

**APLIKASI TRANSLITERASI NOT BALOK DENGAN STRUKTUR  
FILE MusicXML MENJADI NOT ANGKA BERBASIS ANDROID**

**Skripsi**

**Untuk memenuhi sebagian persyaratan  
mencapai derajat Sarjana S-1**

**Program Studi Teknik Informatika**



**diajukan oleh**

**Lutvia Mathofani**

**10650023**

**Kepada**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA**

**YOGYAKARTA**

**2014**



Universitas Islam Negeri Sunan Kalijaga

FM-UINSK-BM-05-07/R0

**PENGESAHAN SKRIPSI/TUGAS AKHIR**


Nomor : UIN.02/D.ST/PP.01.1/1810/2014

Skripsi/Tugas Akhir dengan judul : Aplikasi Transliterasi Not Balok Dengan Struktur File MusicXML Menjadi Not Angka Berbasis Android


Yang dipersiapkan dan disusun oleh :  
Nama : Lutvia Mathofani  
NIM : 10650023  
Telah dimunaqasyahkan pada : Jum'at, 20 Juni 2014  
Nilai Munaqasyah : A / B  
Dan dinyatakan telah diterima oleh Fakultas Sains dan Teknologi UIN Sunan Kalijaga

**TIM MUNAQASYAH :**


Ketua Sidang

  
Aulia Faqih R, M.Kom  
NIP. 19860306 201101 1 009


Penguji I

  
Agung Fatwanto, Ph.D  
NIP.19770103 200501 1 003

Penguji II

  
Ade Ratnasari, M.T  
NIP. 19801217 200604 2 002

Yogyakarta, 23 Juni 2014  
UIN Sunan Kalijaga  
Fakultas Sains dan Teknologi  
Dekan

  
  
Prof. Drs. H. Akh. Minhaji, M.A, Ph.D  
NIP. 19580919 198608 1 002



**SURAT PERSETUJUAN SKRIPSI/TUGAS AKHIR**

Hal : Permohonan

Lamp :-

Kepada

Yth. Dekan Fakultas Sains dan Teknologi

UIN Sunan Kalijaga Yogyakarta

di Yogyakarta

*Assalamu'alaikum wr wb*

Setelah membaca, meneliti, memberikan petunjuk dan mengoreksi serta mengadakan perbaikan seperlunya, maka kami selaku pembimbing berpendapat bahwa skripsi Saudara:

Nama : Lutvia Mathofani

NIM : 10650023

Judul Skripsi : Aplikasi Transliterasi Not Balok dengan Struktur File MusicXML menjadi Not Angka Berbasis Android

sudah dapat diajukan kembali kepada Program Studi Tekni Informatika Fakultas Sains dan Teknologi UIN Sunan Kalijaga Yogyakarta sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu dalam Teknik Informatika

Dengan ini kami berharap agar skripsi/tugas akhir Saudara tersebut di atas dapat segera dimunaqsyahkan. Atas perhatiannya kami ucapkan terima kasih.

*Wassalamu'alaikum wr. wb.*

Yogyakarta, 13 Juni 2014

Pembimbing

Aulia Faqih Rifa'i, M.Kom

NIP: 19860306 201101 1 009

## PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini :

Nama : Lutvia Mathofani  
Nim : 10650023  
Program Studi : Teknik Informatika  
Fakultas : Sains dan Teknologi

Menyatakan bahwa skripsi dengan judul **Aplikasi Transliterasi Not Balok dengan Struktur File MusicXML menjadi Not Angka Berbasis Android** tidak terdapat pada karya yang pernah diajukan untuk memperoleh gelar sarjana di suatu Perguruan Tinggi, dan sepengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 17 Juni 2014

Yang Menyatakan,



Lutvia Mathofani  
NIM : 10650023

## KATA PENGANTAR

Alhamdulillah, puji dan syukur senantiasa penulis panjatkan kepada Allah SWT atas segala rahmat, hidayah dan petunjuk-Nya, sehingga penulis akhirnya dapat menyelesaikan tugas akhir yang berjudul “ **Aplikasi transliterasi not balok dengan struktur file musicXML menjadi not angka berbasis android**”. Laporan tugas akhir ini disusun sebagai syarat untuk memperoleh gelar sarjana Teknik Informatika di Universitas Islam Negeri Sunan Kalijaga Yogyakarta.

Penulis mengucapkan terima kasih kepada pihak – pihak yang telah membantu proses pelaksanaan tugas akhir ini sehingga laporan tugas akhir ini dapat terselesaikan.

Selanjutnya penulis mengucapkan terima kasih kepada:

1. Ayah dan bu'eh tersayang, terimakasih atas segala kasih sayang dan dukungannya selama ini .
2. Bapak Prof. Dr. Akh Minhaji, selaku Dekan Fakultas Sains & Teknologi UIN Sunan Kalijaga.
3. Bapak Agus Mulyanto, M.Kom, selaku Ketua Program Studi Teknik Informatika Fakultas Sains & Teknologi UIN Sunan Kalijaga.
4. Bapak Aulia Faqih Rifa'i, M.Kom., selaku Pembimbing yang dengan kesabarannya telah membimbing selama ini.
5. Bapak Ibu dosen Teknik Informatika yang telah memberi bekal ilmu pengetahuan kepada penulis, semoga ilmunya menjadi amal jariyah di dunia hingga akhirat.

6. Teman- teman Monster Informatika (Teknik Informatika 2010) terima kasih atas dukungan dan motivasinya.
7. Kakak – kakak angkatan 2009, 2008, 2007 dan keluarga besar Teknik Informatika pada umumnya terima kasih untuk semuanya.
8. Keluargaku teman – teman UKM PSM Gita Savana UIN Sunan Kalijaga terima kasih atas semua pengalaman yang tak terlupakan.
9. Teman – teman KKN dan seluruh warga Giricahyo , Purwosari, Gunung Kidul terima kasih, semoga amal baik kalian dibalas Allah Subhanahu wa Ta’ala. Aamiin.
10. Terima kasih juga teman – teman kost 8A yang juga menjadi keluarga baru di Yogyakarta.

Masih banyak sekali kekurangan dalam laporan ini. Oleh karena itu, kritik dan saran senantiasa penulis harapkan. Akhirnya, semoga laporan tugas akhir ini dapat bermanfaat bagi pembaca sekalian.

Yogyakarta, 16 Juni 2014

**Lutvia Mathofani**

## **HALAMAN PERSEMBAHAN**

Penulisan skripsi ini penulis persembahkan kepada:

1. Allah SWT atas rahmat dan hidayah- Nya sehingga laporan skripsi dapat terselesaikan dengan baik.
2. Sholawat serta salam kepada junjungan Nabi Muhammad SAW.
3. Ibuku Siti Fathonah dan ayahku Thoyibul Umam serta kedua adikku tercinta Laila Marthatilova dan Lusy Rizkya Millyartha. Apah – bu'e terima kasih untuk semuanya, you are my everything.
4. Mbah uti, Pakde Imam Susilo dan keluarga, Om Satriyo Prabowo dan keluarga terima kasih atas doa dan dukungannya.
5. Almamaterku UIN Sunan Kalijaga Yogyakarta.
6. Prodi Teknik Informatika, Fakultas Sains dan Teknologi.
7. Dan Para pembaca yang budiman, terima kasih telah menyempatkan diri membaca karya ini.

## MOTTO

قَارِنَ مَعَ الْعُسْرِ يُسْرًا

*Fa inna ma'al 'usriyusraa*

*Maka sesungguhnya bersama kesulitan ada kemudahan*

***QS. Asy-Syarh (94:5)***

Music is a kind of counting performed by the mind without knowing that it is counting.

***G.W. Leibnitz (1714)***

Mathematics is music for the mind. Music is mathematics for the soul.

***Anonymous (18th century)***

Music washes away from the soul the dust of everyday life.

***Johan Sebastian Bach - Musikus***

Take it slow and see what's going on today.

***Anonymous***

Jika kamu tidak mengejar apa yang kamu inginkan;

maka kamu tidak akan pernah memilikinya.

Jika kamu tidak bertanya; maka jawabannya adalah tidak.

Jika kamu tidak mengambil langkah maju; maka kamu selalu berada di tempat yang sama.

***Nora Roberts***



## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN .....	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PERNYATAAN .....	iv
KATA PENGANTAR .....	v
HALAMAN PERSEMBAHAN .....	vii
MOTTO .....	viii
DAFTAR ISI .....	ix
DAFTAR GAMBAR .....	xii
DAFTAR TABEL .....	xiv
ABSTRACT.....	xv
INTISARI.....	xvi
<b>BAB I PENDAHULUAN</b>	
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah .....	4
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
<b>BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI</b>	
2.1 Tinjauan Pustaka .....	6
2.2 Landasan Teori.....	10
2.2.1 Pemrograman Berorientasi Objek.....	10

2.2.2 Musicxml .....	27
2.2.3 Teori Musik.....	32
2.2.4 Teknik Konversi Not Balok ke Not Angka.....	43
2.2.5 Unifed Modeling Language .....	46
2.2.6 Pengujian Usabilitas dan Fungsionalitas.....	51
<b>BAB III METODE PENELITIAN</b>	
3.1 Metode Penelitian .....	56
<b>BAB IV ANALISIS DAN PERANCANGAN SISTEM</b>	
4.1 Analisis Kebutuhan .....	64
4.1.1 Analisis Fungsional .....	64
4.1.2 Analisis Non Fungsional .....	65
4.1.2 Analisis Pengguna .....	66
4.2 Perancangan Sistem .....	67
4.2.1 Gambaran Umum Sistem .....	67
4.2.2 Desain UML .....	76
4.2.3 Desain Interface .....	78
<b>BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM</b>	
5.1 Implementasi Sistem .....	82
5.1.1 Halaman Utama/Home .....	82
5.1.2 Halaman Pilih File .....	83
5.1.3 Halaman Konversi Info Utama .....	84
5.1.4 Halaman Konversi ke Not Angka .....	86
5.2 Pengujian Sistem .....	86

5.2.1 Penguian Alpha .....	86
----------------------------	----

5.2.2 Pengujian Beta .....	87
----------------------------	----

## BAB VI HASIL DAN PEMBAHASAN

6.1 Hasil .....	90
-----------------	----

6.1.1 Penguian Alpha .....	90
----------------------------	----

6.1.2 Pengujian Beta .....	91
----------------------------	----

6.2 Manfaat Aplikasi .....	94
----------------------------	----

6.3 Keandalan Aplikasi .....	94
------------------------------	----

6.4 Kendala .....	94
-------------------	----

## BAB VII PENUTUP

7.1 Kesimpulan .....	96
----------------------	----

7.2 Saran .....	97
-----------------	----

## DAFTAR PUSTAKA

## LAMPIRAN

## DAFTAR GAMBAR

Gambar 2.1 .....	11
Gambar 2.2 .....	27
Gambar 2.3 .....	29
Gambar 2.4 .....	30
Gambar 2.5 .....	30
Gambar 2.6 .....	31
Gambar 2.7 .....	31
Gambar 2.8 .....	33
Gambar 2.9 .....	33
Gambar 2.10 .....	34
Gambar 2.11 .....	37
Gambar 2.12 .....	37
Gambar 2.13 .....	39
Gambar 2.14 .....	41
Gambar 2.15 .....	42
Gambar 2.16 .....	43
Gambar 2.17 .....	44
Gambar 3.1 .....	59
Gambar 4.1 .....	67
Gambar 4.2 .....	68
Gambar 4.3 .....	69

Gambar 4.4 .....	71
Gambar 4.5 .....	72
Gambar 4.6 .....	73
Gambar 4.7 .....	76
Gambar 4.8 .....	77
Gambar 4.9 .....	78
Gambar 4.10 .....	79
Gambar 4.11 .....	79
Gambar 4.12 .....	80
Gambar 4.13 .....	81
Gambar 5.1 .....	82
Gambar 5.2 .....	83
Gambar 5.3 .....	83
Gambar 5.4 .....	84
Gambar 5.6 .....	86

## DAFTAR TABEL

Tabel 2.1.....	9
Tabel 2.2.....	12
Tabel 2.3 .....	35
Tabel 2.4 .....	48
Tabel 5.1 .....	87
Tabel 5.2 .....	88
Tabel 5.3 .....	88
Tabel 6.1 .....	90
Tabel 6.2 .....	92
Tabel 6.3 .....	92

# **APLIKASI TRANSLITERASI NOT BALOK DENGAN STRUKTUR FILE MusicXML MENJADI NOT ANGKA BERBASIS ANDROID**

**Lutvia Mathofani**

**10650023**

## **INTISARI**

*Di Indonesia, terutama bagi sebagian pemusik dan penyanyi kemampuan membaca not baloknya masih belum seperti sebagaimana yang diharapkan. Umumnya praktisi musik di Indonesia masih menggunakan partitur not angka, sedangkan standar penulisan notasi yang berlaku secara internasional adalah notasi balok. Membaca not balok adalah hal yang relatif lebih sulit bila dibandingkan dengan membaca not angka. Hal ini terkait dengan simbol-simbol yang identik antara notasi satu dengan notasi lain. Kesulitan tersebut ditambah dengan kemampuan membaca patokan nada dasar antar satu lagu dengan yang lain. Bila dilakukan secara manual merupakan pekerjaan yang kurang efektif dan efisien, karena membaca not balok sangat rumit untuk dilakukan dengan banyak variasi simbol dan bunyi, serta pembacaan kombinasi simbol yang harus dilakukan secara vertikal dan horizontal. Oleh karena itu, diperlukan transliterasi yang lebih representatif.*

*Aplikasi transliterasi not balok menjadi not angka yang selanjutnya akan disebut “POP UP 123” (Do Re Mi pop up) dibangun menggunakan metode perancangan perangkat lunak Waterfall. Penelitian akan dilakukan menggunakan sebuah komputer dengan spesifikasi cukup untuk mengembangkan aplikasi berbasis android. Bahasa pemrograman yang digunakan adalah java dan menggunakan software eclipse.*

*Dengan adanya aplikasi konversi notasi balok ke notasi angka maka anggota paduan suara maupun orang yang awam terhadap musik, yang tidak dapat membaca notasi balok dapat menggunakan aplikasi tersebut sehingga dapat mempermudah untuk mengenali dan mempelajari notasi balok dan notasi angka. **Keyword: Not balok, not angka, file musicXML, android, konversi***

## ABSTRACT

*In Indonesia, especially for some musicians and singers the ability to read music is not like his logs as expected. General practitioners use music in Indonesia is still not figure scores, while the writing standard notation is the internationally accepted notation beams. Read not beam is relatively more difficult when compared with the numbered notations. It is associated with the symbols that are identical between the notation one another. The difficulty coupled with the ability to read standard tones between one song to another. When done manually is a job that is less effective and efficient, because it is very complicated to read music to do with the many variations of symbols and sounds, as well as the reading of symbol combinations that must be done vertically and horizontally. Therefore we need a more representative transliteration.*

*Transliteration application block notation into the numbered notation will be called "POP UP 123" (Do Re Mi pop up) is built using the Waterfall method of software design. The research will be created using a computer with the specs enough to develop applications based on Android. The programming language used Java and using Eclipse software.*

*With the conversion application a choir member and common people , who can not read notation beams can use these applications so as to make it easier to identify and study the block notation and numbered notation.*

**Keyword: Not beams, numbered notation, MusicXML file, android, conversion**



# `BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi pada era sekarang telah mengalami kemajuan yang sangat pesat. Perkembangan teknologi telah merambah ke berbagai disiplin ilmu. Misalnya musik berbasis android seperti penelitian tentang aplikasi virtual gamelan (Nugraha, 2012) yang memberikan alternatif untuk memainkan instrumen gamelan. Musik adalah salah satunya, ilmu ini juga sangat membutuhkan dukungan teknologi untuk dapat dengan mudah dipelajari oleh masyarakat luas. “*The music in my life will set my spirit free.*” Penggalan lirik lagu As long As I have Music ini berisi bahwa musik dapat berfungsi sebagai ungkapan perhatian, baik bagi para pendengar yang mendengarkan maupun bagi pemusik yang menggubahnya. Musik atau dalam Bahasa Inggris disebut *music* dan dalam Bahasa Belanda disebut *muziek* adalah seni pengungkapan gagasan melalui bunyi, unsur dasarnya berupa melodi, irama, dan harmoni, dengan unsur pendukung berupa bentuk gagasan, sifat, dan warna bunyi. Namun dalam penyajiannya, sering masih berpadu dengan unsur – unsur lain (Savana, 2011). Dalam musik dikenal juga istilah nada. Nada berarti satuan bunyi atau suara yang getarannya teratur dengan tingkat yang juga tetap. Standar penulisan notasi yang berlaku secara internasional adalah notasi balok. Notasi balok adalah notasi yang satuan – satuannya berupa lambang gambar. Sedangkan not angka adalah

penulisan notasi musik dengan angka sebagai simbolnya(Savana, 2011). Not balok sudah diakui di seluruh dunia sejak ratusan tahun silam. Namun untuk bisa menguasainya dibutuhkan waktu yang cukup panjang.

Kemampuan membaca not balok pada kalangan musisi ataupun praktisi seni musik Indonesia tidak sebagaimana yang diharapkan, dalam arti musisi seringkali hanya dapat membaca not angka.(Chrisantyo, dkk., 2007) Bagi sebagian pemusik terutama penyanyi di Indonesia, membaca not balok adalah hal yang relatif lebih sulit bila dibandingkan dengan membaca not angka. Hal ini terkait dengan simbol-simbol yang identik antara notasi satu dengan notasi lain. Kesulitan tersebut ditambah dengan kemampuan membaca patokan nada dasar antar satu lagu dengan yang lain. Bila dilakukan secara manual merupakan pekerjaan yang kurang efektif dan efisien, karena membaca not balok sangat rumit untuk dilakukan dengan banyak variasi simbol dan bunyi, serta pembacaan kombinasi simbol yang harus dilakukan secara vertikal dan horizontal. Oleh karena itu diperlukan transliterasi yang lebih representatif.

Transliterasi itu sendiri berarti peralihan atau perpindahan yang dalam hal ini peralihan dari not balok ke not angka agar mudah dibaca dan dipahami. Mengadopsi kemajuan teknologi yang berkembang pada saat ini diharapkan dapat membantu para musisi untuk membaca notasi balok dengan aplikasi transliterasi not balok menjadi not angka. Kemajuan teknologi tersebut baik yang berupa perangkat keras maupun yang berupa perangkat lunaknya. Untuk mempermudah proses konversi tersebut dirancang sebuah program bantu dengan memanfaatkan file musicXML. Cara membuat file MusicXML dapat dilakukan

dengan program penyunting not balok menjadi MusicXML, misalnya saja Finale atau SmartscoreX(Newell, 2008).

Berdasar alinea diatas selanjutnya dapat diidentifikasi beberapa permasalahan yang muncul. Permasalahan-permasalahan tersebut diantaranya: Apakah dalam Teknik Informatika memuat perangkat lunak yang dapat mengubah not balok menjadi not angka yang mengkonversi keseluruhan lagu dengan baik dan benar. Apakah hasil konversi dapat dimodifikasi, dibaca serta dinyanyikan dengan baik dan benar. Dan apakah aplikasi transliterasi not balok menjadi not angka berbasis android dapat ditanamkan disemua tablet maupun smartpone.

Permasalahan yang teridentifikasi tersebut selanjutnya dipilih satu permasalahan yang akan dikaji dalam tulisan atau penelitian ini. Yaitu apakah dalam Teknik Informatika memuat perangkat lunak yang dapat mengubah not balok menjadi not angka dan perangkat keras apa saja yang dapat mendukung sehingga perangkat lunak transliterasi not balok dapat ditanamkan.

Dari sini akan diketahui kegunaan penelitian, yaitu seberapa jauh aplikasi dapat mengkonversi keseluruhan lagu. Penelitian akan dilakukan menggunakan sebuah komputer dengan spesifikasi cukup untuk mengembangkan aplikasi berbasis android. Bahasa pemrograman yang digunakan adalah java dan menggunakan software eclipse.

## 1.2 Rumusan Masalah

Berdasar latar belakang tersebut, maka rumusan masalah dalam penelitian ini adalah bagaimana membuat aplikasi transliterasi not balok menjadi not angka yang bisa dijalankan pada sistem operasi android pada tablet maupun smartphone.

## 1.3 Batasan Masalah

Batasan masalah pada penelitian ini yaitu:

1. Konsep notasi angka dan notasi balok pada lagu dengan sumber yang digunakan berupa representasi not balok dalam file MusicXML.
2. Aplikasi konverter dari not balok ke not angka dimana bentuk notasi yang akan dikonversi yaitu not penuh, setengah hingga not seperenambelas.
3. Aplikasi berbasis android yang menyajikan satu metode konversi yang cepat.
4. Aplikasi konverter not balok menjadi not angka untuk satu jenis suara saja.
5. Tanda legato, type, nilai not dan lirik tidak termasuk dalam proses konversi.
6. Nada dasar yang dikonversi adalah nada dasar natural C=do dan nada dasar dengan tanda kromatis (mayor).

#### **1.4 Tujuan Penelitian**

Tujuan penelitian ini adalah menciptakan aplikasi transliterasi not balok menjadi not angka yang bisa dijalankan pada sistem operasi android pada tablet maupun smartphone .

#### **1.5 Manfaat Penelitian**

Manfaat penelitian ini adalah:

1. Membuat aplikasi yang dapat mengkonversi not balok menjadi not angka yang dapat dijalankan pada sistem operasi android sehingga memudahkan user untuk mengenali notasi dan bisa belajar dengan menyenangkan dimana dan kapan saja.
2. Mengenalkan lebih dekat tentang android kepada masyarakat.
3. Untuk mengetahui seberapa jauh file Musicxml dapat dimanfaatkan informasinya untuk mengkonversi not balok menjadi not angka dengan cepat dan akurat.

## **BAB VII**

### **PENUTUP**

#### 7.1 Kesimpulan

Berdasarkan hasil pengujian yang telah dilakukan penulis pada aplikasi “ POP UP 123” (*Do Re Mi pop up*) berbasis android maka dapat diambil kesimpulan bahwa penelitian ini berhasil .

Dalam menjalankan fungsinya sebagai perangkat konversi, program ini telah berfungsi. Dengan adanya aplikasi ini menunjukkan bahwa:

1. Aplikasi transliterasi not balok ke not angka juga dapat berjalan di sistem operasi android baik tablet maupun smartphome.
2. File Musicxml dapat diparsing dengan cepat serta merupakan format penyimpanan not balok yang cocok untuk dikonversi menjadi not angka, karena menyimpan begitu banyak informasi not namun tetap dapat dibaca dengan mudah oleh berbagai bahasa pemrograman.
3. Nada dasar yang dapat dikonversi adalah nada C=do dan tangga nada mayor.

#### 7.1 Saran

Penelitian yang dilakukan tidak terlepas dari kekurangan dan kelemahan. Oleh karena itu, untuk kebaikan pengembangan aplikasi lebih lanjut, maka perlu diperhatikan beberapa hal, diantaranya:

1. Dalam pembuatan aplikasi diperlukan *trappingerror*
2. Aplikasi ini perlu ditambahkan fitur ubah nada, informasi tanda tempo, tanda legato dan lirik lagu.
3. Menu help juga dibutuhkan dalam aplikasi.

## DAFTAR PUSTAKA

- Anwariningsih, Sri Huning, dkk (2010) *Estimasi Bentuk Structuring Element Berdasar Representasi Obyek*.
- Barakbah, A.R. (2006) *Pemrograman Berbasis Objek*. diakses : 17 Oktober 2013.
- Chrisantyo, Lukas., dkk. (2007) *Program Konversi Not Balok dengan Struktur MusicXML ke Not Angka*.
- Chrisantyo, Lukas., dkk. (2012) *Pengembangan Aplikasi Konversi Representasi Not Balok ke Not Angka Untuk Paduan Suara Campur*.
- Cunningham (2005) *Suitability Of MusicXML As A Format For Computer Music Notation & Interchange*, University of Wales: North Wales.
- DiMarzio, J.F. (2008) *Android™- A Programmer's Guide*, McGraw-Hill.
- Fajar Nugraha, A. (2012) *Aplikasi Virtual Gamelan Berbasis Android*.
- Fajar Riyadi, R. (2012) *Aplikasi Alat Bantu Ajar Bermain Piano*.
- Faunaldy, G. (2012) *Simulasi Angklung Berbasis PC*.
- Gunawan, A. (2012) *Aplikasi Virtual DJ pada Smartphone Berbasis Sistem Operasi Android*.
- Huda, A.A. (2012) *24 Jam Pintar Pemrograman Android*, ImagineIT. diakses : 24 Mei 2013.
- Kusumaningsih, E. (2005) *Peningkatan Profesionalisme Melalui Pembelajaran Musik bagi Musisi GMAHK*. diakses : 2 Maret 2013.
- Luthfi, E.T. (2012) *Modul Kuliah Program Aplikasi Mobile*.
- Mulyanto, A.R. (2008) *Rekayasa Perangkat Lunak Jilid 1*, Direktorat Pembinaan Sekolah Menengah Kejuruan Departemen Pendidikan Nasional.
- Murphy, M.L. (2009) *Android Programming Tutorials*, CommonsWare. diakses : 28 Mei 2013.
- Newell, C. (2008) *Using\_SmarscoreX*, Musitek. diakses : 27 Mei 2013.
- Oktavila, Chatarina (2011) *Aplikasi Konversi Notasi Balok ke Notasi Angka (Studi Kasus di Sekolah Tinggi Musi)*.
- Recordare (2011), Recordare, LLC. diakses : 26 Oktober 2013.
- Sachirul Umam, I. (2012) *Aplikasi Virtual Gambang Menggunakan Hardware Controller*.
- Shalahuddin, M., dan Rosa A.S. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Beorientasi Obyek*. Bandung: Informatika.



Savana, G. (2011) *Modul Peserta Diksapara*, Yogyakarta.

Wibowo, H.C. (2010) *7 Hari Lancar Membaca Not Balok*, Yogyakarta: Citra Media.

**LAMPIRAN A****Kode Program****# MainActivity.Java**

```
package intent.com;
```

```
import java.io.File;
```

```
import java.io.FilenameFilter;
```

```
import android.app.Activity;
```

```
import android.content.Intent;
```

```
import android.os.Bundle;
```

```
import android.os.Environment;
```

```
import android.util.Log;
```

```
import android.view.Menu;
```

```
import android.view.View;
```

```
import android.view.View.OnClickListener;
```

```
import android.widget.Button;
```

```
public class MainActivity extends Activity implements OnClickListener {
```

```
    Button next;
```

```
    @Override
```

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    next = (Button)findViewById(R.id.Button01);  
    next.setOnClickListener(this);  
  
}  
  
@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
    // Inflate the menu; this adds items to the action bar if it is present.  
    getMenuInflater().inflate(R.menu.main, menu);  
    return true;  
}  
  
@Override  
public void onClick(View arg0) {  
    // TODO Auto-generated method stub  
    Intent i = new Intent(this, callActivity2.class);  
    startActivity(i);  
}  
//In an Activity  
}
```

## # CallActivity.Java

```
package intent.com;
```

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FilenameFilter;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.Vector;

import org.xmlpull.v1.XmlPullParser;
import org.xmlpull.v1.XmlPullParserException;
import org.xmlpull.v1.XmlPullParserFactory;

import android.app.Activity;
import android.app.AlertDialog;
import android.app.AlertDialog.Builder;
import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.webkit.WebView;
import android.widget.Button;
import android.widget.Toast;

public class callActivity2 extends Activity implements OnClickListener {
```

```

WebView webview;

String nada_dasar = null;

Button prev;

protected Button mBackButton;

protected Button mForwardButton;

Button pilihxml;

private String[] mFileList;

private File mPath = new File(Environment.getExternalStorageDirectory() +
"/skripsi/");

private String mChosenFile;

private File filechosen;

private static final String FTYPE = ".xml";

private static final int DIALOG_LOAD_FILE = 1000;

@Override

public void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.main);

    prev = (Button) findViewById(R.id.parsing);

    prev.setOnClickListener(this);

    mBackButton = (Button) findViewById(R.id.b2);

    mBackButton.setOnClickListener(this);

    mForwardButton = (Button) findViewById(R.id.b1);

    mForwardButton.setOnClickListener(this);

    webview = (WebView) findViewById(R.id.webkit2);

    // webview.loadUrl("file:///android_asset/test.html");

    pilihxml=(Button)findViewById(R.id.btnpilihxml);

    pilihxml.setOnClickListener(this);

```

```
}

void formatWebView(String hasil) {

    StringBuilder sb = new StringBuilder();
    sb.append("");
    sb.append("" + hasil);
    sb.append("</body></HTML>");

    webview.loadDataWithBaseURL("file:///android_asset/", sb.toString(),
        "text/html", "utf-8", null);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

public void onClick(View arg0) {
    // TODO Auto-generated method stub

    switch (arg0.getId()) {
    case R.id.parsing:
        try {
            String hasil = parseXml();
            formatWebView(hasil);
        } catch (XmlPullParserException e) {
```

```
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    break;

case R.id.b1:
    Intent i = new Intent(this, MainActivity.class);
    startActivity(i);
    break;

case R.id.b2:
    Intent j = new Intent(this, Parserku.class);
    j.putExtra("nada_dasar", nada_dasar);
    j.putExtra("filepath", filechosen);
    startActivity(j);
    break;

case R.id.btnpilihxml:
    loadFileList();

    showDialog(DIALOG_LOAD_FILE);
default:
    break;
}

}

public String getTempo(int p) {
```

```
String tmp = "";
switch (p) {
case 9:
    tmp = "Moderato";

    break;

default:
    break;
}
return tmp;
}

public String getNadaDasar(int kode) {
String nada = "";
switch (kode) {
case 0:
    nada = "C";
    break;

case 1:
    nada = "G";
    break;

case 2:
    nada = "D";
    break;

case 3:
    nada = "A";
    break;

case 4:
    nada = "E";
```



```
        break;
case 5:
    nada = "B";
    break;
case 6:
    nada = "F#";
    break;
case 7:
    nada = "C#";
    break;
case -1:
    nada = "F";
    break;
case -2:
    nada = "Bb";
    break;
case -3:
    nada = "Eb";
    break;
case -4:
    nada = "Ab";
    break;
case -5:
    nada = "Db";
    break;
case -6:
    nada = "Gb";
    break;
case -7:
    nada = "Cb";
```

```
        break;
    default:
        break;
    }

    return nada;
}

public String changeNada(String not) {
    String nada = "0";
    if (not.equalsIgnoreCase("C")) {
        nada = "1";
    } else if (not.equalsIgnoreCase("D")) {
        nada = "2";
    } else if (not.equalsIgnoreCase("E")) {
        nada = "3";
    } else if (not.equalsIgnoreCase("F")) {
        nada = "4";
    } else if (not.equalsIgnoreCase("G")) {
        nada = "5";
    } else if (not.equalsIgnoreCase("A")) {
        nada = "6";
    } else if (not.equalsIgnoreCase("B")) {
        nada = "7";
    }

    return nada;
}

public String parseXml() throws XmlPullParserException, IOException {
```

```

String credit = "";
String title_tag = "";
Vector<String> birama = new Vector<String>();

int jml_birama = 0;

XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
factory.setNamespaceAware(true);
XmlPullParser xpp = factory.newPullParser();
// System.out.println("parser implementation class is " +
// xpp.getClass());
///
InputStream is =
this.getResources().openRawResource(R.raw.diduniaku2);
// FileReader in = new FileReade
InputStream is=new FileInputStream(filechosen);
xpp.setInput(new InputStreamReader(is));

int eventType = xpp.getEventType();
while (eventType != xpp.END_DOCUMENT) {
    eventType = xpp.next();

    // Mencari Kredit
    if (eventType == xpp.START_TAG) {
        if (xpp.getName().equalsIgnoreCase("credit-words")) {
            eventType = xpp.next();
            if (eventType == xpp.TEXT) {
                credit += title_tag + xpp.getText() +
"<br/>";
            }
        }
    }
}

```

```

    }

    // Mencari nada dasar
    if (eventType == xpp.START_TAG) {

        if (xpp.getName().equalsIgnoreCase("fifths")) {
            eventType = xpp.next();
            if (eventType == xpp.TEXT) {
                credit += "\nNada Dasar= "
                    +
                    getNadaDasar(Integer.parseInt(xpp.getText()))
                    + "<br/>";
                nada_dasar =
                    getNadaDasar(Integer.parseInt(xpp
                        .getText()));
            }
        }
    }

    // Mencari beat
    if (eventType == xpp.START_TAG) {

        if (xpp.getName().equalsIgnoreCase("beats")) {
            eventType = xpp.next();
            if (eventType == xpp.TEXT) {
                credit += "Birama= " + xpp.getText() +
"/";
            }
        }
    }
}

```

```

        if (eventType == xpp.START_TAG) {
            if (xpp.getName().equalsIgnoreCase("beat-type")) {
                eventType = xpp.next();
                if (eventType == xpp.TEXT) {
                    credit += xpp.getText() + "<br/>";
                }
            }
        }

        // Mencari tempo
        if (eventType == xpp.START_TAG) {

            if (xpp.getName().equalsIgnoreCase("sound")) {
                eventType = xpp.next();
                if (eventType == xpp.TEXT) {
                    credit += "\n Tempo= "
                                +
                                getTempo(Integer.parseInt(xpp
                                    .getAttributeValue(0))) + "<br/>";
                }
            }
        }
    }
    return credit;
}

```

```

private void loadFileList() {
    try {
        mPath.mkdirs();
    } catch (SecurityException e) {
        Log.e("unable write",
            "unable to write on the sd card " + e.toString());
    }
    if (mPath.exists()) {
        FilenameFilter filter = new FilenameFilter() {
            public boolean accept(File dir, String filename) {
                File sel = new File(dir, filename);
                return filename.contains(FTYPE) ||
sel.isDirectory();
            }
        };
        mFileList = mPath.list(filter);
        Toast.makeText(this, "berhasil sampai sini",
Toast.LENGTH_LONG).show();
    } else {
        mFileList = new String[0];
        Toast.makeText(this, "gak ada isi",
Toast.LENGTH_LONG).show();
    }
}

protected Dialog onCreateDialog(int id) {
    Dialog dialog = null;
    AlertDialog.Builder builder = new Builder(this);

    switch (id) {
    case DIALOG_LOAD_FILE:

```

```

        builder.setTitle("Choose your file");
        if (mFileList == null) {
            Log.e("load file",
                "Showing file picker before loading the
file list");

            dialog = builder.create();
            return dialog;
        }
        builder.setItems(mFileList, new DialogInterface.OnClickListener()
{
            public void onClick(DialogInterface dialog, int which) {
                mChosenFile = mFileList[which];

                Toast.makeText(getApplicationContext(),mPath+mChosenFile.toString(),
                Toast.LENGTH_LONG).show();

                filechosen=new File(mPath+"/"+mChosenFile);
                // you can do stuff with the file here too
            }
        });
        break;
    }
    dialog = builder.show();
    return dialog;
}
}
}

```

### #Parserku.Java

```
package intent.com;
```

```
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.List;

import org.xmlpull.v1.XmlPullParser;
import org.xmlpull.v1.XmlPullParserException;
import org.xmlpull.v1.XmlPullParserFactory;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import android.widget.Toast;

public class Parserku extends Activity implements OnClickListener {
    WebView webview;
    String nada_dasar = null;
    Button prev;
    protected Button mBackButton;
    protected Button mForwardButton;
```



```

NadaDasar nadadasar;

float[] major = { 1, 1, 0.5f, 1, 1, 1, 0.5f };
char[] OctaveDasar = new char[7];

File path;

int jumlahnote=0;

@Override

public void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.main2);

    nadadasar = new NadaDasar(this);

    // statis
//    nada_dasar = "F#";

    // dinamis

    nada_dasar = (String) getIntent().getExtras().get("nada_dasar");
    nadadasar.setNadaDasar(nada_dasar, 0, getApplication());

    nadadasar.setOctave(4, this);
    nadadasar.setNadaMajor(nada_dasar);
    prev = (Button) findViewById(R.id.p);
    prev.setText("parserku");
    prev.setOnClickListener(this);

    mBackButton = (Button) findViewById(R.id.tombol2);
    mBackButton.setOnClickListener(this);

    //mForwardButton = (Button) findViewById(R.id.tombol);
    //mForwardButton.setOnClickListener(this);

```

```

webView = (WebView) findViewById(R.id.webView1);
webView.setWebViewClient(new WebViewClient() {

    public void onPageFinished(WebView view, String url) {
        // do your stuff here

    }
});

// webView.loadUrl("file:///android_asset/test.html");

// dynamis
path = (File) getIntent().getExtras().get("filepath");
}

void formatWebView(String hasil) {

    StringBuilder sb = new StringBuilder();
    sb.append("<HTML><HEAD><LINK href=\"stylesheet.css\"
type=\"text/css\" rel=\"stylesheet\"></HEAD><body>");
    sb.append("\" + hasil);
    sb.append("</body></HTML>");

}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

```

```
}

public void onClick(View arg0) {
    // TODO Auto-generated method stub

    switch (arg0.getId()) {
        case R.id.p:
            try {

                String hasil = parseXml();
                formatWebView(hasil);
                Log.e("jumlah note", "jumlah notenya
adalah"+jumlahnote);
                Toast.makeText(this,"jumlah notenya adalah"+ jumlahnote,
                Toast.LENGTH_LONG).show();
            } catch (XmlPullParserException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            } catch (IOException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
            }
            break;

        case R.id.b1:
            Intent i = new Intent(this, MainActivity.class);
            startActivity(i);
            break;

        case R.id.b2:
            Intent j = new Intent(this, Activityku.class);
```

```
        // j.putExtra("nada_dasar", nada_dasar);
        // startActivity(j);
        break;

    default:
        break;
    }
}

public String getTempo(int p) {

    String tmp = "";
    switch (p) {
    case 9:
        tmp = "Moderato";

        break;

    default:
        break;
    }
    return tmp;
}

public String parseXml() throws XmlPullParserException, IOException {

    String credit = "";
    String title_tag = "";
    List<String> birama = new ArrayList<String>();

    int jml_birama = 0;
```

```

XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
factory.setNamespaceAware(true);
XmlPullParser xpp = factory.newPullParser();
// System.out.println("parser implementation class is " +
// xpp.getClass());

// statis
//InputStream is =
this.getResources().openRawResource(R.raw.diduniaku);
// dinamis
InputStream is = new FileInputStream(path);

// FileReader in = new FileReade
xpp.setInput(new InputStreamReader(is));
String octave = null;
String step = null;
String alter = "0";
String fifthsoctave;
String typeduration;
int x = 0;
// setDasar('C');

int eventType = xpp.getEventType();
while (eventType != xpp.END_DOCUMENT) {
    eventType = xpp.next();

    // Mencari Kredit
    if (eventType == xpp.START_TAG) {
        if (xpp.getName().equalsIgnoreCase("measure")) {
            // birama.add(jml_birama, "1 ");

```

```

// Log.e("measure", "measure ke"+x);
// Log.e("measure", birama.get(jml_birama)+"
"+jml_birama);

    }
}

if (eventType == xpp.START_TAG) {
    if (xpp.getName().equalsIgnoreCase("rest")) {
        // String br = birama.get(jml_birama) + " 0 ";

        birama.add("0");
        Log.e("rest", birama.get(jml_birama) + " " +
jml_birama);

    }
}

if (eventType == xpp.START_TAG) {
    if (xpp.getName().equalsIgnoreCase("step")) {
        xpp.next();
        step = xpp.getText();
        xpp.nextTag();

    }

    if (xpp.getName().equalsIgnoreCase("octave")) {
        xpp.next();
        octave = xpp.getText();
        xpp.nextTag();

    }

    if (xpp.getName().equalsIgnoreCase("alter")) {

```

```

        xpp.next();
        alter = xpp.getText();
        xpp.nextTag();
    }
    if (xpp.getName().equalsIgnoreCase("typeduration")) {
        xpp.next();
        typeduration = xpp.getText();
        xpp.nextTag();
    }
}

if (eventType == xpp.END_TAG) {
    if (xpp.getName().equalsIgnoreCase("pitch")) {

        // Log.e("jalan", alter);
        jumlahnote++;

        birama.add(getNumericNote(Integer.parseInt(octave),
                                                    step.charAt(0),
Integer.parseInt(alter), 5));

        Log.e("pitch", birama.get(jml_birama) + " " +
jml_birama);

        //
        Log.e("logstep", Integer.toString(step.charAt(0)));

        dan " +
        // Log.e("logstep", "sampai step "+step+" sini"+ "
        // octave+" sini "+alter+" sini
"+Integer.toString(x));

        // birama.set(jml_birama,step);

```

```

        // x++;
        // step = null;
        // octave = null;
        alter = "0";
    }

    if (xpp.getName().equalsIgnoreCase("measure")) {
        // String br = birama.get(jml_birama) + " |";
        birama.add("|");
        Log.e("endmeasure", birama.get(jml_birama));

        jml_birama++;
    }
}

Log.e("jumlah birama", Integer.toString(birama.size()));
for (int i = 0; i < birama.size(); i++) {
    if (birama.get(i) != null) {
        // Log.e("birama", birama.get(i)+" " +Integer.toString(i));
        credit += (birama.get(i) + " ");
    }

    if ((i % 4) == 0) {
        // credit += "<br/>";
    }
}

return credit;
}

```



```

public String getNumericNote(int octave, char step, int alter,
    int fifths octave) {
    String numericnote = " ";
    // Character fifths = getfifths();

    numericnote = operationNote(octave, step, fifths octave,
        alter);
    // operationMajorMinor(
    // operationAsciiOctaveStep(octave, step, fifths, fifths octave),
    // alter, octave, fifths, fifths octave);
    return numericnote;
}

```

```

public String operationNote(int octave, char step,
    int standarnadadasar, int alter) {
    String newstep = null;
    if (alter > 0) {
        newstep = step + "#";
    } else if (alter < 0) {
        newstep = step + "b";
    } else {
        newstep = Character.toString(step);
    }

    if (newstep.equals("B#")) {
        newstep = "C";
    } else if (newstep.equals("E#")) {
        newstep = "F";
    }
}

```

```

}

// Toast.makeText(this,Integer.toString(nadadasar.getOctave(octave,
// Integer.parseInt(nadadasar.getNadaDasar(newstep))),
// Toast.LENGTH_LONG).show());

Integer difoctav = nadadasar.getOctave(octave,
                                     nadadasar.getNadaDasar(newstep));
// Log.e("octave",octave+" "+difoctav);
Log.e("step", newstep);
Log.e("newstep", nadadasar.getNadaDasar(newstep));

if (difoctav.equals(-1)) {
    newstep = "a" + nadadasar.getNadaDasar(newstep);
} else if (difoctav.equals(-2)) {

    newstep = "9" + nadadasar.getNadaDasar(newstep);
} else if (difoctav.equals(1)) {

    newstep = "s" + nadadasar.getNadaDasar(newstep);
} else if (difoctav.equals(2)) {

    newstep = "8" + nadadasar.getNadaDasar(newstep);
} else {

    newstep = nadadasar.getNadaDasar(newstep);
}

Log.e("newsteppp", newstep);
return newstep;

```

```

}

public String operationMajorMinor(int newstep, int alter, int octave,
    char fifths, int fifthsoctave) {

    int difoctav = fifthsoctave - octave;
    int numericfifthsOctave = (int) fifths + (7 * difoctav);
    int numericOctaveStep;

    numericOctaveStep = newstep - numericfifthsOctave;
    Log.e("log saya", " numeric" + Integer.toString(numericfifthsOctave)
        + " dif " + difoctav + "fifths " + (int) fifths);
    if (numericOctaveStep > 0) {
        if (alter > 0) {
            if (major[numericOctaveStep] == 0.5f) {
                numericOctaveStep++;
                return Integer.toString(numericOctaveStep);
            } else {
                return Integer.toString(numericOctaveStep) +
"##";
            }
        }
    } else {
        return Integer.toString(numericOctaveStep);
    }
}
}

```

```

        return "false";
    }
    /*
    public void setDasar(char fifths) {

        int dasar = (int) fifths;
        int cek;
        for (int x = 0; x < OctaveDasar.length; x++) {
            cek = dasar + x;
            if (cek > 71) {
                OctaveDasar[x] = (char) (cek - 7);
            } else {
                OctaveDasar[x] = (char) cek;
            }
            Log.e("octavedasar",
                Integer.toString(
                    String.valueOf(OctaveDasar[x]) + " " +
                        + " " + cek);
            }
        }
    }
    */
}

```

### #Nada Dasar.Java

```
package intent.com;
```

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import android.R.bool;
import android.content.Context;
import android.util.Log;
import android.widget.Toast;

public class NadaDasar {
    String[] kumpulanString = { "C", "C#", "D", "D#", "E", "F", "F#", "G",
        "G#", "A", "A#", "B" };
    String[] kumpulanString2 = { "C", "Db", "D", "Eb", "E", "F", "Gb", "G",
        "Ab", "A", "Bb", "B" };
    String[] kumpulanString3 = new String[12];
    List<String[]> kumpulan = new ArrayList<String[]>();
    String[] interval = { "1", "1", "1/2", "1", "1", "1", "1/2" };
    String[] kumpulannadadasar = new String[8];
    Integer[] kumpulanoctave = new Integer[8];
    Context context;

    public NadaDasar(Context context) {
        // TODO Auto-generated constructor stub
        this.context = context;
        kumpulan.add(kumpulanString);

        kumpulan.add(kumpulanString2);
        Toast.makeText(context, "Berhasildibuat",
            Toast.LENGTH_LONG).show();
    }
}
```

```
}

```

```
public void setNadaDasar(String dasar, int index, Context context) {
    int[] index_nada = new int[8];
    int idx = 0;

    index_nada[0] = Arrays.asList(kumpulan.get(index)).indexOf(dasar);
    index_nada[7] = Arrays.asList(kumpulan.get(index)).indexOf(dasar);
    System.out.println(index_nada[0] + " dan " + index_nada[7]);
    for (int i = 0; i < interval.length; i++) {
        if (i == 0) {
            // WSystem.out.print(index_nada[i] + " ");
            continue;
        }

        if (interval[i - 1].equalsIgnoreCase("1")) {
            // index geser 2 kali
            idx = index_nada[i - 1];
            idx += 2;

            if (idx >= 12) {
                idx = idx - 12;
            }
        }

        } else {
            // index geser 1 kali
            idx = index_nada[i - 1];
            idx += 1;
            if (idx >= 12) {
                idx = idx - 12;
            }
        }
    }
}
```

```

        }
    }

    index_nada[i] = idx;
    // System.out.print(idx + " ");
}
String nada = "";
for (int i = 0; i < index_nada.length; i++) {
    System.out.print(kumpulan.get(index)[index_nada[i]] + " ");
    Log.e("kumpulan", kumpulan.get(index)[index_nada[i]] + " "
        + index_nada.length);
    kumpulannadadasar[i] = kumpulan.get(index)[index_nada[i]];
    nada += kumpulannadadasar[i] + "";
}
Toast.makeText(context, nada, Toast.LENGTH_LONG).show();

}

public String getNadaDasar(String step) {
    String newstep = null;

    //GANTI B# dan E#

    int numericstep = Arrays.asList(kumpulannadadasar).indexOf(step) + 1;
    int cekindex;
    String cekstring;

    if (numericstep <= 0) {

```

```

cekindex = Arrays.asList(kumpulanString3).indexOf(step);
Log.e("getnadadasar", step+" "+numericstep+" "+cekindex);

cekindex--;
cekstring = kumpulanString3[cekindex];
numericstep =
Arrays.asList(kumpulannadadasar).indexOf(cekstring);

newstep = "/" + Integer.toString(numericstep);
} else {
newstep = Integer.toString(numericstep);
}

return newstep;
}

public void setOctave(int octave, Context context) {

boolean cek = false;
String sb = "";
for (int x = 0; x < kumpulannadadasar.length; x++) {
if (kumpulannadadasar[x].contains("C") && (x > 0) && (cek !=
true)) {

octave++;
cek = true;

}

kumpulanoctave[x] = octave;
sb += Integer.toString(kumpulanoctave[x]) + " ";
}

Toast.makeText(context, sb, Toast.LENGTH_LONG).show();

```



```

    }

    public int getOctave(int octave, String newstep) {
        int difoctave = 0;
        if( newstep.contains("/")){
            newstep=String.valueOf(newstep.charAt(1));
        }
        difoctave = kumpulanoctave[Integer.parseInt(newstep) - 1] - octave;
        Log.e("octave", kumpulanoctave[Integer.parseInt(newstep) - 1] + " " +
octave + " "
                + difoctave);
        return difoctave;
    }

    public void setNadaMajor(String nada_dasar) {

        int index = Arrays.asList(kumpulanString).indexOf(nada_dasar);
        String sb = "";
        for (int x = 0; x < kumpulanString.length; x++) {
            if (index >= kumpulanString.length) {
                index = 0;
            }
            kumpulanString3[x] = kumpulanString[index];
            sb += kumpulanString3[x] + " ";

            index++;
        }

        Toast.makeText(context, " nada bolong" + sb,
Toast.LENGTH_LONG).show();
    }

```

```
    }
```

```
}
```

### #Nuneric Convert.Java

```
package intent.com;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
//C 3 4 5 6
```

```
//D
```

```
public class NumericConvert {
```

```
    Character fifths;
```

```
    int fifthsoctave;
```

```
    List<Integer> octave = new ArrayList<Integer>();
```

```
    List<Character> step = new ArrayList<Character>();
```

```
    List<Integer> alter = new ArrayList<Integer>();
```

```
    List<Integer> duration = new ArrayList<Integer>();
```

```
    List<Integer> type = new ArrayList<Integer>();
```

```
    int beat;
```

```
    float[] major = { 1, 1, 0.5f, 1, 1, 1, 0.5f };
```

```
    float[] minor = {};
```

```
    public NumericConvert(Character fifths) {
```

```
        setfifths(fifths);
```

```
    }
```

```
public void setfifths(Character fifths) {
    this.fifths = fifths;
}

public Character getfifths() {
    return this.fifths;
}

public void setBeat(int beat){
    this.beat= beat;
}

public int getBeat(){
    return beat;
}

public void setOctave(int index, int input) {
    this.octave.set(index, input);
}

public int getOctave(int index) {
    return this.octave.get(index);
}

public void setStep(int index, Character input) {
    this.step.set(index, input);
}

public Character getStep(int index) {
    return this.step.get(index);
}
```

```
}

public void setAlter(int index, int input) {
    this.alter.set(index, input);
}

public int getAlter(int index) {
    return this.alter.get(index);
}

public void setDuration(int index, int input) {
    this.duration.set(index, input);
}

public int getDuration(int index) {
    return this.duration.get(index);
}

public void setType(int index, int input) {
    this.type.set(index, input);
}

public int getType(int index) {
    return this.type.get(index);
}

public String getNumericNote() {
```

```

String numericnote=" ";
Character fifths = getfifths();

for (int x=0;x<step.size();x++){
    numericnote=numericnote+"
"+operationMajorMinor(operationAsciiOctaveStep(getOctave(x),getStep(x), fifths),
getAlter(x), getOctave(x), fifths)+" ";
}
return numericnote;
}

public int operationAsciiOctaveStep(int octave, char step, char fifths) {
    int difoctav = fifthsoctave - octave;
    int i = 0;

    int newstep = (int) step;
    while (!(Integer.valueOf(i) == Integer.valueOf(difoctav))) {

        if (difoctav > 0) {
            newstep = newstep + 7;
            difoctav--;
        } else {
            newstep = newstep - 7;
            difoctav++;
        }
    }
}

// int numericOctaveStep= (int)fifths-newstep;

return newstep;
}

```

```
public String operationMajorMinor(int newstep, int alter,int octave, char fifths){

    int difoctav = fifthsoctave - octave;
    int numericfifthsOctave=(int)fifths+(7*difoctav);
    int numericOctaveStep;

    numericOctaveStep=newstep-numericfifthsOctave;
    if (numericOctaveStep>0){
        if(alter>0){
            if (major[numericOctaveStep]==0.5f){
                numericOctaveStep++;
                return Integer.toString( numericOctaveStep);
            }
            else {
                return Integer.toString( numericOctaveStep)+"#";
            }
        }
        else {
            return Integer.toString( numericOctaveStep);
        }
    }

    return "false";
}
}
```

## LAMPIRAN B

### Wawancara Analisis Pengguna

# ANALISIS APLIKASI TRANSLITERASI NOT BALOK MENJADI NOT ANGKA BERBASIS ANDROID

---



Assalamu 'alaikum Wr. Wb

Peneliti mohon keterbukaan dari saudara/saudari dalam memberikan informasi untuk membantu kelancaran penelitian ini. Informasi atau keterangan yang saudara/saudari berikan akan dijaga kerahasiaannya dan murni digunakan hanya untuk keperluan akademis.

No. urut informan :

Waktu wawancara :

Informan : Sri Lutfiwati/ Conductor PSM Gita Savana 2012-2013

1. Apakah Anda familiar dengan not angka atau not balok?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami**
    - c. Sulit dibaca
  - II. Jika not balok, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca**
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca
  
2. Apakah yang Anda pikirkan setelah membaca teks berisi not balok dan not angka?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca

b. Mudah dibaca tapi sulit dipahami

c. Sulit dibaca

II. Jika not balok, menurut Anda not angka itu.... ?

a. Mudah dipahami dan dibaca

b. Mudah dibaca tapi sulit dipahami

c. Sulit dibaca

3. Apakah membaca not balok relatif lebih sulit? **Tidak**

Jika iya, mengapa?

.....

4. Apakah membaca not angka relatif lebih sulit? **Iya**

Jika iya, mengapa?

**Tinggi rendah dan naik turunnya nada tidak tergambar melalui simbol angka**

5. Apakah membaca not balok sulit dilakukan? **Tidak**

Jika iya, kesulitan seperti apa?

.....

6. Apakah membaca not balok (?) **angka po?** sulit dilakukan?

Jika iya, kesulitan seperti apa?

**Sulit membayangkan tinggi rendahnya nada**

7. Aplikasi apa yang Anda inginkan mengenai not angka atau not balok?

**Yang bisa menggambarkan tinggi rendah atau naik turunnya not angka.**

Terima kasih peneliti ucapkan atas partisipasi dan kerjasama dari saudara/saudari.

Semoga hasil penelitian ini bisa menjadi masukan yang dapat dimanfaatkan demi tercapainya tujuan pembuatan aplikasi ini.

Wassalamu 'alaikum Wr. Wb

Hormat peneliti,



## ANALISIS APLIKASI TRANSLITERASI NOT BALOK MENJADI NOT ANGKA BERBASIS ANDROID

---



Assalamu 'alaikum Wr. Wb

Peneliti mohon keterbukaan dari saudara/saudari dalam memberikan informasi untuk membantu kelancaran penelitian ini. Informasi atau keterangan yang saudara/saudari berikan akan dijaga kerahasiaannya dan murni digunakan hanya untuk keperluan akademis.

No. urut informan :

Waktu wawancara : 5 Januari 2013

Informan : Laila Marthatilova (anggota UKM PSM Gita Savana)

1. Apakah Anda familiar dengan not angka atau not balok?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. **Mudah dipahami dan dibaca**
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca
  - II. Jika not balok, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. **Mudah dibaca tapi sulit dipahami**
    - c. Sulit dibaca
  
2. Apakah yang Anda pikirkan setelah membaca teks berisi not balok dan not angka?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. **Mudah dipahami dan dibaca**
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca

II. Jika not balok, menurut Anda not angka itu.... ?

- a. Mudah dipahami dan dibaca
- b. Mudah dibaca tapi sulit dipahami**
- c. Sulit dibaca

3. Apakah membaca not balok relatif lebih sulit?

Jika iya, mengapa?

**Iya, karena ketika membaca not balok otak harus mentranslate kedalam nt angka dulu, dan banyak simbol yang kadang lupa artinya seperti tanda diam**

4. Apakah membaca not angka relatif lebih sulit?

Jika iya, mengapa?

**tidak**

5. Apakah membaca not balok sulit dilakukan?

Jika iya, kesulitan seperti apa?

**Harus mengerti dahulu arti simbol simbolnya..**

6. Apakah membaca not angka sulit dilakukan?

Jika iya, kesulitan seperti apa?

**Tidak, mungkin kesulitan bukan pada membacanya melainkan menembak nadanya.**

7. Aplikasi apa yang Anda inginkan mengenai not angka atau not balok?

**Aplikasi yang dapat mentranslate not balok kedalam not angka, sehingga mudah dibaca dan dipahami**

Terima kasih peneliti ucapkan atas partisipasi dan kerjasama dari saudara/saudari.

Semoga hasil penelitian ini bisa menjadi masukan yang dapat dimanfaatkan demi tercapainya tujuan pembuatan aplikasi ini.

Wassalamu 'alaikum Wr. Wb

Hormat peneliti,

## ANALISIS APLIKASI TRANSLITERASI NOT BALOK MENJADI NOT ANGKA BERBASIS ANDROID

---



Assalamu 'alaikum Wr. Wb

Peneliti mohon keterbukaan dari saudara/saudari dalam memberikan informasi untuk membantu kelancaran penelitian ini. Informasi atau keterangan yang saudara/saudari berikan akan dijaga kerahasiaannya dan murni digunakan hanya untuk keperluan akademis.

No. urut informan :

Waktu wawancara : 5 Januari 2013

Informan : Agustin Rahayuningsih (Mahasiswa)

1. Apakah Anda familiar dengan not angka atau not balok?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca
  - II. Jika not balok, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca
  
2. Apakah yang Anda pikirkan setelah membaca teks berisi not balok dan not angka?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca

II. Jika not balok, menurut Anda not angka itu.... ?

- a. Mudah dipahami dan dibaca
- b. Mudah dibaca tapi sulit dipahami
- c. Sulit dibaca

3. Apakah membaca not balok relatif lebih sulit?

Jika iya, mengapa? Iya

Karena sulit menghafal kuncinya

4. Apakah membaca not angka relatif lebih sulit?

Jika iya, mengapa?tidak

5. Apakah membaca not balok sulit dilakukan?

Jika iya, kesulitan seperti apa?YA

6. Apakah membaca not angka sulit dilakukan?

Jika iya, kesulitan seperti apa?Tidak

7. Aplikasi apa yang Anda inginkan mengenai not angka atau not balok?

Not angka karena mudah dipahami

Terima kasih peneliti ucapkan atas partisipasi dan kerjasama dari saudara/saudari.

Semoga hasil penelitian ini bisa menjadi masukan yang dapat dimanfaatkan demi tercapainya tujuan pembuatan aplikasi ini.

Wassalamu 'alaikum Wr. Wb

Hormat peneliti,

## ANALISIS APLIKASI TRANSLITERASI NOT BALOK MENJADI NOT ANGKA BERBASIS ANDROID

---



Assalamu 'alaikum Wr. Wb

Peneliti mohon keterbukaan dari saudara/saudari dalam memberikan informasi untuk membantu kelancaran penelitian ini. Informasi atau keterangan yang saudara/saudari berikan akan dijaga kerahasiaannya dan murni digunakan hanya untuk keperluan akademis.

No. urut informan :

Waktu wawancara : 5 Januari 2013

Informan : Tri Yunita (Mahasiswa)

1. Apakah Anda familiar dengan not angka atau not balok?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca
  - II. Jika not balok, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca
  
2. Apakah yang Anda pikirkan setelah membaca teks berisi not balok dan not angka?
  - I. Jika not angka, menurut Anda not angka itu.... ?
    - a. Mudah dipahami dan dibaca
    - b. Mudah dibaca tapi sulit dipahami
    - c. Sulit dibaca

II. Jika not balok, menurut Anda not angka itu.... ?

- a. Mudah dipahami dan dibaca
- b. Mudah dibaca tapi sulit dipahami
- c. Sulit dibaca

3. Apakah membaca not balok relatif lebih sulit?

Jika iya, mengapa? Iya

Karena tidak semua orang memahami lambang yang ada pada not balok

4. Apakah membaca not angka relatif lebih sulit?

Jika iya, mengapa? tidak

5. Apakah membaca not balok sulit dilakukan?

Jika iya, kesulitan seperti apa?Ya

6. Apakah membaca not angka sulit dilakukan?

Jika iya, kesulitan seperti apa?Tidak

7. Aplikasi apa yang Anda inginkan mengenai not angka atau not balok?

Untuk aplikasi di android lebih baik not angka , karena lebih mudah dipahami dan lebih familiar.

Terima kasih peneliti ucapkan atas partisipasi dan kerjasama dari saudara/saudari.

Semoga hasil penelitian ini bisa menjadi masukan yang dapat dimanfaatkan demi tercapainya tujuan pembuatan aplikasi ini.

Wassalamu 'alaikum Wr. Wb

Hormat peneliti,

LAMPIRAN C

File Pengujian Alpha

① happy b-day for keyboard stuff

Vibraphone

Marimba

Vib.

Mrm.

# ② Good Night Legend

Gooh neiglet ragelgh

Crichton meadows

Piano

Pno.

Pno.

Pno.



3

# Old MacDonald Had Some Notes in G

F. Phillips

Mi re doh in G doh

Traditional

The musical score consists of four staves, each labeled 'Voice' or 'Vo.' on the left. The music is written in G major (one sharp) and 4/4 time. The lyrics are: 'Old Mac Don - ald had some notes mi mi re re doh And these three notes had names like this mi mi re re doh. With a doh doh here and a doh doh there here a doh there a doh ev - 'ry where a doh doh Old Mac Don - ald had some notes mi mi re re doh.' Above the notes, there are solfège syllables: '1 1 1 5 6 6 5 3 3 2 2 1 0 5' for the first staff, and '1 5 6 6 5 3 3 2 2 1 0 5 5' for the second staff. The third and fourth staves do not have solfège syllables written above them.

④ tiup Lilinnya

[Composer]

i i i i 5 3 3 3 3 i 5 5 5 5 5 2 2 2 5 4 3 4.

Soprano

Alto

Tenor

Bass

The first system of the musical score consists of four staves: Soprano, Alto, Tenor, and Bass. The Soprano staff is in treble clef and contains a series of notes with a complex rhythmic pattern, accompanied by a sequence of numbers above it: 'i i i i 5 3 3 3 3 i 5 5 5 5 5 2 2 2 5 4 3 4.'. The Alto, Tenor, and Bass staves are in their respective clefs and contain corresponding musical notation for their parts.

S

A

T

B

The second system of the musical score consists of four staves: Soprano (S), Alto (A), Tenor (T), and Bass (B). The Soprano staff is in treble clef and contains a sequence of notes with numbers above it: '6 5 6 4 3 2 1'. The Alto, Tenor, and Bass staves are in their respective clefs and contain corresponding musical notation for their parts.

# Twinkle Twinkle Little Star

YuAn

A variation of Mozart's well known tune

Flute *Lively*  
*mf*

C Trumpet

C Trumpet

Piano

Organ

Fl.

C Tpt.

C Tpt.

Pno.

Org.

6

# You Are My Sunshine

Violoncello

Musical notation for Violoncello in 3/4 time. The piece is in G major. The notation includes fingerings: 6, 2, 3, 4, 4, 6, 2, 3, 2, 2, 6, 2, 3, 6, 1, 1, 2, 6, 5.

Vlc.

Empty musical staff for Violoncello.

Vlc.

Empty musical staff for Violoncello.

Empty musical staff for Violoncello.

Empty musical staff for Violoncello.

Empty musical staff for Violoncello.

Empty musical staff for Violoncello.

7

1

Sight singing exercise

Musical notation for exercise 7. The top staff is a vocal line in 4/4 time, G major, with lyrics: 1 2 3 4 5 6 5 | 1 1 7 6 5 6 5 | 1 2 3 4 5 6 5. The bottom staff is a flute line (Fl.) with notes: 4 | 5 | 3 | 1 | 2 | 1.

8

Listen - Sing

Musical notation for exercise 8. The top staff is a voice line (Voice) in 4/4 time, G major, with notes: G4, A4, B4, C5, B4, A4, G4, F4, E4, D4, C4. The bottom staff is a piano line (p.) with notes: G4, A4, B4, C5, B4, A4, G4, F4, E4, D4, C4.

9

Frere Jacques

Flute

Fl.

10

Marry Had a Little Lamb

11

### Trilili

Musical score for 'Trilili' in 4/4 time. The score consists of four staves. The first staff is the melody with a sequence of notes and fingerings: 1 2 3, 1 2 3, 1 3 5 5 4 3 2, 2 3 4. The second staff contains a bass line with notes and fingerings: 2 3 4, 5 4 3 3, 2 2 1, followed by rests. The third and fourth staves are empty, with a '0' written above each staff.

12

### Balonku

Arr by: Melissa Dharmawan

Musical score for 'Balonku' in 4/4 time, arranged by Melissa Dharmawan. The score is for Piano and includes three systems. The first system is labeled 'Piano' and includes a tempo marking of 'tempo = 80'. The second system is labeled 'Pno.' and the third is also labeled 'Pno.'. The score features complex rhythmic patterns and fingerings, including triplets and sixteenth notes. The notation includes various symbols such as 'x' and 'i' above notes, and '1', '2', '3', '4', '5', '6', '7' below notes.

②

Sholatullah

00 1 2 3. 3 2 00 1 1. 000 00 1 1 2 3. 3 2 00 <sup>A</sup>

1 1 000 4 4 4 4 3 0 5 4 3. 0 0 1 1 2 3. 5 2 00

1 1 000

Detailed description: This block contains three staves of musical notation. The first staff has a treble clef and a key signature of one sharp (F#). It contains a sequence of notes with fingerings (00, 1, 2, 3., 3, 2, 00, 1, 1., 000, 00, 1, 1, 2, 3., 3, 2, 00) and an 'A' above the final measure. The second staff has a treble clef and the same key signature, with notes and fingerings (1, 1, 000, 4, 4, 4, 4, 3, 0, 5, 4, 3., 0, 0, 1, 1, 2, 3., 5, 2, 00). The third staff has a treble clef and the same key signature, with notes and fingerings (1, 1, 000) followed by three empty measures.

⑭

diduniaku

<sup>Gita Gutawa</sup>  
0 3 3 4 5 5 3 5 5 3 3 5 5 6 2 0 3 4 5 5 3 5 5 6 2

Detailed description: This block contains two staves of musical notation. The first staff has a treble clef and a key signature of two sharps (F# and C#). It contains a sequence of notes with fingerings (0, 3, 3, 4, 5, 5, 3, 5, 5, 3, 3, 5, 5, 6, 2, 0, 3, 4, 5, 5, 3, 5, 5, 6, 2) and the name 'Gita Gutawa' above the final measure. The second staff has a treble clef and the same key signature, followed by three empty measures.

⑮

syukur

<sup>H. Mutahar</sup>  
6 6 6 i 7 6 0 5 3 3 2 i 7 2 i 7 0

Detailed description: This block contains one staff of musical notation. The staff has a treble clef and a key signature of two sharps (F# and C#). It contains a sequence of notes with fingerings (6, 6, 6, i, 7, 6, 0, 5, 3, 3, 2, i, 7, 2, i, 7, 0) and the name 'H. Mutahar' above the final measure.



**LAMPIRAN D**  
**Pengujian Beta**

**Kuesioner Pengujian Aplikasi Transliterasi Not Balok menjadi Not Angka**

(Pop up 123)

**UIN Sunan Kalijaga Yogyakarta**

Nama / Jurusan : Laila M / Anggota PSMI Tita Savana

Tahun Angkatan : 2012

**1. Tes Fungsionalitas Sistem :**

Berilah tanda Check (✓) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. **YA** jika data yang ditampilkan merupakan data yang sesuai untuk pernyataan tersebut, dan **TIDAK** jika data yang ditampilkan sangat tidak sesuai untuk pernyataan tersebut.

No.	Pernyataan	YA	TIDAK
1.	Aplikasi dapat menampilkan kotak dialog pilih file xml	✓	
2.	Aplikasi dapat mengecek file xml	✓	
3.	Pengguna dapat melihat judul lagu	✓	
4.	Pengguna dapat melihat penyanyi/singer	✓	
5.	Pengguna dapat melihat nada dasar	✓	
6.	Pengguna dapat melihat birama	✓	
7.	Aplikasi dapat mengkonversi tanda diam/istirahat/rest	✓	
8.	Aplikasi dapat mengkonversi step/not	✓	
9.	Aplikasi dapat mengkonversi alter/tanda kromatis	✓	
10.	Aplikasi dapat mengkonversi octave lagu	✓	
11.	Aplikasi dapat mengkonversi garis birama	✓	
12.	Pengguna dapat melakukan penyimpanan file kedalam dokumen	✓	
13.	Aplikasi menyediakan pilihan ubah nada dasar		✓
14.	Aplikasi dapat mengkonversi lirik lagu		✓
15.	Aplikasi dapat mengkonversi tanda legato		✓
16.	Aplikasi dapat mengkonversi nilai dan type not		✓
17.	Aplikasi dapat mengkonversi tanda tempo mutlak		✓
18.	Aplikasi dapat mengkonversi tanda tempo relatif		✓
19.	Aplikasi dapat mengkonversi arah tangkai not balok		✓
20.	Aplikasi dapat mengkonversi not balok ke not angka dengan cepat dan akurat	✓	

**2. Tes Usability Sistem :**

No.	Pernyataan	SS	S	N	TS	STS
1.	Menu-menu yang disediakan mudah dimengerti			✓		
2.	Tampilan aplikasi sederhana dan menarik		✓			
3.	Software dapat mempermudah proses konversi not balok ke not angka	✓				
4.	Software dapat mempermudah user untuk membaca dan menyanyikan not			✓		
5.	Software dapat memberi kemudahan dalam menyimpan hasil konversi dalam format yang umum	✓				
6.	Software dapat menyertakan lirik dalam proses konversi				✓	

**Ket.:** SS:Sangat Setuju, S:Setuju, N:Netral, TS:Tidak Setuju, STS:Sangat Tidak Setuju

**Kuesioner Pengujian Aplikasi Transliterasi Not Balok menjadi Not Angka**

**(Pop up 123)**

**UIN Sunan Kalijaga Yogyakarta**

Nama / Jurusan : PRATAMI P.R / T. INFORMATIKA

Tahun Angkatan : 2010

**1. Tes Fungsionalitas Sistem :**

Berilah tanda Check (✓) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. **YA** jika data yang ditampilkan merupakan data yang sesuai untuk pernyataan tersebut, dan **TIDAK** jika data yang ditampilkan sangat tidak sesuai untuk pernyataan tersebut.

No.	Pernyataan	YA	TIDAK
1.	Aplikasi dapat menampilkan kotak dialog pilih file xml	✓	
2.	Aplikasi dapat mengecek file xml	✓	
3.	Pengguna dapat melihat judul lagu	✓	
4.	Pengguna dapat melihat penyanyi/singer	✓	
5.	Pengguna dapat melihat nada dasar	✓	
6.	Pengguna dapat melihat birama	✓	
7.	Aplikasi dapat mengkonversi tanda diam/istirahat/rest	✓	
8.	Aplikasi dapat mengkonversi step/not	✓	
9.	Aplikasi dapat mengkonversi alter/tanda kromatis	✓	
10.	Aplikasi dapat mengkonversi octave lagu	✓	
11.	Aplikasi dapat mengkonversi garis birama	✓	
12.	Pengguna dapat melakukan penyimpanan file kedalam dokumen		✓
13.	Aplikasi menyediakan pilihan ubah nada dasar		✓
14.	Aplikasi dapat mengkonversi lirik lagu		✓
15.	Aplikasi dapat mengkonversi tanda legato		✓
16.	Aplikasi dapat mengkonversi nilai dan type not		✓
17.	Aplikasi dapat mengkonversi tanda tempo mutlak		✓
18.	Aplikasi dapat mengkonversi tanda tempo relatif		✓
19.	Aplikasi dapat mengkonversi arah tangkai not balok		✓
20.	Aplikasi dapat mengkonversi not balok ke not angka dengan cepat dan akurat	✓	

**2. Tes Usability Sistem :**

No.	Pernyataan	SS	S	N	TS	STS
1.	Menu-menu yang disediakan mudah dimengerti		✓			
2.	Tampilan aplikasi sederhana dan menarik		✓			
3.	Software dapat mempermudah proses konversi not baik ke not angka		✓			
4.	Software dapat mempermudah user untuk membaca dan menyanyikan not		✓			
5.	Software dapat memberi kemudahan dalam menyimpan hasil konversi dalam format yang umum				✓	
6.	Software dapat menyertakan lirik dalam proses konversi				✓	

**Ket.:** SS:Sangat Setuju, S:Setuju, N:Netral, TS:Tidak Setuju, STS:Sangat Tidak Setuju

**Kuesioner Pengujian Aplikasi Transliterasi Not Balok menjadi Not Angka**

**(Pop up 123)**

**UIN Sunan Kalijaga Yogyakarta**

Nama / Jurusan : *Dita Damayanti*

Tahun Angkatan : *Anggota Gita Sarana*

**1. Tes Fungsionalitas Sistem :**

Berilah tanda Check (✓) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. **YA** jika data yang ditampilkan merupakan data yang sesuai untuk pernyataan tersebut, dan **TIDAK** jika data yang ditampilkan sangat tidak sesuai untuk pernyataan tersebut.

No.	Pernyataan	YA	TIDAK
1.	Aplikasi dapat menampilkan kotak dialog pilih file xml	✓	
2.	Aplikasi dapat mengecek file xml	✓	
3.	Pengguna dapat melihat judul lagu	✓	
4.	Pengguna dapat melihat penyanyi/singer	✓	
5.	Pengguna dapat melihat nada dasar	✓	
6.	Pengguna dapat melihat birama	✓	
7.	Aplikasi dapat mengkonversi tanda diam/istirahat/rest		✓
8.	Aplikasi dapat mengkonversi step/not	✓	
9.	Aplikasi dapat mengkonversi alter/tanda kromatis		✓
10.	Aplikasi dapat mengkonversi octave lagu	✓	
11.	Aplikasi dapat mengkonversi garis birama	✓	
12.	Pengguna dapat melakukan penyimpanan file kedalam dokumen	✓	
13.	Aplikasi menyediakan pilihan ubah nada dasar		✓
14.	Aplikasi dapat mengkonversi lirik lagu		✓
15.	Aplikasi dapat mengkonversi tanda legato		✓
16.	Aplikasi dapat mengkonversi nilai dan type not		✓
17.	Aplikasi dapat mengkonversi tanda tempo mutlak		✓
18.	Aplikasi dapat mengkonversi tanda tempo relatif		✓
19.	Aplikasi dapat mengkonversi arah tangkai not balok		✓
20.	Aplikasi dapat mengkonversi not balok ke not angka dengan cepat dan akurat	✓	

**2. Tes Usability Sistem :**

No.	Pernyataan	SS	S	N	TS	STS
1.	Menu-menu yang disediakan mudah dimengerti		✓			
2.	Tampilan aplikasi sederhana dan menarik		✓			
3.	Software dapat mempermudah proses konversi not baik ke not angka		✓			
4.	Software dapat mempermudah user untuk membaca dan menyanyikan not		✓			
5.	Software dapat memberi kemudahan dalam menyimpan hasil konversi dalam format yang umum				✓	
6.	Software dapat menyertakan lirik dalam proses konversi				✓	

**Ket.:** SS:Sangat Setuju, S:Setuju, N:Netral, TS:Tidak Setuju, STS:Sangat Tidak Setuju

**Kuesioner Pengujian Aplikasi Transliterasi Not Balok menjadi Not Angka**

**(Pop up 123)**

**UIN Sunan Kalijaga Yogyakarta**

Nama / Jurusan : Ahmad Imadudin / MTI

Tahun Angkatan : 2013

**1. Tes Fungsionalitas Sistem :**

Berilah tanda Check (✓) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. YA jika data yang ditampilkan merupakan data yang sesuai untuk pernyataan tersebut, dan **TIDAK** jika data yang ditampilkan sangat tidak sesuai untuk pernyataan tersebut.

No.	Pernyataan	YA	TIDAK
1.	Aplikasi dapat menampilkan kotak dialog pilih file xml	✓	
2.	Aplikasi dapat mengecek file xml	✓	
3.	Pengguna dapat melihat judul lagu	✓	
4.	Pengguna dapat melihat penyanyi/singer	✓	
5.	Pengguna dapat melihat nada dasar	✓	
6.	Pengguna dapat melihat birama	✓	
7.	Aplikasi dapat mengkonversi tanda diam/istirahat/rest	✓	
8.	Aplikasi dapat mengkonversi step/not	✓	
9.	Aplikasi dapat mengkonversi alter/tanda kromatis	✓	
10.	Aplikasi dapat mengkonversi octave lagu	✓	
11.	Aplikasi dapat mengkonversi garis birama	✓	
12.	Pengguna dapat melakukan penyimpanan file kedalam dokumen		✓
13.	Aplikasi menyediakan pilihan ubah nada dasar		✓
14.	Aplikasi dapat mengkonversi lirik lagu		✓
15.	Aplikasi dapat mengkonversi tanda legato		✓
16.	Aplikasi dapat mengkonversi nilai dan type not		✓
17.	Aplikasi dapat mengkonversi tanda tempo mutlak		✓
18.	Aplikasi dapat mengkonversi tanda tempo relatif		✓
19.	Aplikasi dapat mengkonversi arah tangkai not balok		✓
20.	Aplikasi dapat mengkonversi not balok ke not angka dengan cepat dan akurat	✓	



**2. Tes Usability Sistem :**

No.	Pernyataan	SS	S	N	TS	STS
1.	Menu-menu yang disediakan mudah dimengerti		✓			
2.	Tampilan aplikasi sederhana dan menarik		✓			
3.	Software dapat mempermudah proses konversi not baiok ke not angka		✓			
4.	Software dapat mempermudah user untuk membaca dan menyanyikan not		✓			
5.	Software dapat memberi kemudahan dalam menyimpan hasil konversi dalam format yang umum				✓	
6.	Software dapat menyertakan lirik dalam proses konversi				✓	

**Ket.:** SS:Sangat Setuju, S:Setuju, N:Netral, TS:Tidak Setuju, STS:Sangat Tidak Setuju

**Kuesioner Pengujian Aplikasi Transliterasi Not Balok menjadi Not Angka**

(Pop up 123)

**UIN Sunan Kalijaga Yogyakarta**

Nama / Jurusan : *Lusy Rizkya M (Pelajar)*

Tahun Angkatan :

**1. Tes Fungsionalitas Sistem :**

Berilah tanda Check (✓) pada pilihan yang menurut anda paling sesuai menggambarkan fungsi sistem. **YA** jika data yang ditampilkan merupakan data yang sesuai untuk pernyataan tersebut, dan **TIDAK** jika data yang ditampilkan sangat tidak sesuai untuk pernyataan tersebut.

No.	Pernyataan	YA	TIDAK
1.	Aplikasi dapat menampilkan kotak dialog pilih file xml	✓	
2.	Aplikasi dapat mengecek file xml	✓	
3.	Pengguna dapat melihat judul lagu	✓	
4.	Pengguna dapat melihat penyanyi/singer	✓	
5.	Pengguna dapat melihat nada dasar	✓	
6.	Pengguna dapat melihat birama	✓	
7.	Aplikasi dapat mengkonversi tanda diam/istirahat/rest	✓	
8.	Aplikasi dapat mengkonversi step/not	✓	
9.	Aplikasi dapat mengkonversi alter/tanda kromatis	✓	
10.	Aplikasi dapat mengkonversi octave lagu	✓	
11.	Aplikasi dapat mengkonversi garis birama	✓	
12.	Pengguna dapat melakukan penyimpanan file kedalam dokumen		
13.	Aplikasi menyediakan pilihan ubah nada dasar		✓
14.	Aplikasi dapat mengkonversi lirik lagu		✓
15.	Aplikasi dapat mengkonversi tanda legato		✓
16.	Aplikasi dapat mengkonversi nilai dan type not		✓
17.	Aplikasi dapat mengkonversi tanda tempo mutlak		✓
18.	Aplikasi dapat mengkonversi tanda tempo relatif		✓
19.	Aplikasi dapat mengkonversi arah tangkai not balok		✓
20.	Aplikasi dapat mengkonversi not balok ke not angka dengan cepat dan akurat	✓	

**2. Tes Usability Sistem :**

No.	Pernyataan	SS	S	N	TS	STS
1.	Menu-menu yang disediakan mudah dimengerti	✓				
2.	Tampilan aplikasi sederhana dan menarik	✓				
3.	Software dapat mempermudah proses konversi not balok ke not angka	✓				
4.	Software dapat mempermudah user untuk membaca dan menyanyikan not	✓				
5.	Software dapat memberi kemudahan dalam menyimpan hasil konversi dalam format yang umum	✓				
6.	Software dapat menyertakan lirik dalam proses konversi				✓	

**Ket.:** SS:Sangat Setuju, S:Setuju, N:Netral, TS:Tidak Setuju, STS:Sangat Tidak Setuju