio;
    if (Octaf == 1) {
        try {
            iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C1.wav"));

```

```

        AudioStream iMusic = new AudioStream(iAudio);
        AudioPlayer.player.start(iMusic);
        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
} else if (Octaf == 2) {
    try {
        iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C2.wav"));
        AudioStream iMusic = new AudioStream(iAudio);
        AudioPlayer.player.start(iMusic);
        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
} else if (Octaf == 3) {
    try {
        iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C3.wav"));
        AudioStream iMusic = new AudioStream(iAudio);
        AudioPlayer.player.start(iMusic);
        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
} else if (Octaf == 4) {
    try {
        iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C4.wav"));
        AudioStream iMusic = new AudioStream(iAudio);

```

```

        AudioPlayer.player.start(iMusic);
        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
} else if (Octaf == 5) {
    try {
        iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C5.wav"));
        AudioStream iMusic = new AudioStream(iAudio);
        AudioPlayer.player.start(iMusic);
        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
} else if (Octaf == 6) {
    try {
        iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C6.wav"));
        AudioStream iMusic = new AudioStream(iAudio);
        AudioPlayer.player.start(iMusic);
        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
} else if (Octaf == 7) {
    try {
        iAudio = new FileInputStream(new
File("D:\\LeapMotion\\Suara\\C7.wav"));
        AudioStream iMusic = new AudioStream(iAudio);
        AudioPlayer.player.start(iMusic);

```

```

        Thread.sleep(1000);

    } catch (IOException | InterruptedException e) {
    }
}
}

```

Gambar 4.5 Source Code Suara

(Sumber : Data Olahan)

Source code diatas merupakan source code menginputkan suara pada aplikasi yang dibuat.

```

public static void Mapping(double X, double Y, double Z) {
    if (Y >= 200 & X <= -73 & X >= -95 & Z <= 68 & Z >= -68) {
        SuaraSatu();
        jButton1.setBackground(new java.awt.Color(153, 153, 153));
        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

    } else if (Y >= 200 & X <= -49 & X >= -71 & Z <= 68 & Z >= -
68) {
        SuaraDua();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));
        jButton2.setBackground(new java.awt.Color(153, 153, 153));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
    }
}

```

```

        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

    } else if (Y >= 200 & X <= -25 & X >= -47 & Z <= 68 & Z >= -
68) {
        SuaraTiga();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));
        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(153, 153, 153));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

    } else if (Y >= 200 & X <= -2 & X >= -23 & Z <= 68 & Z >= -68)
    {
        SuaraEmpat();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));
        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(153, 153, 153));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

    } else if (Y >= 200 & X >= 2 & X <= 23 & Z <= 68 & Z >= -68) {
        SuaraLima();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));

```

```

        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(153, 153, 153));
        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

    } else if (Y >= 200 & X >= 25 & X <= 47 & Z <= 68 & Z >= -68)
    {

        SuaraEnam();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));
        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
        jButton6.setBackground(new java.awt.Color(153, 153, 153));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

    } else if (Y >= 200 & X >= 49 & X <= 71 & Z <= 68 & Z >= -68)
    {

        SuaraTujuh();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));
        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(153, 153, 153));
        jButton9.setBackground(new java.awt.Color(255, 255, 255));

```

```

    } else if (Y >= 200 & X >= 73 & X <= 95 & Z <= 68 & Z >= -68)
    {
        SuaraDelapan();
        jButton1.setBackground(new java.awt.Color(255, 255, 255));
        jButton2.setBackground(new java.awt.Color(255, 255, 255));
        jButton3.setBackground(new java.awt.Color(255, 255, 255));
        jButton4.setBackground(new java.awt.Color(255, 255, 255));
        jButton5.setBackground(new java.awt.Color(255, 255, 255));
        jButton6.setBackground(new java.awt.Color(255, 255, 255));
        jButton8.setBackground(new java.awt.Color(255, 255, 255));
        jButton9.setBackground(new java.awt.Color(153, 153, 153));
    } else {
    }
}

```

Gambar 4.6 Source Code Mapping Area

(Sumber : Data Olahan)

Source code diatas merupakan *source code* penentuan *mapping area* pada aplikasi yang dibuat.

4.1.4.2. Class Main

Class Main sebagai pembuatan desain dari aplikasi piano dan beberapa bahan yang diperlukan dalam pembuatan desain. Berikut ini merupakan *source code* dari *class Main*.

```

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 1;
}

```

```

private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 2;
}
private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 3;
}
private void jButton12ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 4;
}
private void jButton14ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 6;
}
private void jButton13ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 5;
}
private void jButton15ActionPerformed(java.awt.event.ActionEvent evt) {
    LeapMotion test = new LeapMotion();
    test.Octaf = 7;
}

```

Gambar 4.7 Source Code Class Main

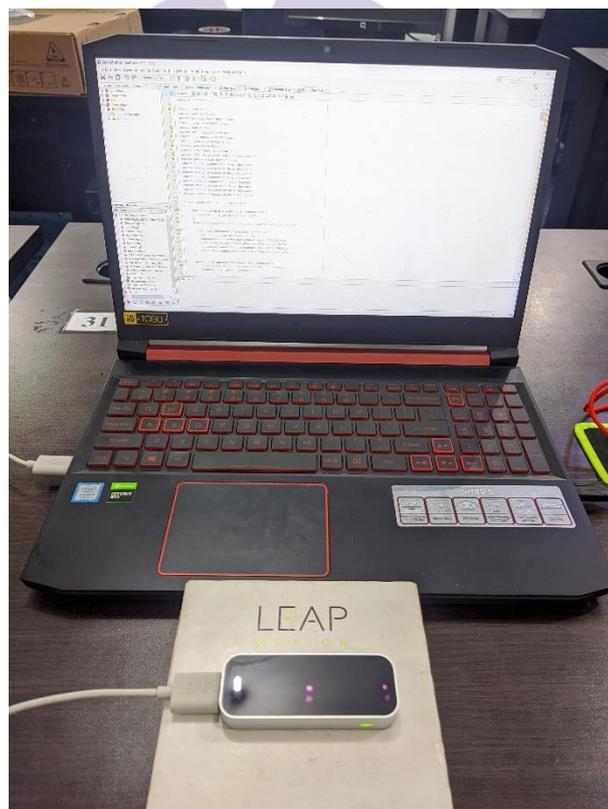
(Sumber : Data Olahan)

Source code diatas merupakan pemanggilan setiap *octaf* suara pada tombol *octaf* yang ada di tampilan main piano

4.1.5. Tampilan Menghubungkan Leap Motion

Pada bagian ini akan menghubungkan antara *Leap Motion* dan perangkat komputer, karna dengan *Leap Motion* data bisa diolah agar aplikasi yang dibuat dapat berjalan sebagaimana mesitinya sesuai dengan perintah yang telah ditetapkan pada pembuatan aplikasi di *software Netbeans*.

Pada setiap perangkat komputer maupun laptop yang akan digunakan untuk memainkan aplikasi piano diharuskan menginstal driver leap motion, agar leap motion bisa beroperasi, setelah di instal hubungkan leap motion ke perangkat komputer, pastikan lampu infrared pada leap motion menyala, karna jika sudah menyala berarti leap motion telah terhubung ke perangkat.

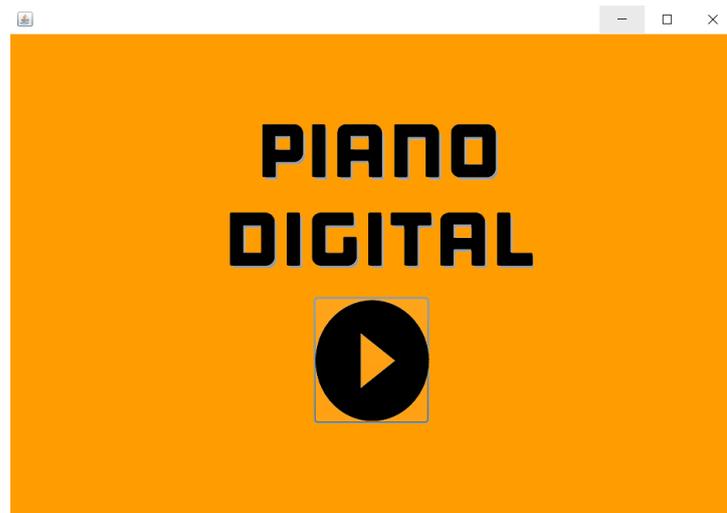


Gambar 4.8 Tampilan menghubungkan *Leap Motion*

(Sumber : Data Olahan)

4.1.6. Tampilan Awal Mulai

Tampilan Awal Mulai merupakan tampilan pertama saat memulai yang menampilkan bagian dari awal aplikasi Piano Digital yang cukup sederhana. Berikut tampilan awal dari aplikasi Piano Digital.



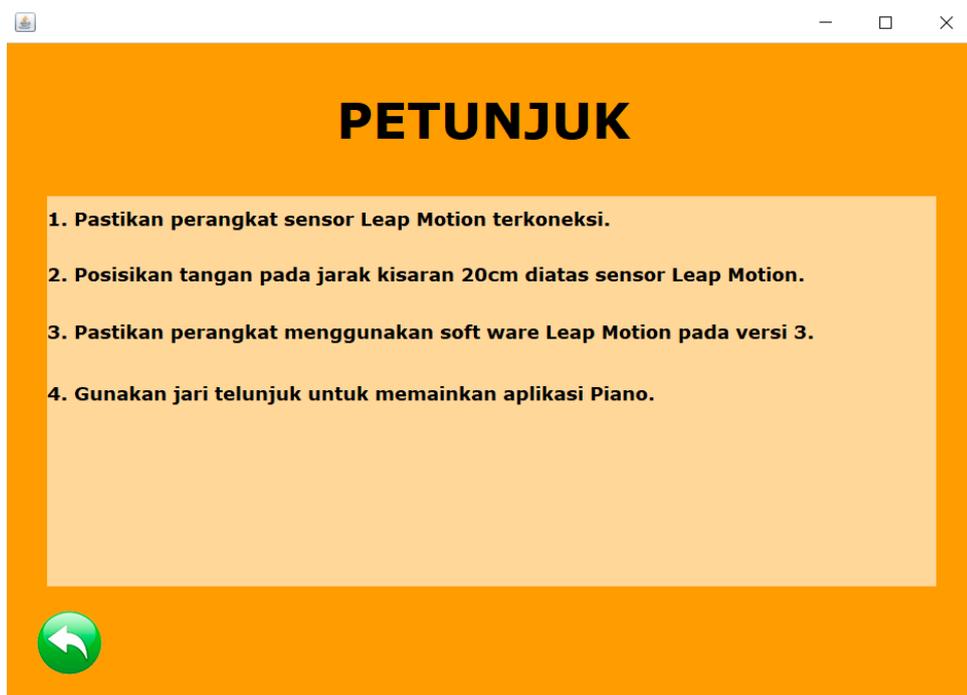
Gambar 4.9 Tampilan Awal

(Sumber : Data Olahan)

Pada tampilan awal tidak memiliki menu apapun selain tombol start untuk memulai aplikasi.

4.1.7. Tampilan Petunjuk

Tampilan ini merupakan tampilan petunjuk cara bermain piano pada aplikasi ini, tampilan ini akan keluar ketika tombol start pada tampilan awal di klik. Berikut tampilan petunjuk dari aplikasi Piano Digital.



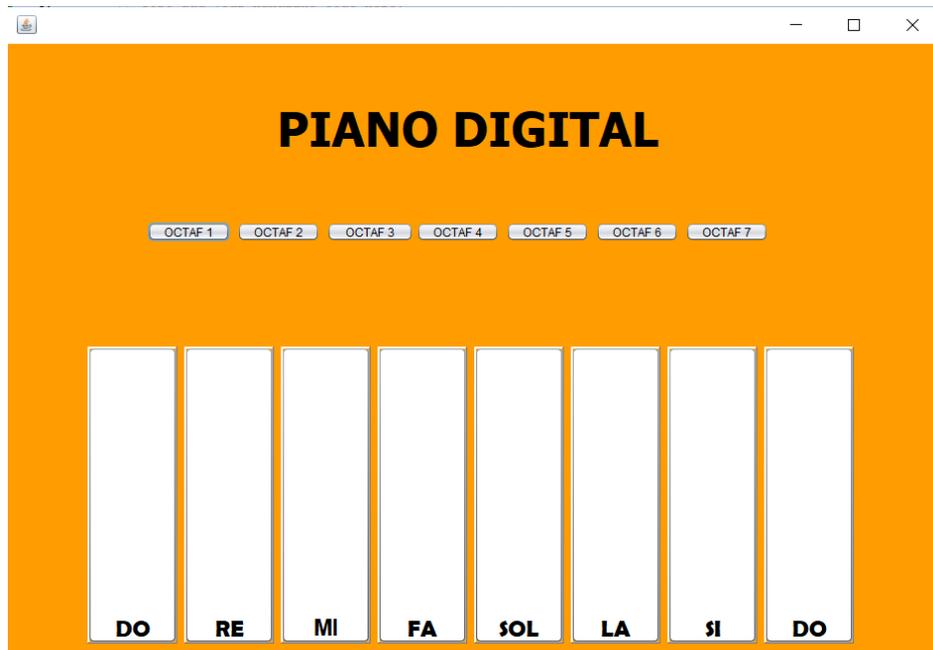
Gambar 4.10 Tampilan Petunjuk

(Sumber : Data Olahan)

Pada tampilan petunjuk tidak memiliki menu apapun selain tombol back yang menuju ke tampilan selanjutnya.

4.1.8. Tampilan *Main* Piano Digital

Tampilan ini merupakan tampilan inti dari aplikasi Piano Digital, aplikasi dapat di mainkan dengan menghubungkan sensor *Leap Motion*. Pada tampilan ini berisikan 8 tuts piano yang berfungsi sebagai tanda bahwa area mapping yang telah ditentukan tersentuh. Berikut tampilan inti dari aplikasi Piano Digital.

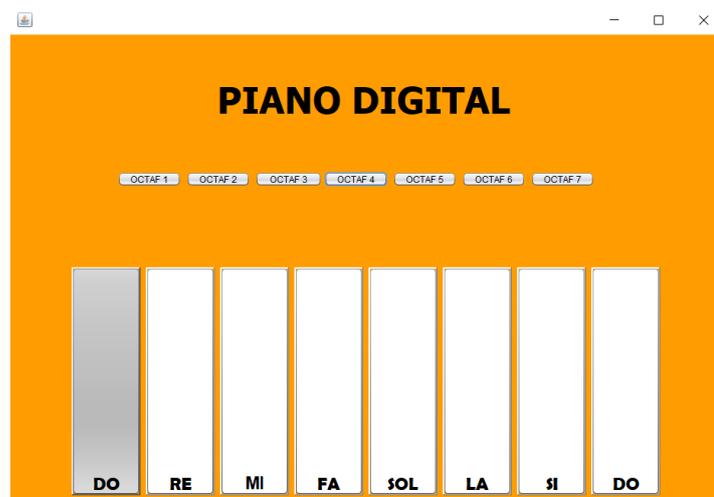


Gambar 4.11 Tampilan *Main*

(Sumber : Data Olahan)

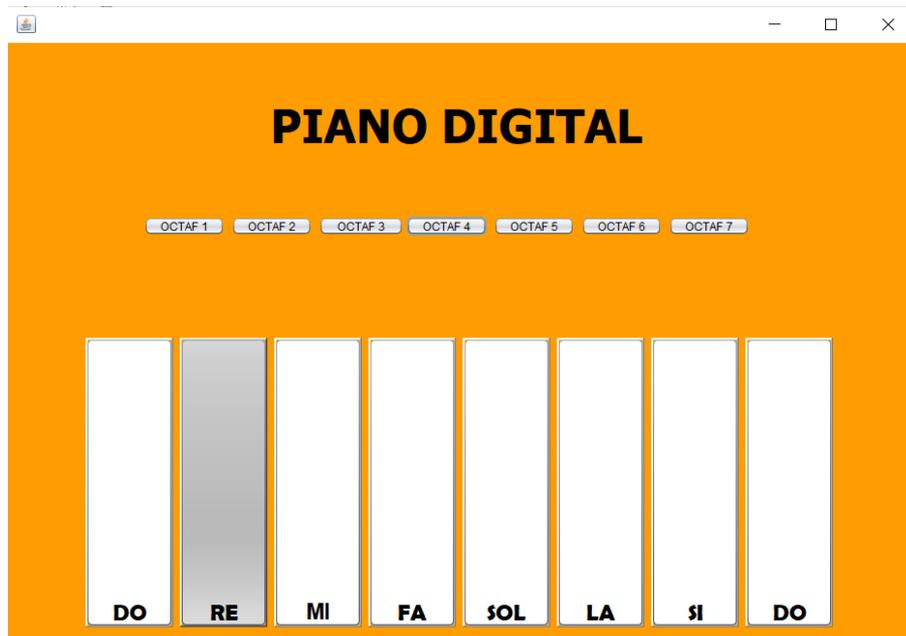
4.1.9. Tampilan *Main* Saat Dijalankan

Pada penjelasan ini akan menunjukkan tampilan dari aplikasi ketika dimainkan. Setiap tuts piano akan berubah warna dan akan mengeluarkan suara apabila area mapping yang di tentukan tersentuh atau terbaca oleh leap motion. Berikut gambaran dari setiap tuts yang terbaca.

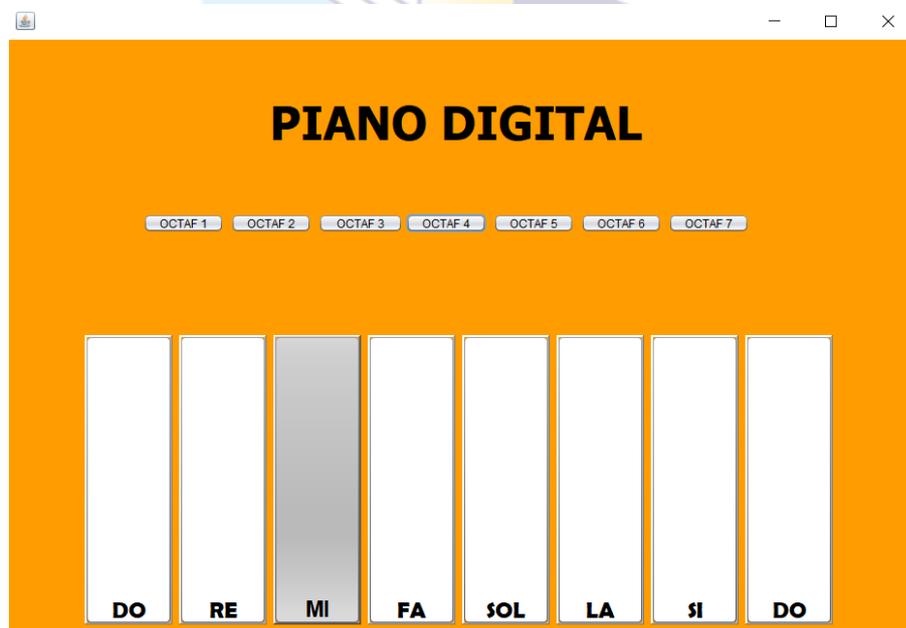


Gambar 4.12 Tampilan area satu terbaca

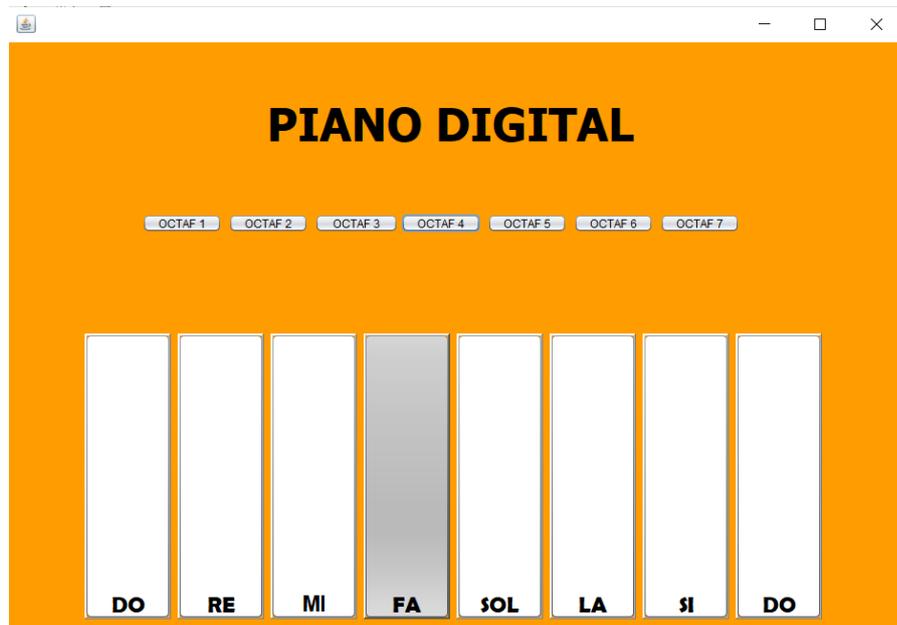
(Sumber : Data Olahan)



Gambar 4.13 Tampilan area dua terbaca
(Sumber : Data Olahan)

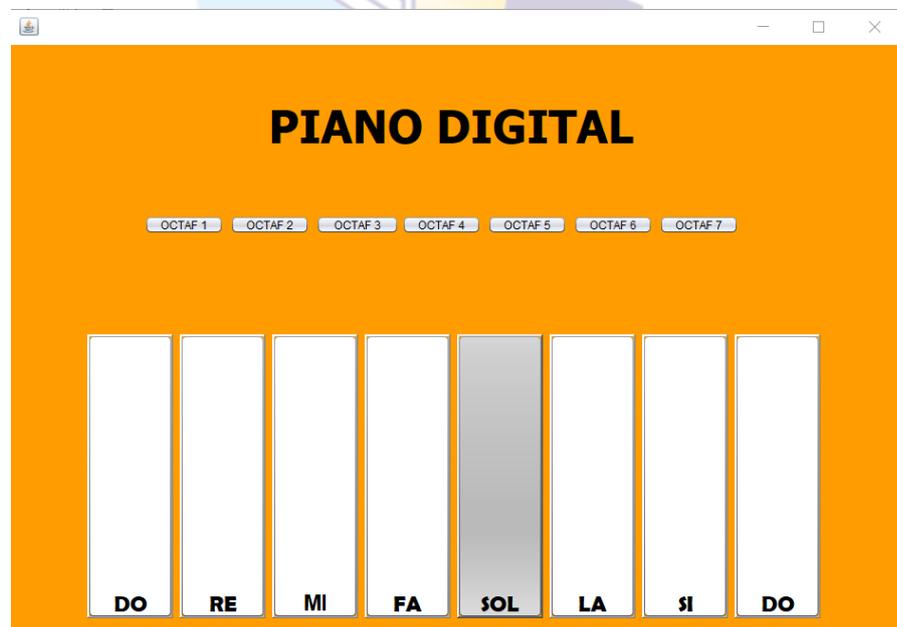


Gambar 4.14 Tampilan area tiga terbaca
(Sumber : Data Olahan)



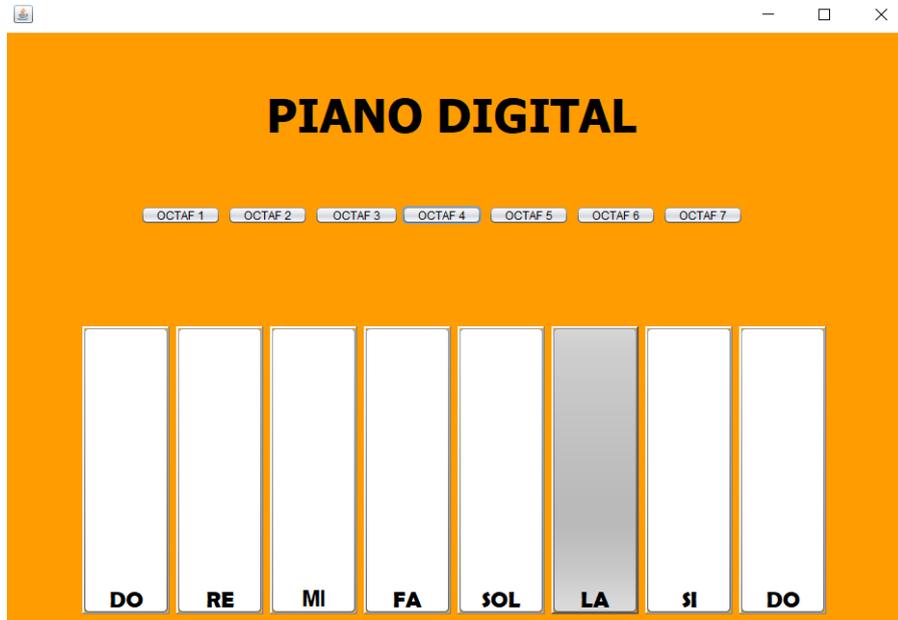
Gambar 4.15 Tampilan area empat terbaca

(Sumber : Data Olahan)

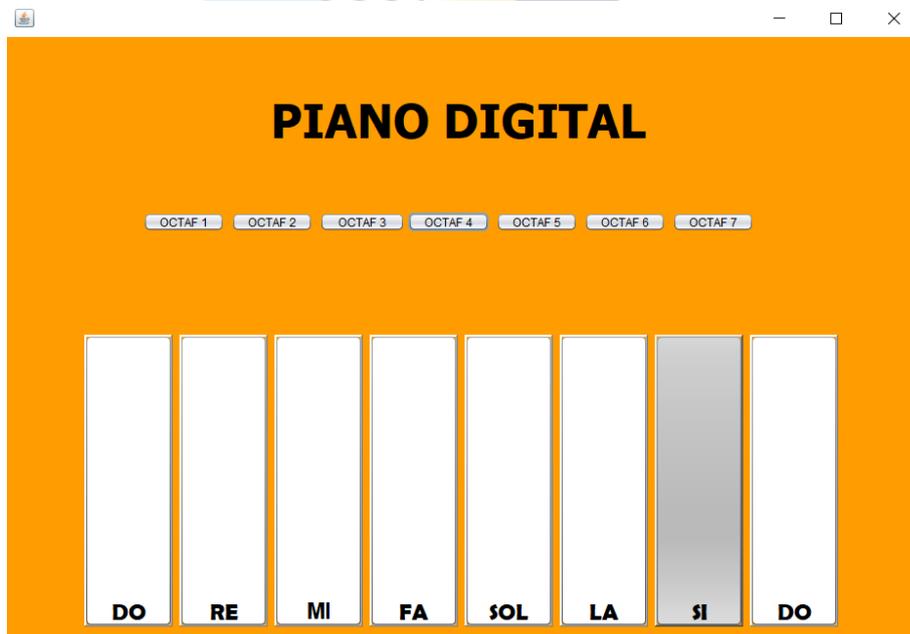


Gambar 4.16 Tampilan area lima terbaca

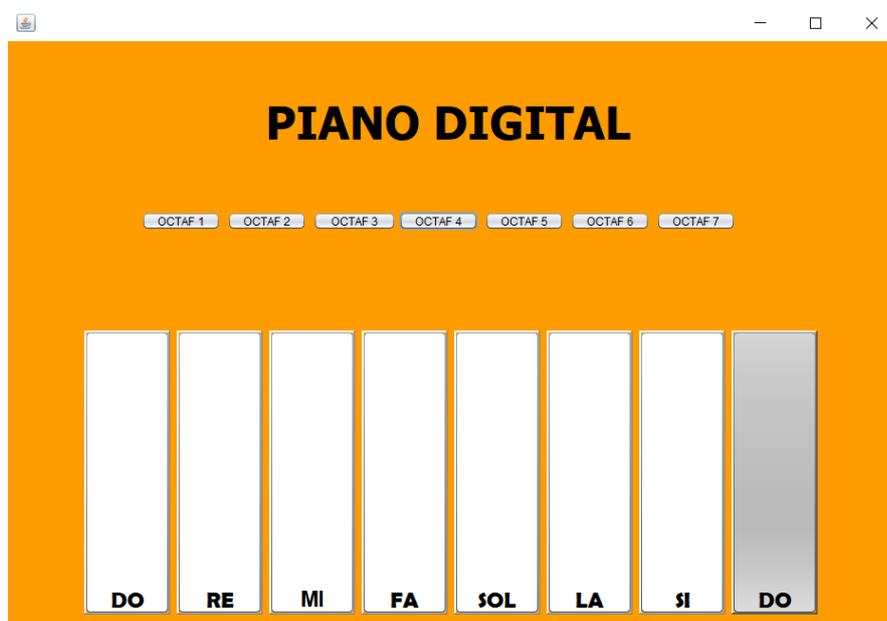
(Sumber : Data Olahan)



Gambar 4.17 Tampilan area enam terbaca
(Sumber : Data Olahan)



Gambar 4.18 Tampilan area tujuh terbaca
(Sumber : Data Olahan)



Gambar 4.19 Tampilan area delapan terbaca

(Sumber : Data Olahan)

4.1.10. Tampilan Titik Koordinat Jari Telunjuk

Pada tampilan ini adalah hasil dari posisi tangan berada yang secara terus menerus dibaca oleh sensor Leap Motion karena sensor membaca gerakan tangan secara real time.

```

Output - LeapMotion (run) x
coba: TYPE_INDEX
      TYPE_DISTAL, end: (-33.0674, 40.4614, 97.5901)
      TYPE_DISTAL, end: (-14.2792, 36.4546, 125.834)
      TYPE_DISTAL, end: (-26.9209, 38.7708, 139.684)
      TYPE_DISTAL, end: (-43.1959, 41.9412, 141.213)
Nilai X :-43.195945739746094
Nilai Y :41.94116973876953
Nilai Z :141.21255493164062
  
```

Gambar 4.20 Tampilan Koordinat Jari Telunjuk

(Sumber : Data Olahan)

Pada gambar 4.20 adalah posisi telunjuk berada dengan nilai X, Y dan Z, nilai ini adalah sebagai data yang akan di tangkap oleh sensor Leap Motion untuk

menentukan apakah posisi jari berada pada area mapping dengan nilai yang telah di tentukan sebelumnya.

4.2. Pengujian Sistem

Pada Pengujian sistem dilakukan bertujuan untuk mengetahui hasil dari sistem yang telah dibuat apakah sudah sesuai yang diharapkan. Sistem dibuat menggunakan *Software NetBeans* untuk membuat sistem aplikasi piano digital.

4.2.1 Pengujian aplikasi

Pengujian aplikasi ini menggunakan perangkat komputer dengan beberapa pengguna dengan cara melakukan percobaan terhadap aplikasi. Setelah melakukan pengujian aplikasi maka di dapatkan hasil pengujian seperti yang terdapat pada Tabel 4.1 Pengujian Aplikasi

berikut :

Tabel 4.1 Pengujian Aplikasi

No	Pengguna	Pengujian								Keterangan
		A1	A2	A3	A4	A5	A6	A7	A8	
1	Aidil Kusuma	2X	1X	2X	4X	2X	1X	3X	1X	Sedikit sulit untuk menepatkan jari pada area mapping
2	Bagus Tri Wahyudi	1X	1X	3X	1X	2X	2X	1X	1X	Aplikasinya sudah bagus tetapi sedikit sulit untuk memainkannya
3	Irfan Hakim	1X	2X	1X	1X	3X	2X	2X	1X	Masih bingung untuk cara memainkannya
4	Muhammad Akmal	2X	2X	1X	2X	2X	1X	3X	1X	Aplikasi sudah bagus namun suara delaynya terlalu lama

5	Herdiyanto	3X	1X	1X	2X	1X	1X	2X	1X	Sudah bagus hanya saja memerlukan alat bantu seperti kaca
6	Mhd. Fakhmi	1X	1X	1X	2X	2X	1X	2X	1X	Aplikasinya sudah baik, tapi terkadang leap motion sedikit lambat membaca jari
7	Anggi Saputra	2X	2X	1X	1X	1X	2X	1X	1X	Sepertinya aplikasi ini membutuhkan alat bantu untuk bermain

Pada tabel pengujian aplikasi di atas dilakukan oleh 7 orang pengguna, pada setiap pengguna mendapatkan hasil yang berbeda – beda, dan komentar yang berbeda juga. Maksud dari X pada tabel tersebut adalah jumlah perulangan pengujian setiap pengguna pada setiap area.

4.2.1 Pengujian Suara

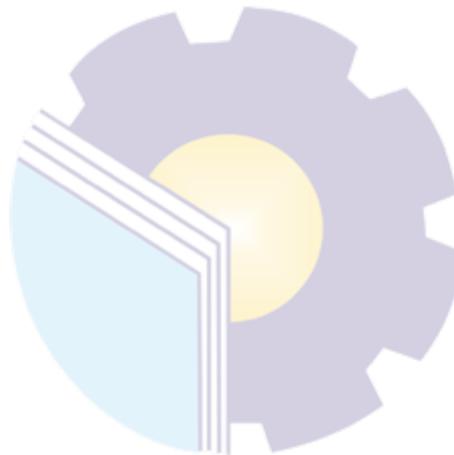
Berikutnya adalah pengujian terhadap suara yang dihasilkan dari setiap octaf, dengan pengujian ini dapat disimpulkan bahwa aplikasi dapat mengeluarkan suara dengan baik atau tidak. Berikut adalah tabel pengujian yang telah dilakukan.

Tabel 4.2 Pengujian Suara

No	Jenis Octaf	Suara								Hasil
		1	2	3	4	5	6	7	8	
1	Octaf 1	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil
2	Octaf 2	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil
3	Octaf 3	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil
4	Octaf 4	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil
5	Octaf 5	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil
6	Octaf 6	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil
7	Octaf 7	✓	✓	✓	✓	✓	✓	✓	✓	Berhasil

Dari hasil uji coba yang dilakukan, aplikasi berjalan dengan baik, aplikasi dapat digunakan oleh siapa saja di setiap perangkat komputer, hanya saja aplikasi saat dijalankan perlu perangkat komputer dengan spesifikasi yang lebih tinggi supaya aplikasi dapat bekerja maksimal.

Secara keseluruhan sistem yang digunakan sudah berjalan dengan maksimal. Tetapi, para pengguna banyak yang kesulitan untuk memainkannya karna, pengguna sulit untuk menepatkan jari pada area mapping yang telah di tentukan, dan juga aplikasi ini membutuhkan alat bantu untuk memainkannya seperti kaca, agar posisi area mapping dapat bisa di ketahui memalui gambaran di atas alat bantu tersebut.



BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pengujian sistem yang telah dilakukan dalam membangun aplikasi piano digital dengan menggunakan sensor *leap motion*, aplikasi dapat digunakan oleh siapa saja di setiap perangkat komputer, hanya saja aplikasi saat dijalankan perlu perangkat komputer dengan spesifikasi yang lebih tinggi supaya aplikasi dapat bekerja. Secara khusus dapat kita tarik kesimpulan dari aplikasi ini yaitu sebagai berikut

1. Aplikasi Piano Digital dibuat dengan menggunakan sensor *leap motion* dan bahasa pemrograman java.
2. Uji coba dilakukan oleh 6 orang pengguna pada setiap tuts piano dan 7 octaf suara.
3. Aplikasi ini dapat beroperasi dengan menggunakan jari telunjuk pengguna diatas sensor *leap motion*.
4. Pada aplikasi ini, suara dan *area mapping* berjalan sesuai dengan rancangan awal pembuatan aplikasi.

5.2 Saran

Beberapa saran yang diberikan sebagai sarana pengembangan yang lebih baik kedepannya yaitu:

1. Untuk hasil optimal disarankan menggunakan komputer dengan spesifikasi yang lebih tinggi dan baik.
2. Aplikasi ini membutuhkan alat bantu untuk bermain seperti kaca, agar pengguna dapat mudah untuk mengenali posisi *area mapping* yang telah ditentukan.

DAFTAR PUSTAKA

- Silva, S, Eduardo., Anderson, Jader., Henrique, Janiel., Teichrieb, V., & Ramalho, G. (2013). A Preliminary Evaluation of the Leap Motion Sensor as Controller of New Digital Musical Instruments. *Centro de Informatica*, 12. Retrieved from http://compmus.ime.usp.br/sbcm/2013/pt/docs/art_tec_1.pdf
- Basuki, A., Zikky, M., Akhmad, J., Hasim, N., & Ramadhan, N. I. (2016). Motion Sensor With Leap Motion To Help Deaf, 8(1994), 317–321.
- Enda, D., & Nasir, M. (2020). Pengenalan Bentuk Tangan Secara Real Time Menggunakan Leap Motion Dan K-Nearest Neighbors, 5(2), 178–184.
- Fernando, R., Supria, S., & Nasir, M. (2017). Aplikasi Marawis Digital Menggunakan Sensor Leap Motion. *INOVTEK Polbeng - Seri Informatika*, 2(2), 141. <https://doi.org/10.35314/isi.v2i2.201>
- Rohensih, Suwarni. (2015). APLIKASI DICTIONARY by DEKSTOP UNTUK 5000 KATA, 1(1), 2015. Retrieved from <http://weekly.cnbnews.com/news/article.html?no=124000>
- Hirza, H. (2018). METODE ANSAMBLE KEYBOARD SALAH SATU CARA MUDAH BERMAIN PIANO.
- Warno. (2012). Pembelajaran Pemrograman Bahasa Java Dan Arti Keyword. *Pembelajaran Pemrograman Bahasa Java Dan Arti Keyword*, 8, 40–51.
- Mooy, T. I., Dr. Darlis Herumurti, S.Kom., M. K., & Ridho Rahman Hariadi, S.Kom., M. . (2017). *Virtual Sasando , Sebagai Media Memainkan Alat Musik*.

Yowanda, A G., Sunaryo, D., Hariadi, R R., 2014, Rancang Bangun Aplikasi Papantulis Virtual Dengan Menggunakan Leap Motion, Jurnal Teknik Pomits, Vol. 3, Nomor 2, hal. 1-6, ISSN : 2337-3539.

