

APLIKASI PANDUAN LAGU-LAGU WAJIB

BERBASIS ANDROID

NASKAH PUBLIKASI



diajukan oleh

GELLA NOERCINTA

10.11.4051

kepada

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER

AMIKOM YOGYAKARTA

YOGYAKARTA

2014

NASKAH PUBLIKASI

APLIKASI PANDUAN LAGU-LAGU WAJIB
BERBASIS ANDROID

disusun oleh

Gella Noercinta
10.11.4051


Dosen Pembimbing,



Heri Siamoto, M.Kom
NIK. 190302057

Tanggal 18 September 2013

Ketua Jurusan
Teknik Informatika



Sudarmawan, MT
NIK. 190302035

APPLICATION GUIDE NATIONAL ANTHEMS ANDROID BASED
APLIKASI PANDUAN LAGU-LAGU WAJIB BERBASIS ANDROID

Gella Noercinta
Heri Sismoro
Jurusan Teknik Informatika
STMIK AMIKOM YOGYAKARTA

ABSTRACT

Technology is growing very rapidly. Humans use it to find information, share information, and communicate. As well as interactive learning media.

Understanding the intent anthem here, that the songs that shall be studied, understood, and internalized the meaning and contents by all youth and students in Indonesia. The way, anyone on youth and student, and must be able to sing and lead the songs as well as possible.

Application Guide National Anthems will have several features such as information about lyrics of the song. This application is expected later to help users of Android based mobile devices, especially the Student / Elementary School Students who want to learn about the national anthems. This application will be made using Eclipse and Adobe Photoshop CS4. Eclipse is used to create Android applications using the Java language.

Keywords: *Android, Guide, Eclipse, Java, Mobile, Application Guide National Anthems.*

1. Pendahuluan

Dengan berkembangnya teknologi, Manusia menggunakannya untuk mencari informasi, membagi informasi, berkomunikasi, dan juga sebagai media belajar yang interaktif. Semua kalangan umum bisa menggunakan teknologi terbaru. Hal itu di karenakan banyak *vendor* yang menawarkan *gadget* yang murah yang bisa di jangkau oleh semua kalangan. Sehingga semua orang bisa memiliki sebuah *gadget* dengan teknologi yang sudah maju tidak terkecuali anak-anak kecil.

Begitu juga dengan musik, musik untuk saat ini semakin berkembang. Penikmatnya pun tidak memiliki batasan usia, mulai dari anak yang masih duduk di sekolah dasar sampai kepada orang tua. Namun sangat disayangkan dengan banyaknya lagu yang bermunculan saat ini kurang di imbangi dengan pendidikan yang tepat untuk anak kecil. Dengan menjamurnya boy band dan girl band ini membuat banyak anak-anak kecil sangat fasih menirukan gaya dan menyanyikan lagu-lagu dari beberapa *girl band* dan *boy band* idola Mereka, baik di dalam maupun Luar Negeri. Hal ini sangat di sayangkan karena masih banyak juga anak-anak Indonesia yang tidak mengerti ataupun hafal dengan lagu wajib dari Negeri Mereka sendiri. Seperti : Bagimu Negeri, Bendera Merah Putih, Garuda Pancasila, Hari Merdeka, Indonesia Raya, dan lain sebagainya.

Dengan dukungan perkembangan teknologi sebagai media belajar yang efektif, teknologi bisa membuat anak-anak tertarik untuk mempelajari dan memahami lebih dalam lagu-lagu wajib dari tanah kelahiran Mereka sendiri. Ini akan menjadi lebih efisien karena bisa dilakukan dimana saja dan kapan saja.

2. Landasan Teori

2.1 Sistem Operasi Android

2.1.1 Sejarah Android

Android merupakan sistem operasi berbasis Linux yang dirancang untuk perangkat selular layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini diliris secara resmi pada tahun 2007, bersamaan dengan didirikannya *Open Handset Alliance*, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler.

2.1.2 Struktur Android

Android merupakan sistem operasi *mobile* yang memiliki berbagai macam fitur didalamnya, mulai dari *gps* sampai *accelerometer*.

Menurut Safaat (2012:6) secara garis besar Arsitektur Android dapat dijelaskan sebagai berikut :

1. *Application and Widgets*

Application dan *widgets* ini adalah *layer* di mana pengguna berhubungan dengan aplikasi saja, di mana biasanya pengguna men-*download* aplikasi kemudian kita lakukan instalasi dan menjalankan aplikasi tersebut. Di *layer* terdapat aplikasi inti termasuk *client email*, program SMS, kalender, peta, *browse*, kontak dan lain-lain. Semua aplikasi dibuat menggunakan bahasa Java.

2. *Applications Frameworks*

Android adalah "*Open Development Platform*" yaitu Android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk mengembangkan aplikasi yang bagus dan inovatif. Pengembang memiliki akses penuh menuju *API framework* seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan.

3. *Libraries*

Libraries ini adalah *layer* di mana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan diatas kernel, *layer* ini meliputi berbagai *library C/C++* inti seperti *Libc* dan *SSL*. Kemampuan ini dapat diakses oleh *programmer* melewati *Android application framework*. Sebagai contoh Android mendukung pemutaran format audio, video, dan gambar.

4. *Android Run Time*

Layer yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan implementasi Linux. *Dalvik Virtual Machine* (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam *Android Run Time* dibagi menjadi dua bagian yaitu: *Core libraries* dan *Dalvik Virtual Machine*

5. *Linux Kernel*

Tumpukan paling bawah pada arsitektur Android ini adalah *kernel*. Linux *kernel* yang digunakan Android adalah linux kernel *release 2.6*. *Kernel* berperan sebagai abstraction layer antara *hardware* dan keseluruhan *software*. Namun secara keseluruhan Android bukanlah linux, karena dalam Android tidak terdapat paket standar yang dimiliki oleh linux lainnya.

2.1.3 Versi Android

- Android 1.0
- Android 1.1
- Android Cupcake (1.5)
- Android Donut (1.6)
- Android Éclair (2.0 – 2.1)
- Android Froyo (2.2 – 2.2.3)
- Android Gingerbread (2.3–2.3.7)
- Android Honeycomb (3.0–3.2.6)
- Android Ice Cream Sandwich (4.0–4.0.4)
- Android Jelly Bean (4.1–4.3)
- Android KitKat (4.4)

2.2 UML (Unified Modeling Language)

2.2.1 Pengertian UML (Unified Modeling Language)

Unified Modelling Language (UML) menurut Hermawan (2004:7) adalah bahasa standar yang digunakan untuk menjelaskan dan memvisualisasikan artifak dari proses analisis dan desain sistem berorientasi obyek. UML menyediakan standar pada notasi dan diagram yang bisa digunakan untuk memodelkan suatu system. UML dikembangkan oleh 3 pendekar “berorientasi objek”, yaitu Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. UML menjadi bahasa yang bisa digunakan untuk berkomunikasi dalam perspektif obyek antara user dan developer analisis dengan developer desain, dan antara developer desain dengan developer pemrograman.

2.2.2 Notasi dan Artifak dalam UML (Unified Modeling Language)

- Actor
- Class
- Interface
- Use case
- Interaction
- Package
- Note
- Dependency
- Association
- Generalization
- Realization
- Use case diagram
- Sequence diagram

- Class diagram

2.3 Perangkat Lunak Yang Digunakan (Software)

2.3.1 Eclipse

Eclipse adalah sebuah *IDE (Integrated Development Environment)* untuk mengembangkan perangkat lunak dan dapat dijalankan di semua *platform (platform-independent)*.

2.3.2 Bahasa Pemrograman Java

Java adalah bahasa pemrograman serbaguna. Java dapat digunakan untuk membuat suatu program sebagaimana Anda membuatnya dengan bahasa seperti Pascal atau C++. Java juga mendukung sumber daya internet yang saat ini populer, yaitu World Wide Web atau yang sering disebut Web saja. Java juga mendukung aplikasi client/server, baik dalam jaringan local (LAN) maupun jaringan berskala luas (WAN).

2.3.3 Photosop

Adobe Photoshop atau biasa disebut *Photoshop*, adalah perangkat lunak editor citra buatan *Adobe Systems* yang dikhususkan untuk pengeditan foto/gambar dan pembuatan efek. Perangkat lunak ini banyak digunakan oleh fotografer digital dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolah gambar/foto, dan, bersama *Adobe Acrobat*, dianggap sebagai produk terbaik yang pernah diproduksi oleh *Adobe Systems*. Versi kedelapan aplikasi ini disebut dengan nama *Photoshop CS (Creative Suite)*, versi sembilan disebut *Adobe Photoshop CS2*, versi sepuluh disebut *Adobe Photoshop CS3*, versi kesebelas adalah *Adobe Photoshop CS4*, versi keduabelas adalah *Adobe Photoshop CS5*, dan versi yang terakhir (ketigabelas) adalah *Adobe Photoshop CS6*.

3. Analisis

3.1 Analisis Kebutuhan Sistem

Analisis sistem menurut Jogiyanto (2005:129) adalah penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan.

3.1.1 Kebutuhan Fungsional

Kebutuhan fungsional berisi proses-proses apa saja yang dapat dilakukan aplikasi panduan lagu-lagu wajib, antara lain sebagai berikut :

1. Aplikasi ini dapat memainkan Lagu-Lagu wajib Nasional yang terdapat di Indonesia dan wajib dipahami oleh seluruh warga Indonesia.
2. Aplikasi ini dapat menampilkan informasi tentang Lagu-Lagu Wajib yang terdapat dalam aplikasi.

3.1.2 Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras / *hardware*, analisis perangkat lunak / *software*, analisis sumber daya manusia / *brainware*.

3.2 Analisis SWOT

3.2.1 Strength (Kekuatan)

Menurut Steven John Metsker (2006) *Design Pattern* adalah sebuah class dan method untuk menyelesaikan permasalahan secara umum dalam pembuatan aplikasi secara berorientasi objek. Dikarenakan android merupakan sistem operasi yang masih terus berkembang, maka dengan adanya *design pattern* ini para pengembang akan dimudahkan dalam pembuatan aplikasi android.

3.2.2 Weakness (Kelemahan)

Aplikasi ini dirancang hanya sebagai salah satu media pembelajaran untuk anak-anak agar lebih tertarik dengan lagu-lagu wajib. Bukan sebagai aplikasi yang dapat di *update* sewaktu-waktu. Tampilannya juga di *design* agar mudah digunakan untuk anak-anak.

3.2.3 Opportunities (Peluang)

Aplikasi ini dapat berjalan dalam beberapa *device android* tanpa perlu membuat dalam berbagai versi android satu persatu.

3.2.4 Threats (Ancaman)

Android merupakan sistem operasi yang masih akan terus berkembang, maka akan semakin banyak aplikasi yang jauh lebih menarik dan lebih baik yang akan muncul.

3.3 Analisis Kelayakan Sistem

Sebuah sistem baru sangat membutuhkan pertimbangan dari beberapa unsur menentukan layak atau tidaknya suatu sistem yang baru sebagai alternatif sistem yang sedang berjalan.

3.3.1 Kelayakan Teknologi (Technical Feasibility)

Android adalah sistem operasi *mobile* yang digunakan pada *smartphone*. Dan di zaman yang *modern* ini pengguna *smartphone* juga semakin banyak dan tidak memandang batas usia. Perangkat ini pun semakin berkembang karena sifatnya yang *open source*.

3.3.2 Kelayakan Hukum (Law Feasibility)

Aplikasi ini dibuat dengan menggunakan *software* yang bersifat *open source* maka tidak melanggar lisensi.

3.3.3 Kelayakan Operasional)

Analisis kelayakan operasional ini akan dijadikan penilaian atas kelayakan operasi suatu sistem, yang berarti kemampuan dalam mengoperasikan sistem karena sistem baru di *design* sedemikian rupa sehingga cukup mudah untuk digunakan.

3.4 Perancangan Sistem dengan UML (Unified Modeling Language)

UML menjadi bahasa yang bisa digunakan untuk berkomunikasi dalam perspektif obyek antara user dan developer analisis dengan developer desain, dan antara developer desain dengan developer pemrograman.

3.4.1 Use Case Diagram

Use Case Diagram menyajikan informasi antara *use case* dan *actor*.

3.4.2 Activity Diagram

Activity diagram memodelkan alur kerja dan urutan kegiatan dalam sebuah sistem. Diagram ini membantu dalam memahami proses kerja secara keseluruhan.

3.4.3 Class Diagram

Class Diagram memperlihatkan hubungan antara kelas satu dengan kelas yang lain dan penjelasan detail mengenai struktur sistem dan menampilkan atribut, operasi beserta *method* yang digunakan.

3.4.4 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario.

3.5 Perancangan User Interface

User interface merupakan interaksi antara komputer atau sebuah perangkat dengan pengguna. Tampilan antar muka sangat berpengaruh terhadap kemudahan user dalam menggunakan aplikasi. Berikut adalah perancangan *user interface* aplikasi Panduan Lagu-Lagu Wajib.

3.5.1 Halaman Splash

Halaman ini adalah halaman awal yang akan dilihat oleh pengguna sebelum masuk ke halaman menu utama.

3.5.2 Halaman Menu Utama

Halaman ini adalah halaman menu utama dimana terdapat tombol *Playlist*, *Credit*, dan *Exit*.

3.5.3 Halaman Menu Credit

Halaman ini adalah halaman yang terdapat pada *menu credit* untuk menampilkan informasi aplikasi.

3.5.4 Halaman Menu Pilih Playlist

Halaman ini adalah halaman yang terdapat pada menu pilihan *playlist* yang akan menampilkan judul-judul lagu wajib yang terdapat dalam aplikasi.

3.5.5 Halaman Lagu Pilihan

Halaman ini adalah halaman yang berisi lirik lagu pilihan yang sudah dipilih oleh pengguna, pada halaman ini terdapat tombol-tombol yang dapat ditekan oleh pengguna sehingga mengeluarkan suara dari lagu yang dipilih, selain itu juga terdapat informasi dari lagu tersebut.

3.5.6 Halaman Info Lagu

Halaman ini adalah halaman untuk menampilkan informasi dari lagu yang dipilih oleh pengguna.

4. Implementasi dan Pembahasan

4.1 Implementasi

Tahap implementasi sistem adalah tahap meletakkan suatu sistem agar siap untuk dioperasikan. Tahap ini dilakukan setelah melakukan perancangan sistem yaitu pembuatan program.

4.1.1 Implementasi Pembuatan Interface

Dalam aplikasi "Lagu-Lagu Wajib" ini terdiri dari beberapa *interface* yaitu *Splash Screen*, Menu Utama, Menu *Credit*, Menu Pilih *Playlist*, Lagu Pilihan, dan Info Lagu.

4.2 Pembahasan

4.2.1 Uji Coba Sistem dan Program

Uji coba sistem merupakan proses mengeksekusi sistem perangkat lunak untuk menentukan apakah sistem perangkat lunak tersebut dapat berjalan baik atau tidak.

Pengujian program dilakukan untuk memeriksa kekompakan antara komponen sistem yang diimplementasikan. Tujuan utama dari pengujian program ini adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari program telah berfungsi sesuai dengan yang diharapkan. Pengujian perlu dilakukan untuk mencari kesalahan-kesalahan yang mungkin terjadi.

4.2.2 Pengujian Sistem

Metode yang digunakan untuk pengujian meliputi *White Box Testing* dan *Black Box Testing*.

1. White Box Testing

White Box Testing adalah cara pengujian dengan melihat ke dalam modul untuk meneliti kode-kode program yang ada, dan menganalisis adanya kesalahan. *White Box Testing* digunakan untuk menguji kerja perangkat lunak secara rinci dan dilakukan selama pembuatan program.

2. *Black Box Testing*

Black Box Testing merupakan *test-case* perangkat lunak berdasarkan kebutuhan (*requirement*) yang ada pada spesifikasi sistem, inputan *data testing* diharapkan bisa menemukan *output* yang masih terdapat kesalahan. Adapun cara pengujian *Black Box Testing* adalah dengan cara menjalankan modul sistem secara keseluruhan, kemudian diamati apakah hasil data yang masuk sudah sesuai dengan data yang diharapkan dari tahap perancangan atau belum.

4.2.3 **Pengujian Program**

Sebelum aplikasi digunakan oleh *user* secara langsung maka sebelumnya program harus menjalani serangkaian tes untuk menguji apakah ada kesalahan-kesalahan yang terdapat pada program tersebut. Pengetesan program ini juga bisa memberikan hasil mengenai kemudahan dalam pengoperasian aplikasi.

4.2.4 **Manual Program**

Aplikasi ini dapat berjalan dengan baik dengan salah satu faktornya adalah kemudahan menjalankan aplikasi. Berikut ini penjelasan sistem yang dibuat.

4.2.5 **Pembahasan Program**

Penulisan program bertujuan untuk mengimplementasikan hasil rancangan yang telah dibuat dengan menuliskan perintah menggunakan bahasa pemrograman yang dipilih. Langkah ini merupakan salah satu langkah akhir dari tahapan implementasi dari rancangan yang telah dibuat.

5. **Penutup**

5.1 **Kesimpulan**

Berdasarkan uraian-uraian yang telah penulis jelaskan pada bab-bab terdahulu dan pembuatan aplikasi, maka penulis dapat mengambil kesimpulan sebagai berikut :

1. Aplikasi Panduan Lagu-Lagu Wajib ini mudah dipergunakan oleh *User*.
2. Aplikasi Panduan Lagu-Lagu Wajib dibuat untuk mempermudah anak mempelajari Lagu-Lagu Wajib Nasional dimana saja dan kapan saja.
3. Aplikasi Panduan Lagu-Lagu Wajib dibuat untuk membantu orang tua untuk memperkenalkan dan menjelaskan Lagu-Lagu Wajib Nasional tanpa membuat anak bosan, karena aplikasi ini di *design* dengan menggabungkan unsur gambar *full color*, *text*, dan suara untuk menarik minat anak untuk mempelajari Lagu-Lagu Wajib.

5.2 **Saran**

Dalam pembuatan aplikasi ini masih memiliki berbagai keterbatasan. Aplikasi ini dirancang hanya sebagai salah satu media pembelajaran untuk anak-anak, bukan sebagai aplikasi yang dapat di *update* sewaktu-waktu. Karena *Android* merupakan sistem

operasi yang masih akan terus berkembang, maka akan semakin banyak aplikasi untuk media pembelajaran anak-anak yang jauh lebih menarik dan lebih baik yang akan muncul, namun tampilannya harus tetap sederhana atau sesuai untuk anak-anak.

DAFTAR PUSTAKA

Jogiyanto. 1990. *Analisis dan Desain Sistem Informasi: pendekatan terstruktur teori dan praktik aplikasi bisnis*. Yogyakarta : Andi Offset

http://en.wikipedia.org/wiki/Open_Handset_Alliance / diakses pada tanggal 24 Oktober 2013

<http://developer.android.com/about/version/index.html> / diakses pada tanggal 12 Oktober 2013 jam 22.00

<http://img.gawkerassets.com/img/17makhqclwirojpg/original.jpg> / diakses pada tanggal 27 Oktober jam 23.00

<http://gizmodo.com/5147830/t+mobile-g1-android-firmware-update-11-now-available> / diakses pada tanggal 27 Oktober jam 23.00

<http://android-developers.blogspot.com/2009/04/ui-framework-changes-in-android-15.html> diakses tanggal 27 Oktober jam 23.00

http://user-agent-string.info/pub/img/os_screenshots/android16.jpg / diakses pada tanggal 27 Oktober jam 23.00

<http://smarterware.org/wp-content/uploads/2010/01/android21-homescreen-180x300.png> / diakses pada tanggal 27 Oktober jam 23.00

<http://www.buzzingup.com/wp-content/uploads/2010/12/Android-gingerbread-home.png> / diakses pada tanggal 27 Oktober jam 23.00

<http://developer.android.com/sdk/images/2.2/22home.png> / diakses pada tanggal 27 Oktober jam 23.00

http://developer.android.com/sdk/images/3.0/home_hero1.png / diakses pada tanggal 27 Oktober jam 23.00

<http://www.smartkeitai.com/wp-content/uploads/2011/10/android-4.0-ice-cream-sandwich.jpg?00d448> / diakses pada tanggal 27 Oktober jam 23.00

[http://upload.wikimedia.org/wikipedia/en/6/67/Android_4.3_on_Nexus_7_%282013%29.jpg /](http://upload.wikimedia.org/wikipedia/en/6/67/Android_4.3_on_Nexus_7_%282013%29.jpg/)
diakses pada tanggal 27 Oktober jam 23.00

http://id.wikipedia.org/wiki/Daftar_versi_Android / diakses pada tanggal 19
oktober 2013

Hermawan, Julius. (2004). "*ANALISA DESAIN & Pemrograman Berorientasi Obyek dengan UML dan VISUAL BASIC.NET*" Andi Publisher.

Safaat H, Nazruddin. (2012). "*ANDROID Pemrograman Aplikasi Mobile Smartphone Dan Tablet PC Berbasis Android*" Informatika Bandung

Kadir, Abdul. (2004). "*Dasar Pemrograman JAVA 2*" Andi Publisher