

APLIKASI PEMBELAJARAN ANTI KORUPSI BAGI ANAK REMAJA BERBASIS ANDROID

Miswanto¹, Mohammad Badrul²
STMIK Nusa Mandiri Jakarta¹, miswanto24@gmail.com
STMIK Nusa Mandiri Jakarta², mohammad.mbl@nusamandiri.ac.id

ABSTRACT

Corruption issue is currently the focus of the Indonesian government to be able to be eradicated down until the roots. One way must to do is to give a learning and understanding to the public about the meaning and scope about corruption as well as preventive measures through the media about the corruption-based mobile learning to the public. With the mobile media-based learning allow people, especially the youth to be able to learn directly from a smartphone whenever and wherever they wanted to learn. With the participation of the public (especially teenagers) can create a good critical of the younger generation in conveying criticism and suggestions to the problems of corruption to the Indonesian government or law enforcement related to Corruption. One role of the writer to be able to contribute to the eradication of corruption, in this case in the form of preventing corruption is to design a system for the application of the Anti-Corruption Learning-Based Android for Teenagers, where it is expected that these applications can provide a basic learning and understanding of the associated problems of corruption.

Keywords : *learning anti-corruption, anti-corruption, anti-corruption android*

ABSTRAK

Permasalahan korupsi saat ini sedang menjadi fokus dari pemerintah Indonesia untuk dapat diberantas hingga ke akar-akarnya. Salah satu cara yang perlu dilakukan adalah memberi pembelajaran dan pemahaman kepada masyarakat mengenai pengertian dan ruang lingkup korupsi serta tindakan pencegahan melalui media pembelajaran tentang korupsi berbasis *mobile* kepada masyarakat. Dengan adanya media pembelajaran berbasis *mobile* memudahkan masyarakat khususnya para remaja untuk dapat belajar secara langsung dari *smartphone* yang dimiliki kapanpun dan dimanapun para remaja ingin belajar. Dengan adanya peran serta masyarakat (khususnya para remaja) dapat menciptakan para generasi muda yang kritis dalam menyampaikan kritik dan saran terhadap permasalahan korupsi kepada pemerintah Indonesia atau para penegak hukum yang berhubungan dengan Tindak Pidana Korupsi. Salah satu peran serta penulis untuk dapat berkontribusi dalam pemberantasan korupsi, dalam hal ini dalam bentuk pencegahan korupsi adalah dengan merancang sistem aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android, dimana diharapkan aplikasi ini dapat memberikan pemahaman dasar terkait masalah korupsi.

Kata Kunci: pembelajaran anti korupsi, anti korupsi, android anti korupsi

PENDAHULUAN

Permasalahan tentang Korupsi saat ini menjadi sorotan dari masyarakat pada umumnya, dan juga pemerintah pada khususnya. Dimana hal tersebut di ekspos melalui media cetak dan juga media elektronik baik secara langsung maupun tidak langsung. Sejak di dirikannya Komisi Pemberantasan Korupsi (KPK) sebagai Lembaga Independen Pemberantasan

Tindak Pidana Korupsi, sepekan terjang pemberantasan korupsi di Indonesia mengalami perubahan yang signifikan ke arah yang lebih baik.

Permasalahan korupsi saat ini tidak terlepas dari semakin berkembangnya teknologi informasi yang ada [1]. Dengan perkembangan dunia teknologi informasi yang semakin canggih, ada yang memanfaatkan teknologi dengan baik

sebagai media pembelajaran dan media informasi berbasis *mobile*. Dan ada juga yang memanfaatkan dengan tidak baik yaitu dengan memanfaatkan celah keamanan untuk melakukan tindakan pidana, salah satunya adalah tindak pidana korupsi.

Dalam hal materi pembelajaran atau informasi, di *Google Playstore* telah menyediakan ribuan aplikasi pembelajaran dan informasi, baik yang bersifat *free* (gratis) maupun yang bersifat *shareware* (bayar)[4]. Dari aplikasi tersebut, diharapkan masyarakat pada umumnya, dan penulis pada khususnya dapat memperoleh informasi baik secara langsung maupun tidak langsung mengenai suatu hal atau masalah. Hal ini agar kita tidak tertinggal isu atau informasi yang saat ini sedang berkembang. Karena saat ini segala bentuk materi atau informasi dapat dikemas sedemikian rupa dalam bentuk sebuah aplikasi *Mobile*.

Berdasarkan perkembangan aplikasi tersebut, dibutuhkan peran serta masyarakat untuk ikut berkontribusi dan turut membantu dalam pemberantasan Tindak Pidana Korupsi mulai dari diri sendiri, dan lingkungan masyarakat sekitar[3]. Dengan dibuatnya sistem aplikasi ini, diharapkan masyarakat khususnya para remaja dapat menggunakan aplikasi ini sebagai bahan pembelajaran dan pemahaman mengenai sikap Anti Korupsi, guna membentuk generasi muda yang lebih baik.

KAJIAN LITERATUR

Android adalah “sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka”[5]. Sehingga perkembangan versi android saat ini berkembang dengan pesat, baik versi resmi maupun versi modifikasi (*custom*) yang pada intinya menjalankan aplikasi android dengan baik.

Android pada mulanya perusahaan kecil yang berada di Palo Alto. Pada Juli 2000, Google bekerjasama dengan Android Inc., perusahaan yang berada di Palo Alto, California Amerika Serikat. Para pendiri

Android Inc. bekerja pada Google, di antaranya Andy Rubin, Rich Miner, Nick Sears, dan Chris White[5].

Saat itu banyak yang menganggap fungsi Android Inc. hanyalah sebagai perangkat lunak pada telepon seluler. Sejak saat itu muncul rumor bahwa Google hendak memasuki pasar telepon seluler. Di perusahaan Google, tim yang dipimpin Rubin bertugas mengembangkan program perangkat seluler yang didukung oleh *kernel Linux*. Hal ini menunjukkan indikasi bahwa Google mulai melirik untuk terjun dalam persaingan pasar telepon seluler. Perkembangan Android terbagi menjadi beberapa versi hingga saat ini. Beberapa versi Android tersebut antara lain:

- a. Android versi 1.1
- b. Android versi 1.5 (*Cupcake*)
- c. Android versi 1.6 (*Donut*)
- d. Android versi 2.0 / 2.1 (*Éclair*)
- e. Android versi 2.2 (*Froyo*)
- f. Android versi 2.3 (*Gingerbread*)
- g. Android versi 3.0 / 3.1 (*Honeycomb*)
- h. Android versi 4.0 (*Ice Cream Sandwich*)
- i. Android versi 5.0 (*Kitkat*)

Platform Android adalah *kernel Linux* versi 2.6 yang cukup *responsible* untuk berbagai *driver device*, sumber daya akses, manajemen daya, dan tugas lainnya. *Driver* perangkat yang disediakan meliputi *display*, *camera*, *keypad*, *WiFi*, kartu memori, *audio*, dan *IPC* (*interprocess communication*). Android merupakan generasi baru *platform mobile*, *platform* yang memberikan pengembang untuk melakukan pengembangan sesuai yang diinginkannya[5].

Perangkat berbasis android hanya mempunyai satu layar *foreground*. Normalnya saat menghidupkan android, yang pertama terlihat adalah *home*. Kemudian bila menjalankan sebuah aplikasi, misalnya catur, *User Interface* nya (UI) akan menumpuk di atas layar sebelumnya (*home*). Kemudian bila melihat *help*-nya catur, maka UI *help* akan menimpa UI sebelumnya (catur), begitu seterusnya. Semua proses diatas direkam di *application stack* oleh sistem *Activity*

manager. Menekan tombol *back* hanya kembali ke halaman sebelumnya. Analoginya mirip dengan *browser* dimana ketika meng-klik tombol *back*, *browser* akan kembali menampilkan halaman sebelumnya. Struktur dalam sistem android ada beberapa bagian, yaitu sebagai berikut[2] :

a. Activity

Normalnya setiap *activity* menampilkan satu buah *user interface* kepada pengguna. Misalnya sebuah *activity* menampilkan daftar menu minuman, kemudian pengguna dapat memilih satu jenis minuman. Contoh lainnya pada aplikasi sms, dimana satu *activity* digunakan untuk menulis pesan, *activity* berikutnya untuk menampilkan nomor kontak tujuan, atau *activity* lainnya digunakan untuk menampilkan pesan-pesan lama. Meskipun *activity-activity* diatas terdapat dalam satu aplikasi sms, namun masing-masing *activity* berdiri sendiri. Untuk pindah dari satu *activity* ke *activity* lainnya dapat melakukan suatu event misalnya tombol diklik atau melalui *trigger* tertentu.

b. Service

Service tidak memiliki *user interface*, namun berjalan di belakang layar. Misalnya *music player*, sebuah *activity* digunakan untuk memilih lagu kemudian di-play. Agar *music player* bisa berjalan di belakang aplikasi lain maka harus menggunakan *service*.

c. Intent

Intent adalah mekanisme untuk menggambarkan sebuah *action* secara detail seperti bagaimana cara mengambil sebuah foto.

d. Content Providers

Menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*, misalnya kita menggunakan aplikasi berbasis peta (MAP). *Activity* membutuhkan cara untuk mengakses data kontak untuk prosedur navigasi. Di sinilah peran *content providers*.

e. Resource

Resource digunakan untuk menyimpan file-file *non-coding* yang diperlukan pada sebuah aplikasi misalnya file icon, file gambar, file audio, file video atau yang lain. Gambar berformat JPG atau PNG sebuah aplikasi biasanya disimpan dalam

folder *res/drawable*, icon aplikasi disimpan dalam *res/drawable-ldpi* dan file audio disimpan dalam folder *res/raw*. File XML untuk membentuk sebuah *user interface* disimpan dalam folder *res/layout*.

Selain struktur di dalam android, adapula *layer - layer* yang tersusun dalam arsitektur android. *Layer-layer* tersebut adalah sebagai berikut :

a. Application Layer

Penggabungan beberapa aplikasi ini termasuk klien email, program SMS, kalender, peta, browser, kontak, dan lain-lain.

b. Application Framework

Dengan penyediaan platform yang *opensource*, pengembangan android menawarkan kemampuan untuk membangun aplikasi yang bervariasi dengan jUMLah banyak.

c. Libraries Layer

Pengembang android memasukkan C/C++ sebagai komponen sistem android.

d. Android Runtime Layer

Sebagian besar fungsi dari sistemnya menggunakan bahasa pemrograman *Java*.

e. Linux Kernel Layer

Keamanan sistem, manajemen memori, manajemen proses, *network stack*, dan model driver bergantung pada linux versi 2.6. Sehingga *kernel* mempunyai peran lain yaitu sebagai lapisan abstraksi antara *hardware* dan seluruh *software stack*.

1. Eclipse

Eclipse adalah sebuah *IDE (Integrated Development Environment)* untuk mengembangkan perangkat lunak dan dapat dijalankan di semua *platform (platform-independent)*. Berikut ini adalah sifat dari *Eclipse*[2] :

a. **Multi-platform**: Target sistem operasi *Eclipse* adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.

b. **Multilanguage**: *Eclipse* dikembangkan dengan bahasa pemrograman *Java*, akan tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti *C/C++*, *Cobol*, *Python*, *Perl*, *PHP*, dan lain sebagainya.

c. **Multi-role**: Selain sebagai IDE untuk pengembangan aplikasi, *Eclipse* pun

bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

2. Android Development Tools

Android Development Tools (ADT) adalah *plugin* untuk *Eclipse* yang didesain untuk pengembangan aplikasi Android. ADT memungkinkan *Eclipse* untuk digunakan dalam membuat aplikasi Android baru, membuat *User Interface*, menambahkan komponen berdasarkan *framework* API Android, *debug* aplikasi, dan pemaketan aplikasi Android.

3. Android Software Developments Kit

Android-SDK merupakan *tools* bagi para programmer yang ingin mengembangkan aplikasi berbasis google android. Android SDK mencakup seperangkat alat pengembangan yang komprehensif. Android SDK terdiri dari *debugger*, *libraries*, *handset emulator*, dokumentasi, contoh kode, dan *tutorial*. Saat ini Android sudah mendukung arsitektur *x86* pada *Linux* (distribusi *Linux* apapun untuk *desktop* modern), *Mac OS X 10.4.8* atau lebih, *Windows XP* atau *Vista*. Persyaratan mencakup *JDK*, *Apache Ant* dan *Python 2.2* atau yang lebih baru. IDE yang didukung secara resmi adalah *Eclipse 3.2* atau lebih dengan menggunakan *plugin Android Development Tools* (ADT), dengan ini pengembang dapat menggunakan teks editor untuk mengedit file *Java* dan *XML* serta menggunakan peralatan *command line* untuk menciptakan, membangun, melakukan *debug* aplikasi Android dan pengendalian perangkat Android (misalnya, *reboot*, menginstal paket perangkat lunak dengan jarak jauh).

4. Java

Java merupakan bahasa pemrograman berorientasi objek (OOP) yaitu cara ampuh

dalam pengorganisasian dan pengembangan perangkat lunak[5]. Pada OOP, program komputer sebagai kelompok objek yang saling berinteraksi. Deskripsi singkat OOP adalah mengorganisasikan program sebagai kumpulan komponen, disebut objek. Objek-objek ini ada secara independen, mempunyai aturan-aturan berkomunikasi dengan objek lain dan untuk memerintahkan objek lain guna meminta informasi tertentu atau meminta objek lain mengerjakan sesuatu.

5. UML (Unified Modelling Language)

UML adalah salah satu tool permodelan sistem berorientasi objek yang digunakan untuk memperlihatkan banyak pandangan terhadap suatu sistem yang akan dibangun[7]. *UML* terbagi menjadi beberapa model diagram, dimana setiap diagram menggambarkan masing-masing aktifitas dari *system* yang berjalan pada aplikasi. Adapun diagram yang akan dipergunakan oleh penulis yaitu terdiri dari *Use case Diagram*, *Sequence Diagram*, *Activity Diagram*, *Class Diagram* dan *Deployment Diagram*.

6. Pengujian Blackbox Testing

Black Box testing adalah tipe-tipe testing yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya[8]. Sehingga para tester memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi cukup dikenal proses testing dibagian luar. Jenis testing ini hanya memandang perangkat lunak dari sisi spesifikasi dan kebutuhan yang telah didefinisikan pada saat awal perancangan. Sebagai contoh, jika terdapat sebuah perangkat lunak yang merupakan sebuah system informasi inventory disebuah perusahaan. Maka pada jenis *White Box* testing perangkat lunak tersebut akan berusaha dibongkar listing programnya untuk kemudian di test menggunakan teknik-teknik yang telah dijelaskan sebelumnya. sedangkan pada jenis *black box testing*, perangkat lunak tersebut akan dieksekusi kemudian berusaha dites apakah telah memenuhi kebutuhan pengguna yang didefinisikan

pada saat awal tanpa harus membongkar listing programnya.

Pengujian *black box* berusaha menemukan kesalahan dalam kategori sebagai berikut[8]:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses database *eksternal*.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

Keunggulan dari pengujian *black box* adalah bisa memilih *subset* tes secara efektif dan efisien, dapat menemukan cacat, memaksimalkan *testing investmen*. Sedangkan kelemahan dari pengujian *black box* adalah *tester* tidak pernah yakin apakah perangkat lunak benar-benar lolos uji.

7. Pengujian Whitebox Testing

White box testing secara umum merupakan jenis *testing* yang lebih berkonsentrasi terhadap isi dari perangkat lunak itu sendiri"[8]. Dengan menggunakan metode *white box testing*, para *software engineer* dapat memperoleh uji kasusnya seperti berikut :

1. Menjamin bahwa semua alur program yang independen dalam sebuah modul telah dilaksanakan minimal satu kali.
2. Melaksanakan semua keputusan logis pada sisi yang benar dan salah.
3. Mengeksekusi semua putaran pada batasannya dan dalam batasan operasionalnya.
4. Menjalankan stuktur data *internal* untuk memastikan validitasnya.

Keunggulan pengujian *white box* adalah mampu mendeteksi kesalahan logika, ketidaksesuaian asumsi dan *case sensitive*. Sedangkan kelemahan dari pengujian *white box* adalah melibatkan sumber daya yang besar sehingga boros.

METODE PENELITIAN

Dalam memudahkan pembuatan dan pengumpulan data-data yang diperlukan dalam penelitian ini, maka peneliti menggunakan metode penelitian sebagai berikut :

1. Teknik Pengumpulan Data
Teknik yang dilakukan untuk pengumpulan data adalah sebagai berikut:

a. Observasi

Observasi dilakukan dengan mengakses website resmi KPK sub bagian Anti Corruption Clearing House (<http://acch.kpk.go.id>) dan dengan mengunjungi Perpustakaan KPK dimana pada web ACCH dan Perpustakaan KPK kita bisa mendapatkan informasi mengenai kasus yang sedang berjalan, kasus yang sebelumnya di tangani dan yang akan di tangani.

b. Studi Pustaka

Dalam penulisan skripsi ini, penulis mengambil studi pustaka dari buku dan Jurnal Ilmiah yang relevan dengan penelitian yang penulis bahas.

2. Analisa Penelitian

Analisa penelitian yang dilakukan terdiri dari :

a. Analisa Kebutuhan

Tahap analisa kebutuhan *software* merupakan salah satu usaha mengidentifikasi kebutuhan dan spesifikasi sistem yang akan diciptakan atau dibuat. Di dalamnya akan dijabarkan apa saja entitas yang dilibatkan, proses yang dijalankan, serta output yang akan dihasilkan. Berdasarkan hasil analisis masalah yang telah dijabarkan sebelumnya, solusi yang ditawarkan adalah dengan perancangan sistem aplikasi "**Pembelajaran Anti Korupsi Bagi Anak Remaja Berbasis Android**" dengan spesifikasi sistem sebagai berikut :

1. Sistem terdiri dari beberapa menu *activity* dimana setiap *activity* memiliki hubungan satu sama lainnya.
2. Sistem aplikasi berbasis *stand alone* dapat di akses kapan saja dan dimana saja jika sudah menginstallnya.
3. Sistem aplikasi akan menyediakan update jika terdapat perubahan dalam peraturan perundang-undangan yang dijadikan pendukung di dalam aplikasi pembelajaran tersebut.
4. Sistem aplikasi memiliki beberapa contoh studi kasus yang akan di perbaharui sesuai dengan situasi dan kondisi permasalahan korupsi.

PEMBAHASAN

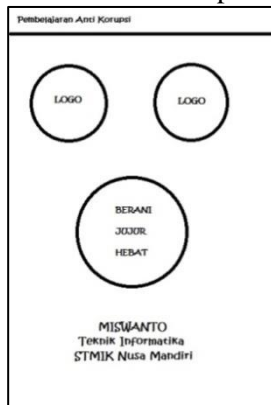
Langkah pertama yang dilakukan oleh penulis adalah menentukan aktor atau pengguna sistem merupakan tahap pertama yang harus dilakukan dalam melakukan analisis berorientasi objek menggunakan UML (*Unified Model Language*). Kata aktor dalam konteks UML, menampilkan peran (*roles*) yang pengguna (atau sesuatu di luar sistem yang dikembangkan dapat berupa perangkat keras, end *User*, sistem yang lain, dan sebagainya). Adapun yang akan digunakan dalam perancangan aplikasi pembelajaran Anti Korupsi bagi Anak Remaja berbasis android ini adalah metode *UML Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram dan Deployment Diagram*.

a. Desain

Perancangan desain antarmuka pada pembuatan aplikasi Pembelajaran Anti Korupsi berbasis Android yaitu terdiri dari beberapa bagian sebagai berikut :

1. Desain *SplashScreen*

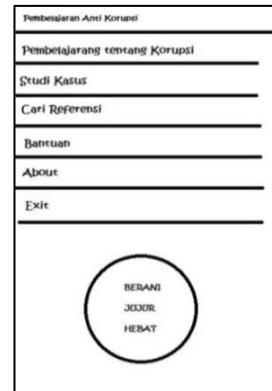
Menampilkan menu pada saat aplikasi pertama di jalankan, yaitu tampilan startup program. Berikut ini tampilannya :



Gambar 1
Desain splash screen

2. Desain Menu Utama

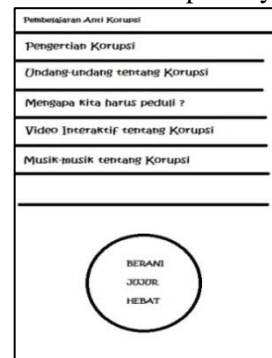
Menampilkan menu setelah *splashscreen*, pada menu utama terdapat 5 menu *activity*. Berikut ini tampilannya :



Gambar 2
Desain Menu utama

3. Desain Menu Pembelajaran Korupsi

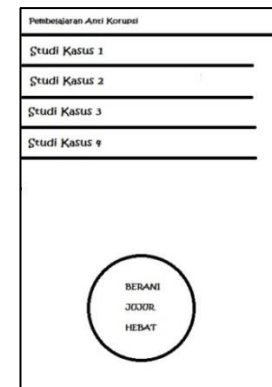
Menampilkan menu pilihan dari menu utama, berisi pengertian korupsi, undang-undang pendukung tentang korupsi, dan lain-lain. Berikut ini tampilannya :



Gambar 3
Desain Menu Pembelajaran Korupsi

4. Desain Menu Contoh Studi Kasus

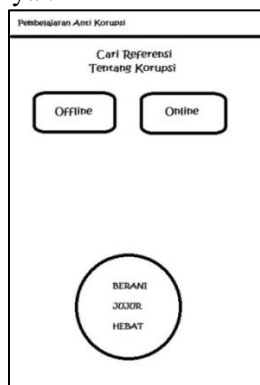
Menampilkan menu pilihan dari menu utama, berisi contoh kasus nyata dalam dunia remaja. Berikut ini tampilannya :



Gambar 4
Desain study kasus

5. Desain Menu Cari Referensi

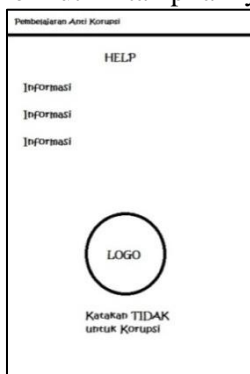
Menampilkan menu untuk mencari referensi dari lokal aplikasi dan atau artikel yang berhubungan dengan korupsi di internet. Berikut ini tampilannya :



Gambar 5
Desian cari referensi

6. Desain Menu Bantuan

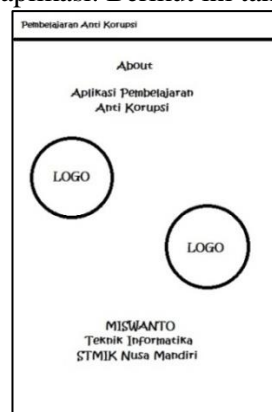
Menampilkan panduan pengoperasian program Pembelajaran Anti Korupsi (PAK). Berikut ini tampilannya :



Gambar 6
Desian menu bantuan

7. Desain Menu About

Menampilkan menu tentang perancang sistem aplikasi. Berikut ini tampilannya :



Gambar 7
Menu About

A. Testing

Melakukan testing aplikasi dan pengujian aplikasi, pada tahap testing akan dilakukan proses berikut ini :

1. Blackbox Testing

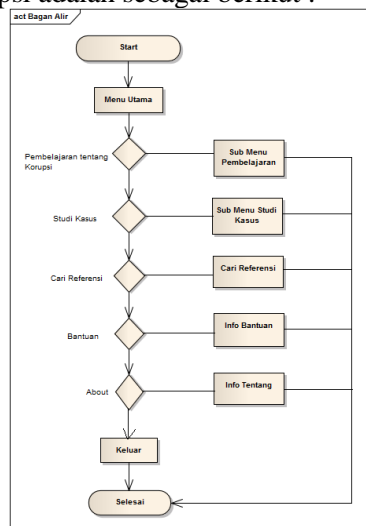
Black box testing adalah tahap pengujian yang berfokus pada fungsi perangkat lunak dan cara beroperasinya, apakah *interface* sudah bisa berjalan dengan baik atau belum. Pengujian *black box* berusaha menemukan kesalahan dalam hal fungsi-fungsi yang tidak benar atau hilang, kesalahan *interface*, kesalahan dalam akses *database*, kesalahan kinerja sistem. Hasil dari *Black Box Testing* pada penerapan aplikasi Pembelajaran Anti Korupsi adalah sebagai berikut :

Tabel 1
Hasil test menggunakan *Black Box Testing*

No	Skenario Uji	Test Case	Hasil yang diharapkan	Keterangan
1	Tampil Menu Utama	Menampilkan menu utama aplikasi	Tampil halaman utama	Valid
2	Memilih Menu Pembelajaran	Menampilkan menu pembelajaran tentang korupsi	Tampil halaman pembelajaran	Valid
3	Memilih Pengertian Korupsi	Menampilkan pengertian korupsi	Tampil halaman pengertian korupsi	Valid
4	Memilih UU Pendukung	Menampilkan UU pendukung korupsi	Tampil halaman UU pendukung	Valid
5	Memilih Mengapa Peduli Korupsi	Menampilkan alasan kita harus peduli korupsi	Tampil halaman mengapa harus peduli tentang korupsi	Valid
6	Memilih Video tentang Korupsi	Menampilkan video tentang korupsi	Tampil halaman video tentang korupsi	Valid
7	Memilih Musik tentang Korupsi	Menampilkan musik tentang korupsi	Tampil halaman musik tentang korupsi	Valid
8	Memilih Menu Studi Kasus	Menampilkan studi kasus tentang korupsi	Tampil halaman studi kasus	Valid
9	Memilih Studi Kasus1	Menampilkan studi kasus 1	Tampil halaman studi kasus 1	Valid
10	Memilih Studi Kasus2	Menampilkan studi kasus 2	Tampil halaman studi kasus 2	Valid
11	Memilih Studi Kasus3	Menampilkan studi kasus 3	Tampil halaman studi kasus 3	Valid
12	Memilih Menu Cari Referensi	Menampilkan menu cari referensi tentang korupsi	Tampil halaman cari referensi	Valid
13	Memilih Bantuan	Menampilkan info bantuan aplikasi	Tampil halaman bantuan	Valid
14	Memilih About	Menampilkan info tentang pembuat	Tampil halaman about	Valid
15	Memilih Keluar	Keluar dari Menu Utama	Tampil halaman keluar	Valid

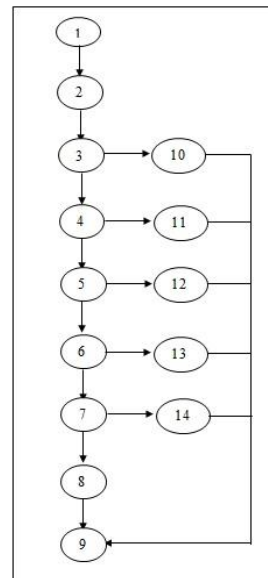
2. Whitebox Testing

White box testing adalah cara pengujian untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan output yang tidak sesuai dengan proses yang dilakukan, maka baris-baris program, variabel dan parameter pada unit tersebut akan dicek satu persatu dan diperbaiki kemudian di *compile* ulang. Hasil dari *White Box Testing* pada penerapan aplikasi Pembelajaran Anti Korupsi adalah sebagai berikut :



Gambar 8
Bagan Alir

Sedangkan untuk grafik alir adalah sebagai berikut ini :



Gambar 9
Grafik Alir

Kompleksitas siklomatis dari grafik alir *white box* dapat di peroleh dengan perhitungan:

$$V(G) = E - N + 2$$

Dimana:

E = Jumlah *Edge* yang ditentukan gambar panah

N = Jumlah simpul grafik alir ditentukan dengan gambar lingkaran

$$V(G) = 18 - 14 + 2 = 6$$

$V(G) < 10$ berarti memenuhi syarat kekompleksitasian siklomatisnya. Baris set yang dihasilkan dari jalur *independent* adalah sebagai berikut:

- a. 1-2-3-10-9
- b. 1-2-3-4-11-9
- c. 1-2-3-4-5-12-9
- d. 1-2-3-4-5-6-13-9
- e. 1-2-3-4-5-6-7-14-9
- f. 1-2-3-4-5-6-7-8-9

Ketika aplikasi dijalankan, maka terlihat bahwa satu set baris yang dihasilkan adalah 1-2-3-10-9-1-2-3-4-11-9-1-2-3-4-5-12-9-1-2-3-4-5-6-13-9-1-2-3-4-5-6-7-14-9-1-2-3-4-5-6-7-8-9 dan terlihat bahwa simpul telah dieksekusi satu kali.

3. Implementasi

Untuk proses implementasi ada beberapa hal yang harus diperhatikan, adapun hal tersebut adalah sebagai berikut :

1. Pastikan *source* dari Aplikasi Pembelajaran Anti Korupsi bagi anak remaja sudah tersedia atau sudah di download.
2. Pastikan *smartphone* android memiliki versi di atas 2.3 (*Gingerbread*) agar memudahkan proses instalasi.

Untuk proses implementasi aplikasi Pembelajaran Anti Korupsi (PAK) ini penulis juga memberikan hasil implementasi dari pembuatan aplikasi ini. Adapun hasil dari implementasi adalah sebagai berikut :

A. Menu Splash Screen

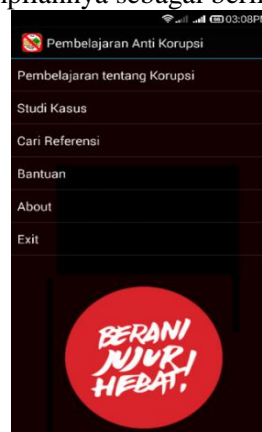
Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu *splashscreen* untuk tampilannya sebagai berikut :



Gambar 10
Menu *Splash Screen*

B. Menu Utama Aplikasi PAK

Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu utama untuk tampilannya sebagai berikut :



Gambar 11
Menu Utama Aplikasi

C. Menu Pembelajaran tentang Korupsi

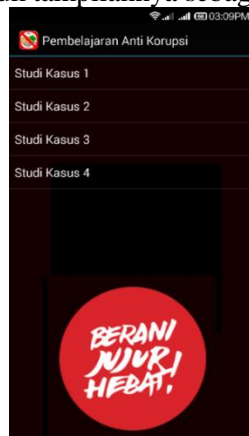
Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu pembelajaran tentang Korupsi untuk tampilan sebagai berikut :



Gambar 12
Menu Pembelajaran tentang Korupsi

D. Menu Studi Kasus

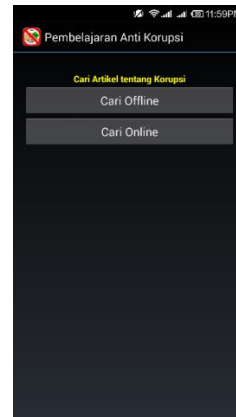
Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu studi kasus untuk tampilannya sebagai berikut :



Gambar 13
Menu Studi Kasus tentang Korupsi

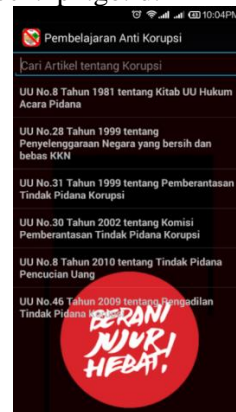
E. Menu Cari Referensi

Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu cari referensi untuk tampilannya sebagai berikut :

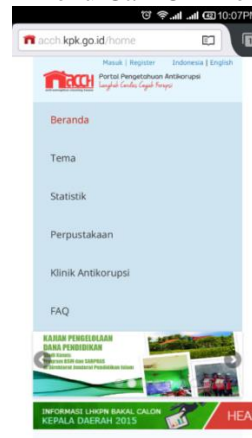


Gambar 14
Menu Cari Referensi tentang Korupsi

Untuk menu cari offline maka akan menampilkan *database local* berisi file PDF yang tentang referensi undang-undang yang berhubungan dengan Tindak Pidana Korupsi. Sedangkan untuk menu cari online, maka pengguna akan langsung di arahkan ke *web browser* untuk membuka internet dengan index halaman ke <http://acch.kpk.go.id>.



Gambar 15
Menu Cari Offline



Gambar 16
Menu Cari Online

F. Menu Bantuan

Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu bantuan untuk tampilannya sebagai berikut :



Gambar 17
Menu Bantuan

G. Menu About

Berikut ini adalah hasil implementasi dari aplikasi “Perancangan Sistem Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja Berbasis Android”, menu about untuk tampilannya sebagai berikut :



Gambar 18
Menu About

4. Support

Adapun perangkat yang penulis gunakan untuk pemasangan sistem yang telah berhasil di ujikan, dan di implementasikan oleh penulis ke beberapa perangkat *smartphone android*, adalah sebagai berikut :

a. Hardware

1. Komputer (*Personal Computer*) atau Laptop (*Notebook*)

Spesifikasi :

- Processor minimum Intel Core 2 Duo 1,8 GHz*
- Processor minimum AMD E2-1800 APU 1,7 GHz*
- RAM 2 GB*
- VGA Card Memory 512 MB*
- LAN Card 100 Mbps*
- Harddisk 250 GB*

2. *Smartphone Android*

Spesifikasi :

- Processor Dual Core 1,6 GHz*
- RAM 512 MB*
- Wifi / bisa akses internet*
- Minimum versi Android 2.3 (Gingerbread)*

b. Software

Spesifikasi :

- Eclipse 3.6.2 (Helios) atau lebih tinggi*
- Eclipse JDT Plugin versi 6 atau lebih tinggi*
- Java Development Kit (JDK) versi 6 atau lebih tinggi*
- Android Development Tools Pugin versi 22 atau lebih tinggi*

PENUTUP

Kesimpulan yang dapat penulis ambil dari Perancangan Aplikasi Pembelajaran Anti Korupsi Berbasis Android ini adalah sebagai berikut :

- Aplikasi “Pembelajaran Anti Korupsi” ini memberikan kemudahan bagi masyarakat, khususnya remaja untuk mengakses berbagai informasi yang berkaitan dengan pengertian Korupsi dan pentingnya sikap Anti Korupsi dengan menggunakan perangkat *Mobile* sehingga jauh lebih fleksibel dan efektif bagi remaja, untuk mengetahui seluk beluk tentang korupsi yang terjadi di Indonesia.
- Dengan adanya Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja berbasis Android ini, *user* dapat mengetahui hal-hal yang berkaitan dengan tindak pidana korupsi seperti undang-undang tentang Korupsi dan

undang-undang yang berhubungan dengan korupsi.

3. Aplikasi Pembelajaran Anti Korupsi bagi Anak Remaja berbasis android ini berhasil dibangun dengan menggunakan metode algoritma *Brute Force* dan dapat di install minimal versi android adalah 2.3 (*Gingerbread*).

REFERENSI

- Artikel Internet dari Komisi Pemberantasan Korupsi. 2015. Rekapitulasi Penindakan Pidana Korupsi. Diambil dari : <http://acch.kpk.go.id/statistik-rekapitulasi-penindakan>. 19 Juli 2015
- Irawan. Membuat Aplikasi Android untuk Orang Awam. Palembang : MAXIKOM, 2012.
- Kurniawan, Teguh. 2009. Peranan Akuntabilitas Publik dan Partisipasi Masyarakat dalam Pemberantasan Korupsi di Pemerintahan. ISSN : 0854 - 3844. Jakarta : Bisnis & Birokrasi, Jurnal Ilmu Administrasi dan Organisasi, Vol. 16, No. 2 Mei Agustus 2009 : 116-121.
- Mulyawan, Sang dan Arief Budiman. Pembelajaran tentang Tindak Pidana Korupsi berbasis Mobile. ISSN : 2089 - 9815. Yogyakarta : Seminar Nasional Teknologi Informasi dan Komunikasi, 9 Maret 2013.
- Safaat, Nazruddi. Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung : INFORMATIKA, 2014.
- Sarno, Riyanto. Semantic Search Pencarian Berdasarkan Konten. Yogyakarta : Andi, 2012.
- Sholiq. Permodelan Sisten Informasi Berorientasi Obyek dengan UML. Yogyakarta : Graha Ilmu, 2006.
- Soetam, Rizky. 2011. Konsep Dasar Rekayasa Perangkat Lunak, Jakarta: PT.Prestasi Pustaka Raya

SMPN 6 Jakarta Lulus Tahun 2004, SMKN 24 Jakarta Lulus Tahun 2007, Kemudian Penulis Melanjutkan Program Diploma 3 di AMIK BSI Jakarta Jurusan Teknik Komputer Lulus Tahun 2011 dan melanjutkan Program Strata Satu di Kampus STMIK Nusa Mandiri dengan Jurusan Teknik Informatika. Saat ini Penulis bekerja di PT. Infomedia Solusi Humanika untuk Posisi Network Engineer dari tahun 2012 sampai dengan sekarang.

Mohammad Badrul, M.Kom

Penulis adalah Dosen Tetap di STMIK Nusa Mandiri Jakarta. Penulis Kelahiran di Bangkalan 01 Januari 1984. Penulis menyelesaikan Program Studi Strata 1 (S1) di Kampus STMIK Nusa Mandiri Prodi Sistem Informasi dengan gelar S.Kom pada tahun 2009 dan menyelesaikan program Strata 2 (S2) di Kampus yang sama dengan Prodi ilmu Komputer dengan gelar M.Kom pada tahun 2012. Selain mengajar, Penulis juga aktif dalam membimbing mahasiswa yang sedang melakukan penelitian khususnya di tingkat Strata 1 dan penulis juga terlibat dalam tim konsorsium di Jurusan Teknik Informatika STMIK Nusa Mandiri untuk penyusunan bahan ajar. Saat ini penulis memiliki Jabatan Fungsional Asisten ahli di kampus STMIK Nusa Mandiri Jakarta. Penulis tertarik dalam bidang kelimuan Data mining, Jaringan komputer, Operating sistem khususnya open source, Database, Software engineering dan Research Metode.

BIODATA PENULIS

Miswanto

Penulis kelahiran Jakarta, 17 Maret 1988, Riwayat Pendidikan penulis dari SDN Klender 14 Pagi Lulus Tahun 2001,