

PERANCANGAN APLIKASI EDUKASI PENGENALAN BAHASA INGGRIS BERBASIS ANDROID PADA TK AN – NASIR TANGERANG

¹⁾Wahyudin, ²⁾Abdul Romansyah, ³⁾Maya Selvia Lauryn

Program Studi Sistem Komputer Universitas Serang Raya

¹⁾wahyudin@gmail.com, ³⁾

Abstrak - Perkembangan teknologi saat ini semakin cepat dan ilmu pengetahuan yang semakin maju di butuhkan penguasaan bahasa inggris. Bahasa inggris akan lebih mudah di ajarkan pada anak usia dini. Untuk daya tangkap pembelajaran yang baik pada saat ini lebih di tekankan pada pendidikan visual. Konten edukasi ternyata paling di sukai oleh pengguna android. Media pembelajaran selalu mengalami perkembangan seiring perkembangan teknologi. Untuk daya tangkap pembelajaran yang baik pada saat ini atau sebelumnya lebih di tekankan kepada pendidikan visual yang terlihat menarik dengan tambahan gambar dan suara. Aplikasi edukasi berbasis android hanya bisa digunakan oleh handphone berbasis android saja. Dan untuk mempelajari aplikasi (m-learning) edukasi pengenalan bahasa inggris berbasis android lebih mudah dan praktis. Diharapkan aplikasi edukasi pengenalan bahasa inggris bisa membantu dalam memahami dan mempelajari bahasa Inggris.

Kata Kunci : Bahasa Inggris, Aplikasi edukasi, *android*, *m-learning*

I. PENDAHULUAN

Perkembangan teknologi saat ini semakin cepat dan ilmu pengetahuan yang semakin maju di butuhkan penguasaan bahasa inggris. Bahasa inggris akan lebih mudah di ajarkan pada anak usia dini. Untuk daya tangkap pembelajaran yang baik pada saat ini lebih di tekankan pada pendidikan *visual*. Konten edukasi ternyata paling di sukai oleh pengguna *android*. Salah satunya adalah aplikasi yang berbasis pendidikan atau edukasi seperti *m-learning*.

Untuk daya tangkap pembelajaran yang baik pada saat ini atau sebelumnya lebih di tekankan kepada pendidikan *visual* yang terlihat menarik dengan tambahan gambar dan suara.

Menurut Sobri dalam Marsa, Sardiarinto (2012:156) “bahasa inggris akan lebih mudah di ajarkan pada anak usia dini”. Seperti kata pepatah, belajar di waktu kecil bagai mengukir di atas batu, belajar sesudah dewasa serupa mengukir di atas air. Begitu juga ketika belajar bahasa asing. Memang akan lebih efektif jika memulai belajar bahasa asing dari kecil karena di usia itulah anak-anak bisa dengan mudah meniru pelafalan yang benar.

Menurut Pacidda dalam Marsa, Sardiarinto (2010:156) “kebanyakan orang belajar bahasa inggris karena ingin berkomunikasi secara lebih baik dalam bahasa inggris”. Jika ini adalah tujuan yang telah di terapkan, penting untuk mempelajari secara seimbang ke empat *skill* utama. *Listening, Speaking, Reading* dan *Writing* adalah *skill-skill* utama (makro) yang di perlukan untuk berkomunikasi dalam bahasa apapun.

Menurut Louiciano dalam Marsa, Sardiarinto (2011:156) “konten edukasi ternyata paling di sukai oleh pengguna *android*”. Hal tersebut ditampilkan oleh *opera software* melalui laporan *state of the mobile web* pada september 2011. Melalui data anonim, *opera* memperoleh gambaran konten apa saja yang di akses melalui *opera mini* yang ada di *smartphone* berbasis *android, windows, iOS, feature phone* dan *RIM*.

Dan juga perkembangan aplikasi-aplikasi berbasis *android* semakin melesat jauh di bandingkan aplikasi-aplikasi berbasis lainnya. Di antaranya aplikasi yang

menyangkut tentang edukasi ataupun pendidikan, dimana belajar menjadi mudah dan *efisien* dengan menggunakan perangkat *mobile* yaitu *m-learning*. Apalagi aplikasi edukasi untuk pembelajaran anak usia dini dengan menggunakan perangkat *android* akan lebih mudah dan cepat mencerna setiap apa yang akan di pelajari terutama aplikasi edukasi tentang bahasa inggris.

Beberapa materi dalam *m-learning* ini adalah mengajarkan tentang pengenalan bahasa inggris untuk murid-murid TK AN-NASIR Tangerang. Materi yang diajarkan dalam aplikasi *m-learning* ini adalah pengenalan huruf *alphabet*, pengenalan hewan, pengenalan anatomi, pengenalan nama buah, pengenalan angka, pengenalan warna, pengenalan sayuran, dan pengenalan nama-nama benda dalam bahasa inggris dan penyebutannya dalam bahasa indonesia dan inggris. Serta terdapat *quiz* yang diambil dari materi-materi pengenalan yang diambil dari pembelajaran yang telah diterapkan pada aplikasi ini seperti menebak gambar dan menulis. Dan aplikasi ini dibuat untuk meningkatkan daya ingat anak agar dapat mengingat materi yang sudah di pelajari di aplikasi *m-learning* ini.

II. METODE PENELITIAN

Penulis melakukan beberapa metode dalam proses pengumpulan data antara lain :

- Observasi dilakukan dengan cara mengumpulkan data dengan melakukan penelitian dan peninjauan langsung terhadap tempat sebenarnya yaitu pada TK AN-NASIR TANGERANG.
- Wawancara dilakukan untuk mengumpulkan data dengan cara tanya jawab secara langsung pada staff pengajar atau guru, murid-murid dan orang tua di TK (taman kanak-kanak) AN-NASIR Tangerang.
- Studi pustaka dilakukan untuk mendapatkan teori untuk menyelesaikan permasalahan dengan mengumpulkan teori-teori lain yang mendukung dan membaca sumber seperti buku, jurnal, *ebook*, maupun karangan yang berkaitan dengan judul penelitian.

II. LANDASAN TEORI

2.1. Tinjauan Jurnal

Menurut Sari, Sasongko (2013:23) “media pembelajaran merupakan wahana dan penyampaian informasi atau pesan pembelajaran pada siswa”. Dengan adanya media pada proses belajar mengajar, diharapkan dapat membantu guru dalam meningkatkan prestasi belajar pada siswa. Media pembelajaran selalu mengalami perkembangan seiring perkembangan teknologi. Sebagai salah satu contohnya penggunaan media pembelajaran interaktif untuk proses pembelajaran.

Menurut Rahadi (2014:661) menyimpulkan bahwa : “*interface* pengguna *Android* didasarkan pada manipulasi langsung menggunakan masukan sentuh yang serupa dengan tindakan di dunia nyata, seperti menggesek, mengetuk, mencubit, dan membalikkan cubitan untuk memanipulasi obyek di layar”.

Masukkan penggunaan direspon dengan cepat dan juga tersedia antarmuka sentuh layaknya permukaan air, seringkali menggunakan kemampuan getaran perangkat untuk memberikan umpan balik haptik kepada pengguna. Perangkat keras internal seperti *Akselerometer*, *giroskop*, dan *sensor prokmitas* digunakan oleh beberapa aplikasi untuk merespon tindakan pengguna, misalnya untuk menyesuaikan posisi layar dari *potrait* ke *lanskap*, tergantung pada bagaimana perangkat diposisikan, atau memungkinkan pengguna untuk mengarahkan kendaraan saat bermain balapan dengan memutar perangkat sebagai simulasi kendali setir.

2.2. Konsep Dasar Program

Menurut Kurniawan, (2011:65) menjelaskan bahwa : “Program adalah kumpulan intruksi yang disusun dengan urutan nalar yang tepat untuk menyelesaikan suatu persoalan”. Dalam pembuatan program, pemrograman mempunyai cara pandang terhadap eksekusi terhadap program yang disebut sebagai paradigma.

Pembuatan program tentunya tidak lepas dari tahapan-tahapan yang harus dikerjakan secara terstruktur, untuk membantu pemrogram dalam menyelesaikan programnya dengan baik. Untuk lebih jelasnya tahapan-tahapan perancangan program secara umum adalah sebagai berikut :

1. Definisi Masalah.
2. Analisis Kebutuhan.
3. Definisi Algoritma.
4. Pengkodean.
5. *Testing* Program.
6. Dokumentasi Program.
7. Pemeliharaan

2.2.1. *Android*

1. Mengenal *Android*

Menurut Murya (2014:3) “*Android* merupakan sebuah sistem operasi yang berbasis *Linux* untuk perangkat portabel seperti *smartphone* dan komputer tablet”. *Android* menyediakan *platform* terbuka (*open source*) bagi programmer untuk mengembangkan aplikasi sendiri pada berbagai perangkat dengan sistem *Android*.

Awalnya, *google* sebagai perusahaan raksasa membeli *Android Inc* beserta teknologinya. *Google* melanjutkan untuk mengembangkan *Android* untuk bersaing pada pasar *smartphone*. Dalam usaha pengembangan *Android*, dibentuklah *open handset alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan

telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcom*, *T-Mobile*, dan *Nvidia*.

Saat itu *Android* diresmikan sebagai produk pertamanya, sebuah *platform* perangkat seluler yang menggunakan *karnel Linux Versi 2.6*, dan untuk setiap *release*-nya diberi kode nama berdasarkan nama hidangan makanan penutup. Telepon seluler komersial pertama yang menggunakan sistem operasi *Android* adalah *HTC Dream*, yang di luncurkan pada 22 Oktober 2008.

2. *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.

3. *Resource*

Resource digunakan untuk menyimpan *file-file non-coding* yang diperlukan pada sebuah aplikasi misalnya *file icon*, *file picture*, *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*.

4. *XML*

XML (eXtensible Markup Language) dikembangkan mulai tahun 1996 dan mendapatkan pengakuan dari *W3C* pada bulan Februari 1998. Teknologi yang digunakan pada *XML* sebenarnya bukan teknologi baru, tapi merupakan turunan dari *SGML* yang telah dikembangkan pada awal 80-an dan telah banyak digunakan pada dokumentasi teknis sebagai proyek berskala besar. Ketika *HTML* dikembangkan pada tahun 1990, para pengggagas *XML* mengadopsi bagian paling penting pada *SGML* dan dengan berpedoman pada pengembangan *HTML* menghasilkan markup language yang tidak kalah hebatnya dengan *SGML*.

Seperti halnya *HTML*, *XML* juga menggunakan elemen yang ditandai dengan tag pembuka (diawali dengan „<” dan diakhiri dengan „>”), tag penutup (diawali dengan „</” dan diakhiri „>”) dan *atribut elemen* (*parameter* yang dinyatakan dalam tag pembuka misal <form name=“isidata”>). Hanya bedanya, *HTML* mendefinisikan dari awal tag dan atribut yang dipakai didalamnya, sedangkan pada *XML* kita bisa menggunakan tag dan atribut sesuai kehendak kita.

5. *Android Manifest*

File AndroidManifest.xml diperlukan oleh setiap aplikasi *Android*. Diletakkan di *folder root aplikasi*, dan mendeskripsikan nilai *global* paket aplikasi kita. Termasuk komponen yang ada pada aplikasi (*Activity*, *service*) yang memang diekspos untuk sistem diluar aplikasi yang membutuhkan. Termasuk juga data pada tiap *Activity* kita, bagaimana menanganinya dan menluncurkan (*launched*).

Hal terpenting yang harus diperhatikan dari *File AndroidManifest.xml* merupakan *intentFilters*. *intentFilter* menjelaskan dimana dan kapan *Activity* dapat dijalankan. Ketika *Activity* (atau sistem operasi) akan melakukan sesuatu, misalnya membuka halaman *web* atau membuka layar kontak, itu akan membuat *objek Intent*. *Objek Intent* ini akan menangani beberapa informasi yang mendeskripsikan apa yang akan dilakukan oleh aplikasi dan data apa yang dibutuhkan untuk itu, selain itu juga beberapa informasi lain.

2.2.2. Java

Java Deployment Kit (JDK) berisi sekumpulan kaskas baris perintah (*command-line tool*) untuk menciptakan program *java*. Rilis terakhir *JDK* dapat di *download* dari alamat [URL](http://java.sun.com/products/JDK/index.html) berikut : <http://java.sun.com/products/JDK/index.html>. *JDK* berisi sekumpulan kaskas, utilitas, dan dokumentasi serta kode *applet* contoh untuk pengembangan program *java*.

2.2.3. Software Development Kit (SDK)

Menurut Irawan (2012:24) "*Android SDK* digunakan untuk mengembangkan aplikasi *Android*". *Android SDK* berisi *API*, *sample*, dan dokumentasi dari *Android* yang akan dikembangkan. *Android SDK* mencakup perangkat *tools* pengembangan yang *komprehensif*. *Android SDK* terdiri dari *debugger*, *libraries*, *handset emulator*, *dokumentasi*, contoh kode program 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*.

IDE yang didukung secara resmi adalah *Eclipse 3.2* atau lebih dengan menggunakan *plugin Android Development Tools (ADT)*, dengan ini pengembang dapat menggunakan *IDE* untuk mengedit dokumen *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).

2.2.4. Android Development Tools (ADT)

Menurut Arifianto (2011:36) "*Android Development Tools (ADT)* adalah aplikasi yang digunakan untuk mengembangkan aplikasi *android* pada *eclipse*". Untuk membuat aplikasi pada *Android* dengan menggunakan *Eclipse*, *Android* menawarkan custom *plugin* untuk *Eclipse IDE* yang disebut *Android Development Tools (ADT)*. *ADT* dirancang untuk memberikan lingkungan yang terintegrasi untuk membangun *aplikasi Android*. *ADT* meningkatkan kemampuan *Eclipse* untuk membuat pengguna lebih cepat membuat *project Android*, membuat *aplikasi UI*, menambahkan komponen berdasarkan *Android Framework API* dan men-*debug aplikasi* Anda menggunakan *SDK Android tools*.

2.2.5. Delvik Virtual Machine (DVM)

Menurut Hermawan (2011:5) "*Delvik Virtual Machine (DVM)* adalah untuk mengoptimalkan pada perangkat *mobile*". *Android* berjalan di dalam *Dalvik Virtual Machine (DVM)* bukan di *Java Virtual Machine (JVM)*, sebenarnya bnyak persamaannya dengan *Java Virtual Machine (JVM)* seperti *Java ME (Java Mobile Edition)*, tetapi *Android* menggunakan *Virtual Machine* dikustomisasi dan dirancang untuk memastikan bahwa beberapa *feature-feature* berjalan lebih *efisien* pada perangkat *mobile*.

Semua *hardware* yang berbasis *Android* dijalankan dengan menggunakan *Virtual Machine* untuk *eksekusi aplikasi*, pengembang tidak perlu khawatir tentang *implementasi* perangkat keras tertentu. *Dalvik Virtual Machine* mengeksekusi *executable file*, sebuah *format* yang dioptimalkan untuk memastikan memori yang digunakan sangat kecil. *The executbale file* diciptakan dengan mengubah kelas bahasa *java* dan kompilasi menggunakan *tools* yang disediakan dalam *SDK Android*.

2.2.6. Eclipse Integrated Development Enviroment (IDE)

Menurut Maulana (2014:60) "*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* :

- Multi-platform: Target sistem operasi Eclipse* adalah *Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X*.
- Multilanguage* : *Eclipse* dikembangkan dengan bahasa pemrograman *Java*, akan tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lain seperti *C/C++, Cobol, Python, Perl, PHP*, dan lain sebagainya.
- Multi-role* : Selain sebagai *IDE* untuk pengembangan *aplikasi*. *Eclipse* pun bisa digunakan untuk *aktivitas* dalam siklus pengembangan perangkat lunak seperti *dokumentasi, pengujian perangkat lunak, pengembangan web*, dan lain sebagainya.

Pada saat ini, *Eclipse* merupakan salah satu *IDE* favorit karena gratis dan *open source*. *Open source* 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 membuat komponen yang disebut *plug-in*.

2.3. Metode Algoritma

Menurut ngoen (2011:205) "*Algoritma mergesort* adalah mengurutkan data dengan pendekatan *divide-and-conguar*, Yaitu membagi masalah semula menjadi sejumlah submasalah sejenis yang masing-masing ukurannya lebih kecil".

Algoritma-algoritma pengurutan sebelumnya manampung hasil pengurutan pada larik yang sama tanpa bantuan larik lain. Pengurutan seperti itu disebut *in-place-sorting*. Berbeda dengan *algoritma-algoritma* tersebut, *mergesort* menampung hasil pengurutan sementara di larik lain sehingga memerlukan memori yang lebih besar.

Mergesort melakukan proses pengurutan dengan cara menggabung data. Penggabungan data dilakukan mulai dari kelompok data yang paling kecil, yaitu satu data digabung satu data. Hasil gabungan yang berupa dua data selanjutnya digabung dengan dua data hasil gabung kelompok lain.

2.4. Pengujian Sistem

Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah dimana dapat menempatkan desain kasus uji yang spesifik dan metode pengujian.

Sering perangkat lunak mengandung kesalahan (*error*) pada proses-proses tertentu pada saat perangkat lunak sudah berada ditangan user. Kesalahan-kesalahan (*error*) pada perangkat lunak ini sering disebut dengan "*bug*". Untuk menghindari banyaknya *bug* maka diperlukan

adanya pengujian perangkat lunak sebelum perangkat lunak diberikan ke pelanggan atau selama perangkat lunak masih terus dikembangkan.

Setelah pengujian sistem selesai dilakukan maka dapat dilakukan pengujian penerimaan perangkat lunak oleh pelanggan (*customer*) atau pemakai perangkat lunak (*user*). Pengujian penerimaan digunakan untuk mengetahui kepuasan pelanggan atau user terhadap perangkat lunak yang telah dibuat. Jika pelanggan sudah puas dengan perangkat lunak, maka perangkat lunak dapat diserahkan kepada pelanggan (*customer*).

Pengujian untuk validasi memiliki beberapa pendekatan sebagai berikut :

1. Pengujian *Black Box*
2. Pengujian *White Box*

2.5. Peralatan Pendukung

Membuat suatu aplikasi *Android* diperlukan berbagai macam pendukung yang dapat membantu terbentuknya aplikasi mulai dari logika hingga konsep pemrogramannya.

1. *UML(Unified Modeling Language)*

Menurut Rosa, Shalaluddin (2013:137) "*UML* merupakan bahasa *visual* untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung". *UML* hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan *UML* tidak terbatas pada metodologi tertentu, meskipun pada kenyataan *UML* paling banyak digunakan pada metodologi berorientasi objek.

Pada perkembangan teknologi perangkat lunak, diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang di berbagai negara dapat mengerti pemodelan perangkat lunak. Banyak orang yang telah membuat bahasa pemodelan pembangunan perangkat lunak sesuai dengan teknologi pemrograman yang berkembang pada saat itu, misalnya yang sempat berkembang dan digunakan oleh banyak pihak adalah data *flow diagram (DFD)* untuk memodelkan perangkat lunak yang menggunakan pemrograman prosedural atau struktural, kemudian juga ada *State Transition Diagram (STD)* yang digunakan untuk memodelkan sistem realtime (waktu nyata).

Seperti yang kita ketahui bahwa banyak hal di dunia sistem informasi yang tidak dapat dibakukan, semua tergantung kebutuhan, lingkungan dan konteksnya. Begitu juga dengan perkembangan penggunaan *UML* bergantung pada level abstraksi penggunaannya.

- a. *Activity Diagram*
- b. *Component diagram*
- c. *Deployment diagram*

III. ANALISIS DAN RANCANGAN SOFTWARE

3.1. Analisa Kebutuhan Software

Untuk membuat aplikasi ini maka penulis ingin mempermudah *user* dalam mempelajari bahasa inggris, seiring dengan perkembangan *gadget* saat ini mempelajari bahasa inggris bisa dengan praktis yaitu dengan menggunakan *handphone* yang berbasis *android*.

3.1.1 Identifikasi Masalah

Pengenalan bahasa inggris pada zaman sekarang ini sangatlah kurang dan banyak yang tidak mengerti dalam memahami bahasa inggris. Untuk mempelajari bahasa inggris pada zaman sekarang ini sangat minim

dikarenakan banyak dari mereka yang tidak mengerti dan sulit untuk mengingat kata-kata dalam bahasa inggris yang mereka pelajari. Dan juga kemauan untuk belajar bahasa inggris pada zaman sekarang ini sedikit, banyak dari mereka yang tidak bisa berbahasa inggris. Padahal bahasa inggris di zaman sekarang ini sangatlah penting dalam dunia karir mereka dalam bekerja. Karena perusahaan-perusahaan sekarang ini jika ingin bekerja di perusahaan pasti akan di tes bahasa inggris baik dasar maupun yang sulit.

Padahal jika kita mau belajar bahasa inggris itu pasti mudah dan menyenangkan karena belajar bahasa inggris di usia dini itu mudah untuk mengingat, karena otak di usia dini ini gampang untuk mengingat. Karena permasalahan tersebut maka penulis merancang aplikasi pengenalan bahasa inggris untuk anak usia dini dengan pengenalan bahasa inggris yang mendasar atau mudah yang berbasis *mobile android*. Namun dalam membuat sebuah aplikasi *mobile* yang sederhana ini tidaklah mudah, karena itu penulis melakukan riset pada sekolah TK An-Nasir untuk memberikan materi dan menguji anak-anak yang belajar di sekolah tersebut dengan memberikan sebuah materi yang ada di aplikasi *mobile* tersebut. Dan mengetahui bagaimana cara belajar anak-anak usia dini dalam belajar bahasa inggris.

3.1.2. Analisa kebutuhan

1. Tahap Analisa Kebutuhan

Tahap analisa kebutuhan mencakup hardware, software, aplikasi, dan *output* yang digunakan adalah sebagai berikut :

a. Komponen Hardware

Dalam pembuatan aplikasi *android* minimal harus menggunakan *processor Dual Core* karena akan berpengaruh terhadap pembuatan *virtual android* yang akan digunakan.

Sedangkan perangkat *Gadget* yang digunakan oleh penulis mempunyai klasifikasi sebagai berikut :

- 1) *Vendor : Handphone Smartfren Andromax C*
- 2) *Tipe : AD686G*
- 3) *Procesor : Adreno 203 Dual Core 1 GHz*
- 4) *Os Android : 4.0 Ice Cream Sandwich (ICS)*
- 5) *RAM : 512Mb*
- 6) *Sd Internal : 4Gb Rom*
- 7) *Resolution : 4" TFT LCD with WVGA 480x800*

b. Spesifikasi Perangkat Lunak

Adapun perangkat lunak (*software*) yang penulis gunakan dalam merancang aplikasi *English For Children* yaitu :

- 1.) *Windows 7 Profesional 64 bit.*
- 2.) *Eclipse juno*
- 3.) *Paint*
- 4.) *Audacity 1.3 beta*
- 5.) *Youwave Android 4.1.2*
- 6.) *Adobe Photoshop CS6*

2. kebutuhan Informasi

Dalam merancang aplikasi *English For Children*, harus berpedoman pada karakteristik dan unsur yang terdapat pada aplikasi tersebut yaitu :

- a. *Format.* Program telah di *format .apk* sehingga pengguna dapat menginstal aplikasi *English For Children* pada perangkat *handphone* yang berbasis *Android*.

- b. *Rules*. Pada aplikasi *English For Children* ini terdapat beberapa menu pilihan untuk mengenal bahasa inggris dasar yaitu *alphabet*, hewan, anatomi, buah, sayuran, warna, angka, benda, dan *quiz*. Sehingga pengguna dapat memilih mana yang akan dipelajari dan bisa memahami bahasa inggris dengan mudah.
- c. *Policy*. Untuk menjalankan aplikasi ini pengguna harus menginstal aplikasi ber ekstensi *.apk dengan nama file *English For Children.apk* ini pada *gadget* atau *tablet* berbasis *android*. Jika ingin menjalankan aplikasi ini pada *laptop* atau *PC* dengan sistem operasi *Windows* maka pengguna sebelumnya harus menginstal *emulatorenya* yaitu *YouWave* secara *offline* pada *Laptop* atau *PC* pengguna. Setelah *YouWave* terinstall, cara menjalankan aplikasi ini adalah menaruh atau *copy file* aplikasi di *direktori C* dengan nama *folder YouWave*, kemudian jalankan emulator *YouWave* dan Klik Aplikasi *English For Children*.
- d. *Score Model*. *Score model* digunakan dalam menu *Quiz* yaitu soal pilihan ganda dimana penilaian terdiri dari 10 soal masing-masing soal mempunyai *score* 10, jika menjawab salah maka *score* 0 dan jika menjawab benar maka *score* 10.
- e. *Scenario*. *Scenario* dalam aplikasi *English For Children* ini yaitu :
 - 1.) Menu utama. Pada menu ini akan menampilkan beberapa pilihan untuk memulai berjalannya aplikasi yang akan di jalankan dan akan menuju menu pilihan.
 - 2.) *Alphabet*. Pada menu ini akan menampilkan tentang pengenalan huruf alfabet A-Z dan juga cara pengucapannya dalam bahasa inggris dan indonesia
 - 3.) *Animal*. Pada menu ini akan menampilkan tentang pengenalan hewan dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 4.) *Anatomi*. Pada menu ini akan menampilkan tentang bagian dari tubuh manusia dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 5.) *Number*. Pada menu ini akan menampilkan tentang Angka dari 0-10 dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 6.) *Colour*. Pada menu ini akan menampilkan tentang warna-warna dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 7.) *Fruit*. Pada menu ini akan menampilkan tentang buah-buahan dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 8.) *Object*. Pada menu ini akan menampilkan tentang benda-benda yang ada di kelas dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 9.) *Vegetable*. Pada menu ini akan menampilkan tentang sayur-sayuran dan juga cara pengucapannya dalam bahasa inggris dan indonesia.
 - 10.) *Quiz*. Pada menu ini terdapat beberapa soal yang akan di tampilkan yaitu soal pilihan ganda, menulis dan menebak gambar, soal ini di ambil dari menu-menu yang ada di *English For Children*.
 - 11.) *About*. Pada menu ini akan menampilkan tentang aplikasi *English For Children*.
 - 12.) Kritik dan Saran. Pada menu ini akan menampilkan tentang saran dan kritik yang bisa di kirimkan

melalui email yang tertulis di bagian kritik dan saran.

- 13.) *Help*. Pada menu ini akan menampilkan bantuan untuk bisa menjalankan aplikasi *English For Children*.

- 14.) *Exit*. Pada menu ini akan menutup aplikasi yang sedang berjalan dan akan terdapat pesan tidak dan iya, jika iya maka aplikasi akan keluar tapi jika tidak aplikasi tetap berjalan.

3.3. Desain

sebelum aplikasi android ini di implementasikan dalam bentuk package .apk, maka perlu dirancang terlebih dahulu. Tahap perancangan bertujuan untuk memenuhi kebutuhan pengguna dan memberikan gambaran yang jelas mengenai aplikasi yang akan dibuat. Keseluruhan dari perancangan ini akan di implementasikan dalam gadget berbasis android dengan menggunakan perangkat lunak Eclipse.

3.3.1. Rancangan Algoritma

Algoritma yang digunakan adalah algoritma *MergeSort*, Algoritma *MergeSort* digunakan dalam menu "quiz pilihan ganda" yang memiliki variable benar untuk jawaban benar dan variable salah untuk jawaban salah setiap soal memiliki dua buah variable tetap misalkan untuk soal nomor 1 variabelnya adalah benar=10 dan salah=0, dalam aplikasi ini setiap soal memiliki dua pilihan jawaban dan jawaban yang bisa dipilih hanya satu. Jika user memilih jawaban yang benar maka akan mendapatkan variable benar dan jika memilih jawaban yang salah maka akan mendapatkan variable salah. penulis mengambil contoh quiz pilihan ganda dari soal nomor satu sampai soal nomor sepuluh, bentuknya adalah sebagai berikut:

```
public int nilai_01, nilai_02, nilai_03,
nilai_04, nilai_05, nilai_06,
nilai_07, nilai_08, nilai_09, nilai_10,
jumlah_nilai;
public ImageButton cek_nilai;
public static int nilai_total = 0;
public String jawaban, hasil_jawaban,
isi_jawaban;
public static String data_jawaban = "";
int[] kunci = { 1, 2, 1, 1, 2, 1, 1, 2,
1, 2 };
int[] soal = { 1, 2, 3, 4, 5, 6, 7, 8, 9,
10 };
protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_soal_1);
menu_jawab_01 = (Spinner)
findViewById(R.id.jwb_01);
menu_jawab_02 = (Spinner)
findViewById(R.id.jwb_02);
public void btn_check_Clicked(View v) {
int pilihan_jawaban[] = { nilai_01, nilai_02,
nilai_03, nilai_04,
nilai_05, nilai_06, nilai_07, nilai_08,
nilai_09, nilai_10 };
int nomor_pilihan;
for (int i = 0; i < kunci.length; i++) {
if (pilihan_jawaban[i] == 1) {
jawaban = " A ";
} else if (pilihan_jawaban[i] == 2) {
jawaban = " B ";
} else {
jawaban = " - ";
}
if (kunci[i] == 1) {
isi_jawaban = "A";
} else if (kunci[i] == 2) {
```

```

isi_jawaban = "B";
} else {
isi_jawaban = "Tidak dijawab";
}
if (pilihan_jawaban[i] == kunci[i]) {
nilai_total = nilai_total + 10;
hasil_jawaban = jawaban + " (Benar) = " +
isi_jawaban
+ ", (Nilai = 10)\n";
} else {
hasil_jawaban = jawaban + " (Salah) = " +
isi_jawaban
+ ", (Nilai = 0)\n";
}
nomor_pilihan = i + 1;
data_jawaban = data_jawaban +
Integer.toString(nomor_pilihan)
+ ". " + hasil_jawaban;
}
Intent panggil_class = new Intent(this,
Hasil_Soal_1.class);
panggil_class.putExtra("nilai_hasil",
nilai_total);
panggil_class.putExtra("rinci_jawaban",
data_jawaban);
startActivity(panggil_class);
finish();
System.exit(0);
}

```

3.3.2. Software Architecture

Berikut ini adalah *source code* yang menjadi kontruksi dasar dari aplikasi mobile *English For Children*, yaitu :

1. Kontruksi tampilan Splash Screen *English For Children*

```

package com.example.englishforchildren;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.MotionEvent;
import android.content.pm.ActivityInfo;
public class English extends Activity {
protected boolean _aktif = true;
protected int splash_ = 5000;
@Override
protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_english);
;
setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
Thread splashTread = new Thread()
{
@Override
public void run() {
try {
int delay_ = 0;
while
(_aktif && (delay_ < splash_)) {
sleep(100);
if (_aktif)
{
delay_ += 100;
}
}
catch
(InterruptedException e) {
} finally {
finish();
Intent
panggil_class = new Intent(English.this,
Menu_Utama.class);
startActivityForResult(panggil_class, 0);
}
};
splashTread.start();
}
}

```

```

}
@Override
public boolean onTouchEvent(MotionEvent
event) {
if (event.getAction() ==
MotionEvent.ACTION_DOWN) {
_aktif = false;
}
return true;
}
@Override
public void onBackPressed() {
finish();
System.exit(0);
}
}

```

2. Kontruksi Tampilan Menu Utama *English For Children*

```

package com.example.englishforchildren;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.view.View;
public class Menu_Utama extends Activity {
private MediaPlayer MPlayer;
@Override
protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_menu_ut
ama);

setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
buka_file_suara();
}
@Override
public void onResume() {
super.onResume();
stop_intro();
play_intro();
}
private void play_intro() {
if (MPlayer != null &&
!MPlayer.isPlaying()) {
MPlayer.setLooping(true);
MPlayer.start();
} else {
MPlayer.setLooping(true);
}
}
private void stop() {
MPlayer.stop();
try {
MPlayer.prepare();
MPlayer.seekTo(0);
} catch (Throwable t) {
error_(t);
}
}
private void stop_intro() {
if (MPlayer.isPlaying()) {
stop();
}
}
private void buka_file_suara() {
try {
MediaPlayer MPlayer =
MediaPlayer.create(this, R.raw.marsha);
} catch (Throwable t) {
error_(t);
}
}
private void error_(Throwable t) {
}
}

```

```

        AlertDialog.Builder pesan_error =
new AlertDialog.Builder(this);

        pesan_error.setTitle("Failed!").setMessage(
t.toString())
        .setPositiveButton("OK", null).show();
    }
    public void btn_about_Clicked(View v) {
        Intent panggilan_class = new
Intent(this, About.class);
        startActivity(panggil_class);
    }
    public void btn_menu_Clicked(View v) {
        stop_intro();
        Intent panggilan_class = new
Intent(this, Menu_Pilihan.class);
        startActivity(panggil_class);
    }
    public void btn_kritik_Clicked(View V) {
        AlertDialog.Builder builder = new
AlertDialog.Builder(this);
        builder.setMessage("Kritik dan saran kamu
dapat di kirim ke email ini ya...
(romansyah.abdul@gmail.com)")
        .setNegativeButton("oke",
new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface
dialog, int id) {
                dialog.cancel();
            }
        }).show();
    }
    public void btn_help_Clicked(View V) {
        Intent panggilan_class = new Intent(this,
Help.class);
        startActivity(panggil_class);
    }
    public void btn_exit_Clicked(View v) {
        AlertDialog.Builder builder = new
AlertDialog.Builder(this);
        builder.setMessage("Thank You !!!
Apakah kamu Ingin Tutup Aplikasi English For
Children ?")
        .setCancelable(false)
        .setPositiveButton("Iya",
new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog,
int id) {
                Intent keluar = new
Intent(Intent.ACTION_MAIN);
                keluar.addCategory(Intent.CATEGORY_HOME);
                keluar.setFlags(Intent.FLAG_ACTIVITY_NEW_
TASK);
                stop_intro();
                finish();
                startActivity(keluar);
            }
        })
        .setNegativeButton("Tidak",
new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface
dialog, int id) {
                dialog.cancel();
            }
        }).show();
    }
    @Override
    public void onBackPressed() {
        AlertDialog.Builder builder = new
AlertDialog.Builder(this);
        builder.setMessage("Thank You !!!
Apakah Ingin Tutup Aplikasi English For
Children ?")
        .setCancelable(false)

```

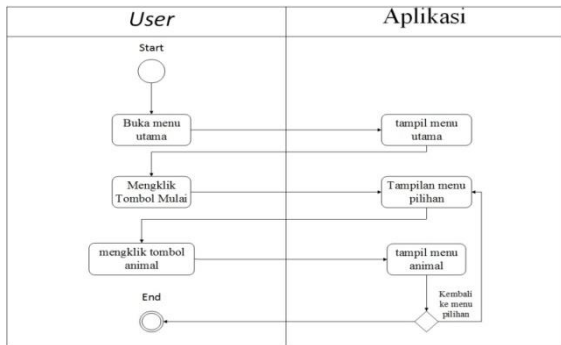
```

        .setPositiveButton("Iya",
new
DialogInterface.OnClickListener() {
            @Override
            public void
onClick(DialogInterface dialog, int id) {
                Intent
keluar = new Intent(Intent.ACTION_MAIN);
                keluar.addCategory(Intent.CATEGORY_HOME);
                keluar.setFlags(Intent.FLAG_ACTIVITY_NEW_
TASK);
                stop_intro();
                finish();
                startActivity(keluar);
            }
        })
        .setNegativeButton("Tidak",
new
DialogInterface.OnClickListener() {
            @Override
            public void
onClick(DialogInterface
dialog, int id) {
                dialog.cancel();
            }
        }).show();
    }
}

3. Konstruksi Tampilan Menu Pilihan English For
Children
package com.example.englishforchildren;
import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.view.View;
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.media.MediaPlayer;
public class Menu_Pilihan extends Activity {
    private MediaPlayer MPlayer;
    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_menu_pi
lihan);
        setRequestedOrientation(ActivityInfo.SCRE
EN_ORIENTATION_PORTRAIT);
        buka_file_suara();
    }
    @Override
    public void onResume() {
        super.onResume();
        stop_intro();
        play_intro();
    }
    private void play_intro() {
        if (MPlayer != null &&
!MPlayer.isPlaying()) {
            MPlayer.setLooping(true);
            MPlayer.start();
        } else {
            MPlayer.setLooping(true);
        }
    }
    private void stop() {
        MPlayer.stop();
        try {
            MPlayer.prepare();
            MPlayer.seekTo(0);
        } catch (Throwable t) {
            error_(t);
        }
    }
    private void stop_intro() {
        if (MPlayer.isPlaying()) {
            stop();
        }
    }
    private void buka_file_suara() {
        try {

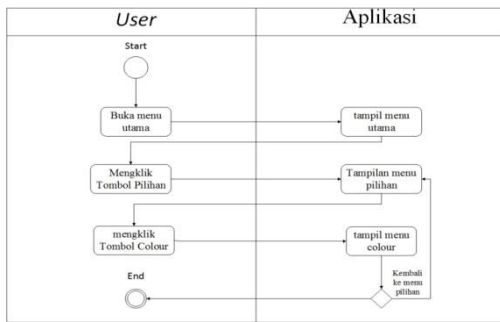
```


- d. *Activity Diagram* Menu *Animal*. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *animal* untuk ke menu *animal*.



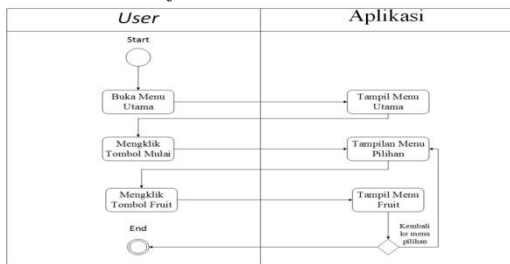
Gambar *Activity Diagram* Menu *Animal*

- e. *Activity Diagram* Menu *Colour*. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *colour* untuk ke menu *colour*.



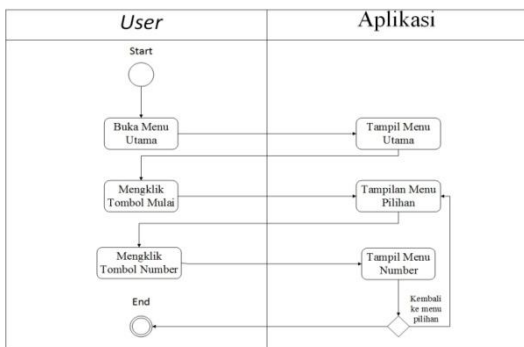
Gambar *Activity Diagram* Menu *Colour*

- f. *Activity Diagram* Menu *Fruit*. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *fruit* untuk ke menu *fruit*.



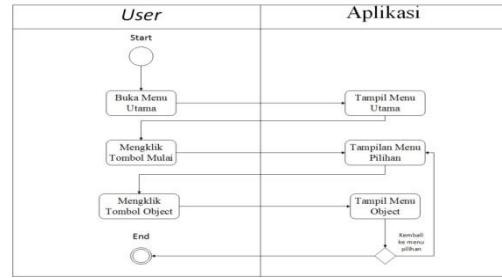
Gambar *Activity Diagram* Menu *Fruit*

- g. *Activity Diagram* Menu *Number*. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *number* untuk ke menu *number*.



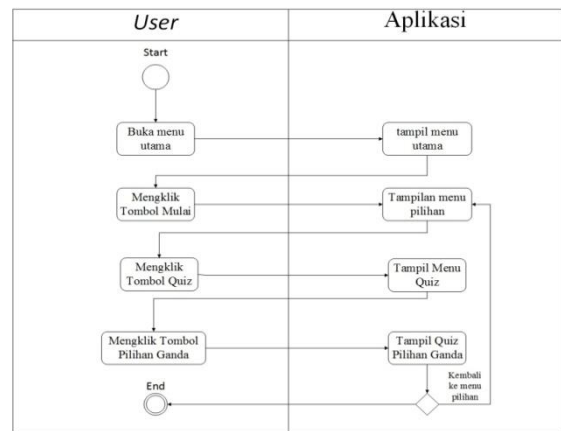
Gambar *Activity Diagram* Menu *Number*

- h. *Activity diagram* Menu *Object*. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *object* untuk ke menu *object*.



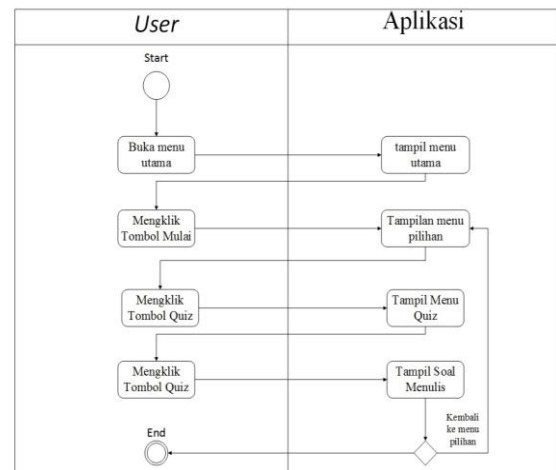
Gambar *Activity Diagram* Menu *Object*

- i. *Activity Diagram* Menu *Quiz* Pilihan Ganda. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *Quiz* untuk ke menu *Quiz* pilihan ganda.



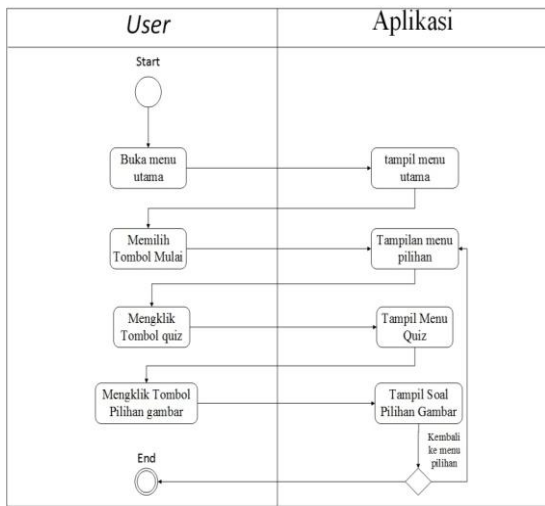
Gambar *Activity Diagram* Menu *Quiz* Pilihan Ganda

- j. *Activity Diagram* Menu *Quiz* Menulis. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *Quiz* untuk ke menu *Quiz* Menulis.



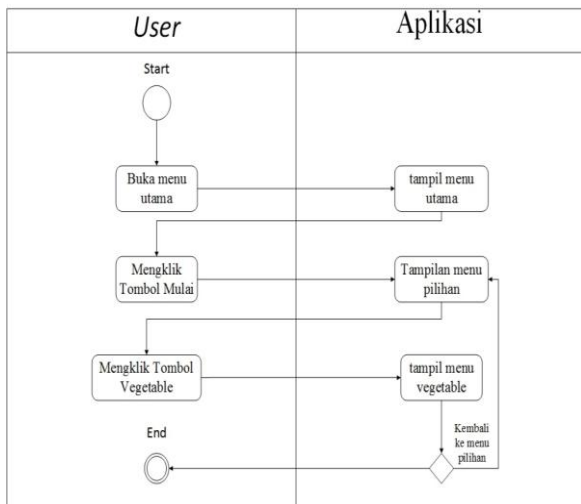
Gambar *Activity Diagram* Menu *Quiz* Menulis

- k. *Activity Diagram* Menu *Quiz* Pilihan Gambar. Diagram ini menggambarkan mengenai *activity* yang ada pada menu pilihan yang nantinya terdapat tombol *Quiz* untuk ke menu *Quiz* Pilihan Gambar.



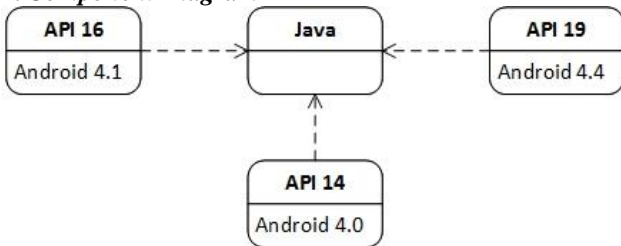
Gambar Activity Diagram Quiz Pilihan Gambar

1. Activity Diagram Menu Vegetable. Diagram ini menggambarkan mengenai activity yang ada pada menu pilihan yang nantinya terdapat tombol vegetable untuk ke menu vegetable.



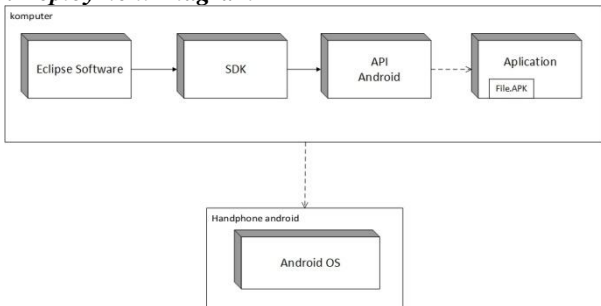
Gambar Activity Diagram Menu Vegetable

2. Component Diagram



Gambar Component Diagram

3. Deployment Diagram



Gambar Deployment Diagram

4. Perancangan Antarmuka

Sebelum aplikasi *English For Children* ini diimplementasikan dalam bentuk aplikasi handphone, makan perlu dirancang terlebih dahulu hasil pada tahap perancangan yang berkaitan erat dengan hasil tahap analisis karena pada tahap analisis telah ditemukan fungsi-fungsi yang diperlukan, perangkat keras dan perangkat lunak yang digunakan, serta karakteristik yang diharapkan. Keseluruhan dari perancangan ini akan diimplementasikan dalam aplikasi *English For Children*. Perancangan ini dibagi menjadi 2 bagian yaitu :

a. Perancangan Story Board

Story board merupakan konsep komunikasi dan ungkapan kreatif, teknik dan media untuk menyampaikan pesan dan gagasan secara visual, termasuk audio dengan mengolah elemen desain grafis berupa bentuk dan gambar, huruf dan warna, serta tata letaknya. Berikut adalah *story board* yang penulis gunakan pada perancangan aplikasi *English For Children*, yaitu :

1.) *Story Board Splash Screen*. Berikut ini adalah gambaran dari *story board splash screen* seperti yang dijelaskan oleh tabel berikut ini:

Tabel *Story Board Splash Screen*

Visual	Sketsa	Audio
Dalam theme dialog ini merupakan tampilan <i>Splash Screen</i> terdapat <i>image view</i> , <i>progress bar</i> dan tulisan <i>please wait</i> .		

2.) *Story Board Menu utama*. Berikut ini adalah gambaran dari *story board menu utama* seperti yang dijelaskan oleh tabel berikut ini :

Tabel *Story Board Menu Utama*

Visual	Sketsa	Audio
Dalam tampilan menu utama ini terdapat beberapa tombol pilihan yaitu : kritik dan saran, <i>about</i> , menuju menu berikutnya, <i>help</i> , dan <i>exit</i> , dan juga terdapat <i>background</i> .		Marsha. MP3

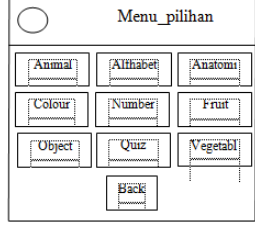
3.) *Story Board Kritik dan Saran*. Berikut ini adalah gambaran dari *story board kritik dan saran* seperti yang dijelaskan oleh tabel berikut ini :

Tabel *Story Board Menu Kritik dan Saran*

Visual	Sketsa	Audio
Dalam tombol kritik dan saran akan menampilkan <i>message box</i> yang berisikan tentang kritik dan saran yang bisa di kirimkan oleh <i>user</i> .		Marsha. MP3

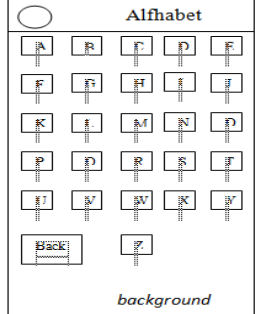
4.) *Story Board* Menu Pilihan. Berikut ini adalah gambaran dari *story board* menu pilihan seperti yang dijelaskan oleh tabel berikut ini:

Tabel *Story Board* Menu Pilihan

Visual	Sketsa	Audio
Dalam tampilan menu pilihan terdapat beberapa tombol pilihan yaitu : <i>animal, alphabet, anatomi, colour, number, fruit, object, quiz, vegetable, back</i> . Dan juga terdapat <i>background</i> .		Splash bg.MP3

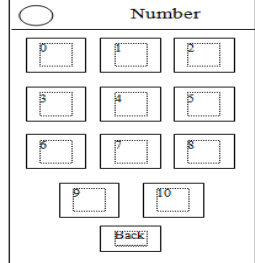
5.) *Story Board* Alhabet. Berikut ini adalah gambaran dari *story board* alhabet seperti yang dijelaskan oleh tabel berikut ini :

Tabel *Story Board* Alhabet

Visual	Sketsa	Audio
Dalam menu alhabet terdapat tombol pilihan huruf dari A-Z, dan juga terdapat <i>background</i> .		Intro. MP3

6.) *Story Board* Number. Berikut ini adalah gambaran dari *story board* number seperti yang dijelaskan oleh tabel berikut ini :

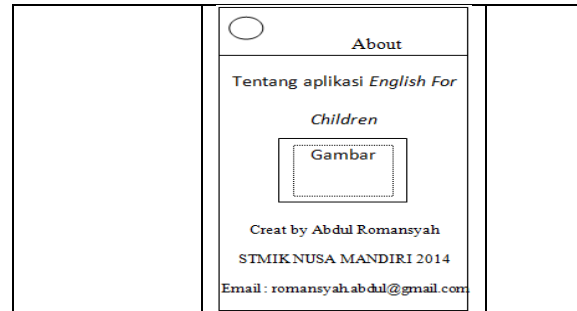
Tabel *Story Board* Number

Visual	Sketsa	Audio
Dalam menu <i>number</i> terdapat tombol angka dari 0-10 dan juga terdapat <i>background</i> .		Intro2. mp3

7.) *Story Board* About Berikut ini adalah gambaran dari *story board* about seperti yang dijelaskan oleh tabel berikut ini :

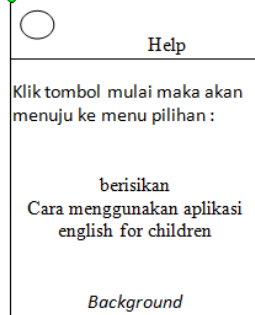
Tabel *Story Board* About

Visual	Sketsa	Audio
Dalam tampilan <i>about</i> menampilkan tentang aplikasi <i>english for childred</i> .		Marsha .Mp3



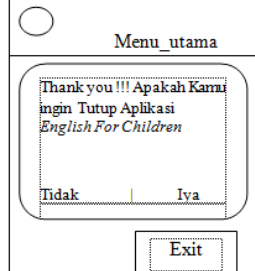
8.) *Story Board* Help. Berikut ini adalah gambaran dari *story board* help seperti yang dijelaskan oleh tabel berikut ini :

Tabel *story Board* Help

Visual	Sketsa	Audio
Dalam tombol <i>help</i> akan menampilkan tentang bagaimana menggunakan aplikasi <i>english for children</i> .		Marsha .MP3

9.) *Story Board* Exit. Berikut ini adalah gambaran dari *story board* exit seperti yang dijelaskan oleh tabel berikut ini :

Tabel *Story Board* Exit

Visual	Sketsa	Audio
Dalam tombol <i>exit</i> akan menampilkan <i>message box</i> untuk keluar dari aplikasi <i>english for children</i> , terdapat 2 pilihan yaitu ya dan tidak.		Marsha .MP3

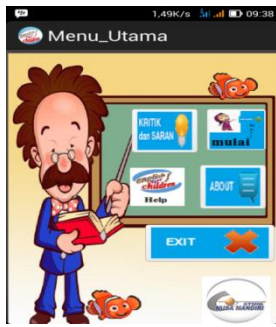
b. Tampilan Antarmuka

1.) Tampilan *Splash Screen*. *Splash screen* merupakan tampilan awal dari aplikasi *english for children* yang nantinya akan masuk ke menu utama.



Gambar Tampilan *Spalsh Screen*

2.) Tampilan Menu Utama. Menu utama menampilkan 5 pilihan menu yaitu, menu pilihan, *about*, kritik dan saran, *help*, dan *exit*.



Gambar Tampilan Menu Utama

- 3.) Tampilan Menu Pilihan. Menu pilihan menampilkan 10 pilihan menu yaitu, *alfhabet*, *animal*, *anatomi*, *colour*, *fruit*, *number*, *object*, *quiz*, *vegetable* dan *back*.



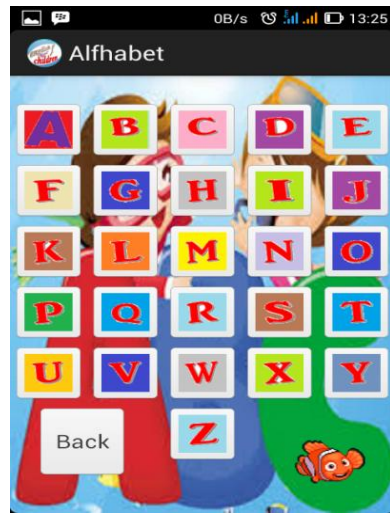
Gambar Tampilan Menu Pilihan

- 4.) Tampilan *About*. *About* menampilkan tentang aplikasi *English For Children*.



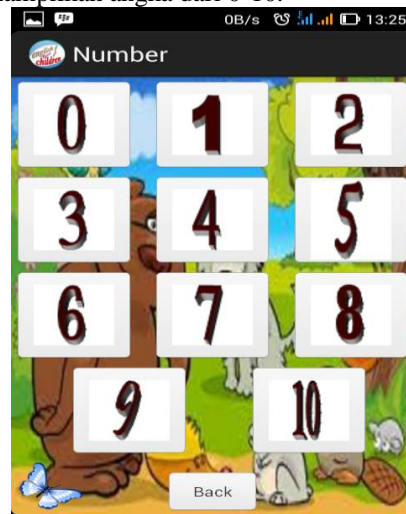
Gambar Tampilan *About*

- 5.) Tampilan *Alfhabet*. Pada menu alfhabet menampilkan huruf A-Z.



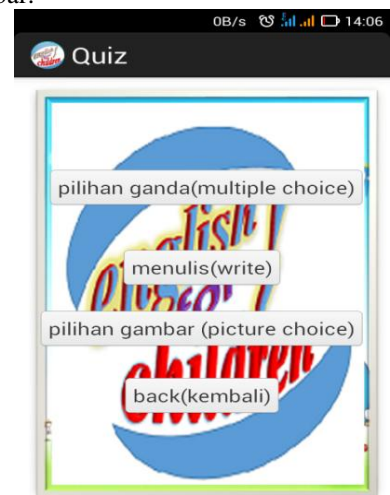
Gambar Tampilan *Alfhabet*

- 6.) Tampilan Menu *Number*. Pada menu *number* menampilkan angka dari 0-10.



Gambar Tampilan *Number*

- 7.) Tampilan Menu *Quiz*. Pada menu *Quiz* menampilkan 3 soal yaitu pilihan ganda, menulis dan pilihan gambar.



Gambar Tampilan *Quiz*

- 8.) Tampilan *Exit*. Tombol *exit* untuk keluar dari aplikasi *English For Children* dan terdapat 2 pilihan yaitu *tidak* dan *iya*, jika *iya* maka aplikasi akan tertutup dan jika *tidak* maka aplikasi tetap berjalan.



Gambar Tampilan Exit

- 9.) Tampilan Kritik dan saran
Tombol kritik dan saran untuk memberikan saran atau kritik dari aplikasi *English For Children* dengan membuka email dan mengirimkan email yang ada di aplikasi.



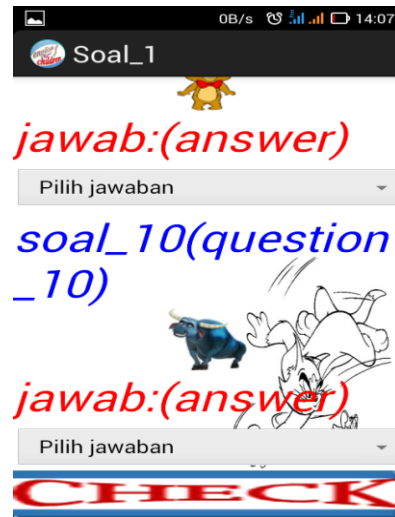
Gambar Tampilan Kritik dan Saran

- 10.) Tampilan Huruf A. Menampilkan tampilan huruf A dan terdapat 2 tombol yaitu Bahasa inggris dan bahasa indonesia dan terdapat penulisan huruf A dalam bahasa inggris.



Gambar Tampilan Huruf A

- 11.) Tampilan Soal Pilihan Ganda. Menampilkan tampilan soal pilihan ganda yaitu pilihan A dan B. Dan terdapat soal gambar untuk materi pada soal.



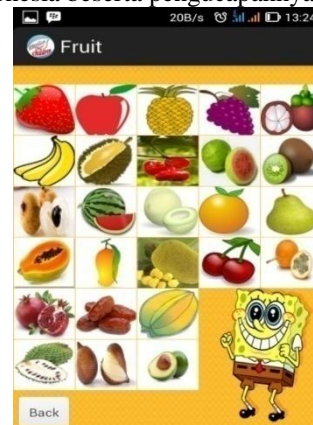
Gambar Tampilan Soal Pilihan Ganda

- 12.) Tampilan Hasil Soal Pilihan Ganda
Menampilkan tampilan hasil soal pilihan ganda dan terdapat nilai benar dan salah.



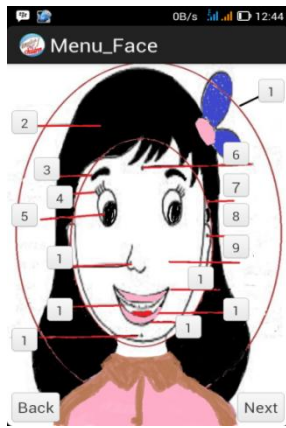
Gambar Tampilan hasil Soal Pilihan Ganda

- 13.) Tampilan Menu *Fruit*. Pada menu *fruit* menampilkan buah, jika dipilih salah satu buah maka akan menampilkan penulisan bahasa inggris dan indonesia beserta pengucapannya.



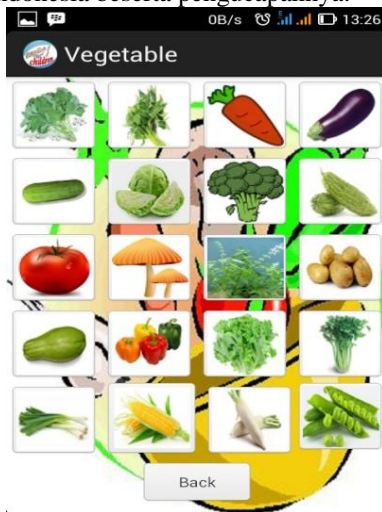
Gambar Tampilan Menu *Fruit*

- 14.) Tampilan Menu *Face*. Pada menu *face* menampilkan wajah manusia dan terdapat peletakan tombol sesuai dengan nama bagian dari wajah manusia, jika di klik salah satu tombol maka akan menampilkan penulisan dalam bahasa inggris dan indonesia beserta pengucapannya.



Gambar Tampilan Menu Face

15.) Tampilan Menu Vegetable Pada menu *vegetable* menampilkan sayuran, jika dipilih salah satu buah maka akan menampilkan penulisan bahasa inggris dan indonesia beserta pengucapannya.



Gambar Tampilan Menu vegetable

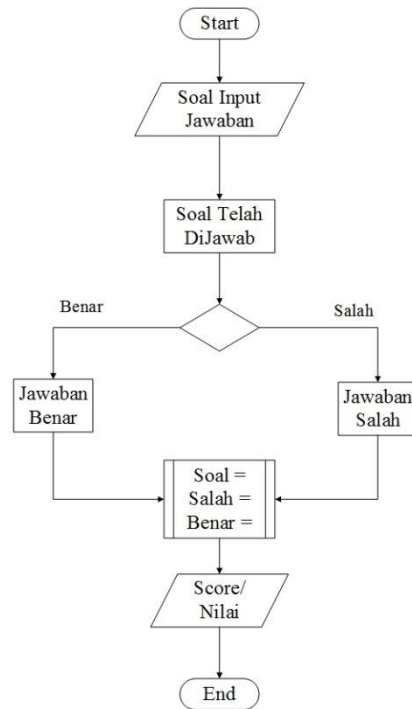
16.) Tampilan Menu Animal. Pada menu *animal* menampilkan hewan, jika dipilih salah satu buah maka akan menampilkan penulisan bahasa inggris dan indonesia beserta pengucapannya.



Gambar Tampilan Menu animal

3.4. Implementasi

1. *flowchart* Aplikasi *english for children* dari quiz pilihan ganda dengan menggunakan metode *marge sort* antara lain :



Gambar Flow Chart Algoritma quiz pilihan ganda

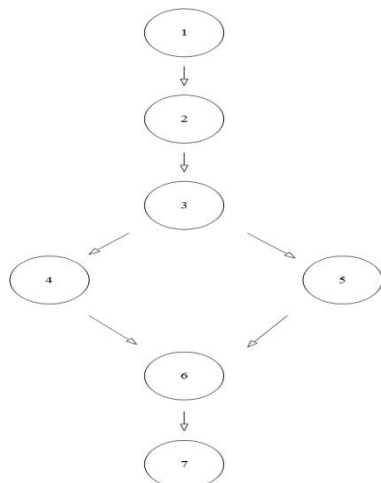
2. Instalasi aplikasi

- a. letakkan *file english for children.apk* di dalam memori *handphone*.
- b. Kemudian klik *file* tersebut hingga muncul *message box* pada *handphone android* untuk konfirmasi *install* dan terdapat pilihan *cancel* dan *install*, klik *install* dan terdapat pilihan untuk *install* yaitu *action package instal* dan *verifikasi* dan pemasangan. Klik *action package install* atau *verifikasi* dan pemasangan pilih salah satu kemudian *yes*.
- c. Kemudian klik *install* dan tunggu hingga aplikasi *terinstall* sukses.
- d. Ada dua pilihan jika aplikasi *terinstall* yaitu *finish* dan *open*, untuk *open* aplikasi akan langsung terbuka dan bisa dijalankan dan untuk *finish* aplikasi tidak langsung terbuka.
- e. Jika aplikasi *terinstall* sukses maka akan terdapat di menu pilihan *handpone* atau *desktop*.

3.5. Testing

Testing menggunakan pengujian *white box* dan *black box*, pengujian tersebut dilakukan pada aplikasi *english for children* antara lain :

1. **Pengujian White Box.** pengujian *white box* dengan menggunakan skema diagram alir dan pengujian *source code*, berikut merupakan diagram alir dan *source code* dari quiz pilihan ganda pada aplikasi *english for children*:



Gambar Skema diagram alir

Source code pada quiz pilihan ganda

```

public int nilai_01, nilai_02, nilai_03,
nilai_04, nilai_05, nilai_06,
nilai_07, nilai_08, nilai_09, nilai_10,
jumlah_nilai;
public static int nilai = 0;
public Spinner menu_jawab_01, menu_jawab_02,
menu_jawab_03, menu_jawab_04,
menu_jawab_05, menu_jawab_06, menu_jawab_07,
menu_jawab_08,
menu_jawab_09, menu_jawab_10;
public ImageButton cek_nilai;
public static int nilai_total = 0;
public String jawaban, hasil_jawaban,
isi_jawaban;
public static String data_jawaban = "";
int[] kunci = { 1, 2, 1, 1, 2, 1, 1, 2,
1, 2 };
int[] soal = { 1, 2, 3, 4, 5, 6, 7, 8, 9,
10 };
String[] jawaban_01 = { " Pilih jawaban", "A.
Dog", "B. Pig" };
String[] jawaban_02 = { " Pilih jawaban", "A.
Tiger", "B. Swan" };
String[] jawaban_03 = { " Pilih jawaban", "A.
Pig", "B. Bird" };
String[] jawaban_04 = { " Pilih jawaban", "A.
Chicken", "B. Snake" };
String[] jawaban_05 = { " Pilih jawaban", "A.
Goat", "B. Cat" };
String[] jawaban_06 = { " Pilih jawaban", "A.
Bear", "B. Lion" };
String[] jawaban_07 = { " Pilih jawaban", "A.
Seal", "B. Horse" };
String[] jawaban_08 = { " Pilih jawaban", "A.
Leopard", "B. Lion" };
String[] jawaban_09 = { " Pilih jawaban", "A.
Mouse", "B. Bull" };
String[] jawaban_10 = { " Pilih jawaban", "A.
Sheep", "B. Buffalo" };
@Override
protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_soal_1);
menu_jawab_01 = (Spinner)
findViewById(R.id.jwb_01);
menu_jawab_02 = (Spinner)
findViewById(R.id.jwb_02);
menu_jawab_03 = (Spinner)
findViewById(R.id.jwb_03);
menu_jawab_04 = (Spinner)
findViewById(R.id.jwb_04);
menu_jawab_05 = (Spinner)
findViewById(R.id.jwb_05);
menu_jawab_06 = (Spinner)
findViewById(R.id.jwb_06);
menu_jawab_07 = (Spinner)
findViewById(R.id.jwb_07);

```

```

menu_jawab_08 = (Spinner)
findViewById(R.id.jwb_08);
menu_jawab_09 = (Spinner)
findViewById(R.id.jwb_09);
menu_jawab_10 = (Spinner)
findViewById(R.id.jwb_10);
cek_nilai = (ImageButton)
findViewById(R.id.btn_check);
@Override
public void onNothingSelected(AdapterView<?>
arg0) {
}
};
}
public void btn_check_Clicked(View v) {
int pilihan_jawaban[] = { nilai_01, nilai_02,
nilai_03, nilai_04,
nilai_05, nilai_06, nilai_07, nilai_08,
nilai_09, nilai_10 };
int nomor_pilihan;
for (int i = 0; i < kunci.length; i++) {
if (pilihan_jawaban[i] == 1) {
jawaban = " A ";
} else if (pilihan_jawaban[i] == 2) {
jawaban = " B ";
} else {
jawaban = " - ";
}
if (kunci[i] == 1) {
isi_jawaban = "A";
} else if (kunci[i] == 2) {
isi_jawaban = "B";
} else {
isi_jawaban = "Tidak dijawab";
}
if (pilihan_jawaban[i] == kunci[i]) {
nilai_total = nilai_total + 10;
hasil_jawaban = jawaban + " (Benar) = " +
isi_jawaban
+ ", (Nilai = 10)\n";
} else {
hasil_jawaban = jawaban + " (Salah) = " +
isi_jawaban
+ ", (Nilai = 0)\n";
}
nomor_pilihan = i + 1;
data_jawaban = data_jawaban +
Integer.toString(nomor_pilihan)
+ ". " + hasil_jawaban;
}
Intent panggil_class = new Intent(this,
Hasil_Soal_1.class);
panggil_class.putExtra("nilai_hasil",
nilai_total);
panggil_class.putExtra("rinci_jawaban",
data_jawaban);
startActivity(panggil_class);
finish();
System.exit(0);
}

```

Kompleksitas siklomatis dari grafik alir dapat diperoleh dengan perhitungan:

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

Dimana:

$$V(G) = \text{Jumlah region}$$

E = jumlah Edge yang ditentukan dengan gambar panah

N = Jumlah simpul grafik alir ditentukan dengan gambar lingkaran

Sehingga didapat:

$$V(G) = 7 - 7 + 2 = 2$$

$V(G) < 10$ berarti memenuhi syarat kekompleksitasi siklomatisnya.

Baris set yang dihasilkandari jalur independent adalah sebagai berikut:

a. 1 - 2 - 3 - 4 - 6 - 7

b. 1 - 2 - 3 - 5 - 6 - 7

Ketika aplikasi dijalankan, maka terlihat bahwa satu set baris yang dihasilkan adalah 1 – 2 – 3 – 4 – 6 – 7 – 1 – 2 – 3 – 5 – 6 – 7 dan terlihat bahwa simpul telah dieksekusi satu kali.

2. **Pengujian Black Box.** Pengujian selanjutnya dilakukan dengan pengujian Black Box untuk memastikan bahwa suatu event atau masukan menjelaskan proses yang tepat dan menghasilkan output yang sesuai dengan rancangan.

Tabel Pengujian Black Box

INPUT /EVENT	PROSES	OUTPUT /NEXT STAGE	HASIL PENG UJIAN
Tombol about	public void btn_about_Clicked(View v) { stop_intro(); Intent panggil_class = new Intent(this, About.class); startActivity(panggil_class); }	Menampilkan tentang aplikasi English For Children	sesuai
Tombol menuju menu berikutnya	public void btn_menu_Clicked(View v) { stop_intro(); Intent panggil_class = new Intent(this, Menu_pilihan.class); startActivity(panggil_class); }	Menampilkan menu pilihan	Sesuai
Tombol menu alfabet	public void btn_alfabet_Clicked(View v) { stop_intro(); Intent panggil_class = new Intent(this, Alfabet.class); startActivity(panggil_class); }	Menampilkan menu Alfabet	Sesuai
Tombol menu animal	public void btn_animal_Clicked(View v) { stop_intro(); Intent panggil_class = new Intent(this, Menu_Animal.class); startActivity(panggil_class); }	Menampilkan menu hewan	sesuai
Tombol menu colour	public void btn_colour_Clicked(View v) { stop_intro(); Intent panggil_class = new Intent(this, Colour.class); startActivity(panggil_class); }	Menampilkan menu warna	sesuai
Tombol menu Number	public void btn_number_Clicked(View v) { stop_intro(); Intent panggil_class = new Intent(this, Number.class); startActivity(panggil_class); }	Menampilkan menu angka	sesuai
Tombol Exit	public void btn_exit_Clicked(View v) { AlertDialog.Builder builder = new AlertDialog.Builder(this);	Keluar dari aplikasi English For Children	sesuai

3.6. Support

Untuk memasang aplikasi ini digunakan perangkat atau media gadget berbasis Android minimal Android API 14 4.0 (Ice Cream Sandwich), RAM 512 MB, resolution 480x800, memory eksternal 1GB dan memori internal 1GB.

4. Kesimpulan

Berdasarkan hasil analisis, implementasi, dan pengujian terhadap aplikasi pembelajaran bahasa inggris usia dini yaitu english for children, maka dapat ditarik kesimpulan sebagai berikut :

1. Aplikasi edukatif *english for children* untuk pengenalan bahasa inggris usia dini dapat mempermudah dalam belajar bahasa inggris untuk anak dengan di dampingi orang tua atau guru, karena aplikasi ini menggunakan ponsel (*handphone*) berbasis *android* dan bisa belajar kapan saja dan dimana saja.
2. Materi pembelajaran yang disajikan mengenai keseharian yang dijumpai oleh anak-anak usia dini.
3. Aplikasi edukatif *english for children* untuk pengenalan bahasa inggris usia dini dibangun dengan menggunakan teknologi *java* berbasis *android* dan hanya dapat diimplementasikan pada ponsel (*handphone*) berbasis *android*.
4. *Eclipse IDE* dalam pembuatan aplikasi *english for children* mendukung untuk pengembangan aplikasi *android*, jika aplikasi *english for children* ingin dikembangkan.

DAFTAR PUSTAKA

- [1] Bambang Hariyanto.2011.Esensi-Esensi Bahasa Pemograman Java.Bandung. Informatika.
- [2] Dadang Marsa, Sardiarinto.2013.Pengenalan Bahasa Inggris Untuk Anak Melalui Aplikasi Edukasi Berbasis Android.ISSN:2089-9815. Yogyakarta: SENTIKA Maret 2013:156-162 Diambil Dari http://Fti.Uajy.Ac.Id/Sentika/Publikasi/Makalah/2013/2013_22.Pdf. (21Oktober 2014).
- [3] Dedi Rianto Rahadi.2014.Pengukuran Usability Sistem Menggunakan Use Questionnaire Pada Aplikasi Android.ISSN:2085-1588. Palembang: Jurnal Sistem Informasi Vol.6, No.1 April 2014:661-671 Diambil Dari: <http://Ejournal.Unsri.Ac.Id/Index.Php/Jsi/Article/Download/772/426>. (28 Oktober 2014).
- [4] Irawan.2012. Membuat Aplikasi Android Untuk Orang Awam. Palembang. Maxicom.
- [5] Linda Kartika Sari, Dimas Sasongko.2013.Media Pembelajaran Interaktif Bahasa Inggris Untuk Siswa Sekolah Dasar Kelas II.ISSN:2302-1136.Surakarta:FTI UNSA Vol.2, No.1 Maret 2013:23-27 Diambil Dari : [http:// download.portalgaruda.org/article.php?article=69470&val=4872](http://download.portalgaruda.org/article.php?article=69470&val=4872). (4 Desember 2014).
- [6] Rosa A.S, M.Shalahuddin.2013. Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek. Bandung. (Cetakan Kedua Juli 2014): Informatika.
- [7] Stephanus Hermawan S. 2011. Mudah Membuat Aplikasi Android. Yogyakarta.Andi.
- [8] Teguh Arifianto. 2011. Membuat Interface Aplikasi Android Lebih Keren Dengan LWUIT. Yogyakarta.Andi.
- [9] Yosef Murya.2014.Pemrograman Android Black Box.Purwokerto.Jasakom.