

Perancangan Aplikasi Quiz Menggunakan Metode Pengacakan Linear Congruential Generator (LCG) Berbasis Android

Riki Wahyudi, Hendra Handoko Syahputra Pasaribu

Program Studi S1 Teknik Informatika Fakultas Teknologi dan Ilmu Komputer

Universitas Prima Indonesia

Jl. Belanga No. 1 simp. Ayahanda

Email : rikiwahyudi2@gmail.com, hendra_pa1000@hotmail.com

Abstrak. Sebuah aplikasi quiz yang memiliki unsur *Mobile Learning (M-Learning)* yang ada dalam *smartphone* tentu dapat memberikan hiburan yang menarik kepada pengguna, serta dapat memberikan media pembelajaran untuk menambah wawasan tentang ilmu pengetahuan. Quiz merupakan bentuk permainan atau pikiran dimana pemain (sebagai individu atau dalam tim) berusaha untuk menjawab pertanyaan dengan benar dalam mencapai satu tujuan dalam pembelajaran. Pada aplikasi ini pengguna akan diberi pertanyaan tentang ilmu pengetahuan umum, sehingga pengguna dipaksa untuk berfikir dalam menjawab pertanyaan tersebut. Metode yang digunakan dalam perancangan aplikasi ini adalah metode *Linear Congruential Generator (LCG)* dan metode *String Matching (SM)*, dimana metode *LCG* berfungsi untuk mengacak pertanyaan yang ada pada aplikasi secara random, sedangkan metode *String Matching (SM)* berfungsi untuk pencocokan jawaban yang dipilih "true and false".

Kata Kunci : *Linear Congruential Generator, LCG, Quiz, Genius Quiz, Quiz Android.*

Abstract. A quiz application that has element of *Mobile Learning (M-Learning)* in the *Smartphone* can certainly provide an attractive entertainment to the users, and can provide instructional media to add knowledge. Quiz is a form of game or mind in which the players (as individuals or in teams) attempt to answer questions correctly to achieve the goal in learning. In this application, the users will be asked some questions about general knowledge, so the users are forced to think in answering the questions. The method used in the design of this application is the *Linear Congruential Generator (LCG)* method and the *String Matching (SM)* method, in which the *LCG* method functions to randomize question on the application, while the *SM* method functions to match the selected answer "true and false".

Keywords: *Linear Congruential Generator, LCG, Quiz, Genius Quiz, Quiz Android*

PENDAHULUAN

Keberadaan sistem operasi Android di Indonesia sudah sangat populer belakangan ini, karena tidak terlepas dari banyaknya aplikasi – aplikasi yang tersedia di *Android Market*, yang membuat pengguna mempunyai banyak pilihan untuk menggunakan aplikasi mana saja yang disukai dan terhitung gratis jika di dibandingkan dengan IPhone dan Windows Phone. Sebuah aplikasi *quiz* yang memiliki unsur *Mobile Learning (M-Learning)* yang ada dalam *smartphone* tentu dapat memberikan hiburan yang menarik kepada pengguna, serta dapat memberikan media pembelajaran untuk menambah wawasan tentang ilmu pengetahuan. *Quiz* merupakan bentuk permainan atau pikiran dimana pemain (sebagai individu atau dalam tim) berusaha untuk menjawab pertanyaan dengan benar dalam mencapai satu tujuan dalam pembelajaran. Oleh karena itu beberapa pendekatan terus dikembangkan untuk membuat sebuah aplikasi *quiz* yang dapat digunakan dalam perangkat bergerak atau *mobile*, salah satunya dengan menggunakan *android*.

Seiring dengan berkembangnya ilmu teknologi informasi, diiringi juga dengan berkembangnya metode edukasi, salah satunya menambah wawasan tentang ilmu pengetahuan dengan menggunakan aplikasi *quiz* yang terdapat pada *smartphone*. Bermain aplikasi *quiz* pada *smartphone* bukan salah satu cara untuk mengurangi tingkat kejenuhan. Akan tetapi, merupakan salah satu cara terbaik untuk menstimulasi otak, dimana pada saat ini sudah banyak aplikasi yang memiliki unsur edukasi. Contohnya, bermain aplikasi *quiz* ilmu pengetahuan umum, dimana pada permainan ini pengguna akan diberi pertanyaan tentang ilmu pengetahuan umum, sehingga pengguna dipaksa untuk berfikir

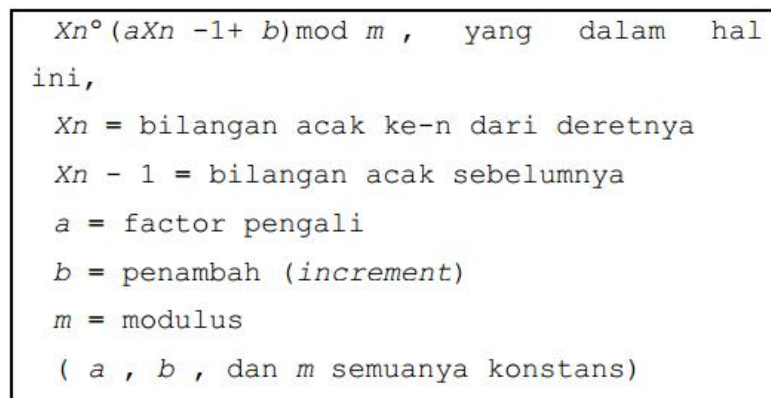
dalam menjawab pertanyaan tersebut. Metode yang digunakan dalam perancangan aplikasi ini adalah metode *Linear Congruential Generator (LCG)* dan metode *String Matching (SM)*, dimana metode *LCG* berfungsi untuk mengacak pertanyaan yang ada pada aplikasi secara *random*, sedangkan metode *String Matching (SM)* berfungsi untuk pencocokan jawaban yang dipilih “*true and false*”.

METODE

Quiz adalah bentuk permainan atau pikiran dimana pemain (sebagai individu atau dalam tim) berusaha untuk menjawab pertanyaan dengan benar. Di beberapa negara, *quiz* juga penilaian singkat yang digunakan dalam bidang pendidikan dan mirip dengan mengukur pertumbuhan dalam pengetahuan, kemampuan, atau keterampilan. *Quiz* juga digunakan sebagai metode permainan dalam mencapai satu tujuan dalam pembelajaran.

Metode *Linear Congruential Generators (LCG)*

Linear Congruential Generators (LCG) adalah salah satu pembangkit bilangan acak tertua dan sangat terkenal. LCG adalah algoritma yang sering diimplementasikan pada beberapa bahasa pemrograman untuk membangkitkan bilangan acak. LCG didefinisikan dalam relasi rekurens (Schneier, 1996) :



```
Xn = (aXn -1+ b)mod m , yang dalam hal
ini,
Xn = bilangan acak ke-n dari deretnya
Xn - 1 = bilangan acak sebelumnya
a = factor pengali
b = penambah (increment)
m = modulus
( a , b , dan m semuanya konstans)
```

Gambar 1. Relasi Rekurens

Kunci pembangkit adalah X^0 yang disebut *seed (secret seed)*. LCG mempunyai periode tidak lebih besar dari m , dan pada kasus periodenya kurang dari itu. LCG mempunyai periode penuh $(m-1)$ jika memenuhi syarat berikut :

1. b relative prima terhadap m
2. $a-1$ dapat dibagi dengan semua faktor prima dari m
3. $m > \max(a, b, x_0)$
4. $a > 0, b > 0$

Meskipun LCG secara teoritis mampu menghasilkan bilangan acak yang lumayan, namun sangat sensitive terhadap pemilihan nilai – nilai yang tidak sesuai dapat mempengaruhi implementasi pada LCG.

LCG tidak dapat digunakan untuk kriptografi karena bilangan acaknya dapat diprediksi urutan kemunculannya. Oleh karena itu LCG tidak aman digunakan untuk kriptografi. Namun demikian, LCG tetap berguna untuk aplikasi non-kriptografi seperti simulasi, sebab LCG memperlihatkan sifat statistik yang bagus dan sangat tepat (Putranto, 2010).

Metode *String Matching (SM)*

String Matching (SM) adalah pencarian sebuah *pattern* pada sebuah teks. Algoritma *string matching* adalah algoritma yang ditujukan untuk melakukan *string matching*. Prinsip kerja algoritma *string matching* adalah sebagai berikut:

1. Men-*scan teks* dengan bantuan sebuah *window* yang ukurannya sama dengan panjang *pattern*.
2. Menempatkan *window* pada awal *teks*.

3. Membandingkan karakter pada *window* dengan karakter dari *pattern*. Setelah pencocokan (baik hasilnya cocok atau tidak cocok), dilakukan *shft* ke kanan pada *window*. Prosedur ini dilakukan berulang-ulang sampai *window* berada pada akhir teks. Mekanisme ini disebut mekanisme *sliding-window*

Analisa

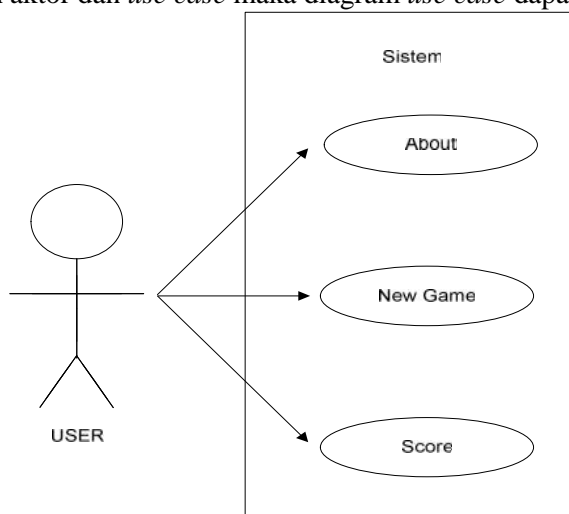
Aplikasi *Genius Quiz* yang dirancang merupakan aplikasi *quiz* yang termasuk ke dalam jenis *Mobile Learning (M – Learning)*, dimana aplikasi ini memiliki unsur edukasi serta dapat memberikan hiburan kepada pengguna. Aplikasi *Genius Quiz* dirancang untuk *smartphone* dengan sistem operasi Android, pengguna dituntut agar dapat menjawab soal – soal yang telah dipersiapkan pada aplikasi *quiz* tersebut, dan pengguna juga diberikan batas waktu maksimal dalam menjawab di setiap pertanyaan. Jika pengguna dapat menjawab soal dengan benar, maka pengguna akan memperoleh *point* dan akan lanjut ke soal berikutnya.

Analisa yang dimaksud disini adalah bagaimana cara memberikan suatu media pembelajaran yang menarik dan tidak membosankan kepada seluruh masyarakat, terutama anak – anak dengan cara memanfaatkan perkembangan teknologi saat ini. Dimana pada saat ini kebanyakan anak – anak lebih memilih bermain dari pada untuk belajar, sehingga minimnya ilmu pengetahuan yang dimengerti oleh anak – anak zaman sekarang. Maka dari itu, penulis ingin mencoba merancang suatu aplikasi *Genius Quiz*, yang dapat membantu anak – anak dengan bermain juga dapat memperoleh ilmu pengetahuan.

HASIL DAN PEMBAHASAN

Use Case Diagram

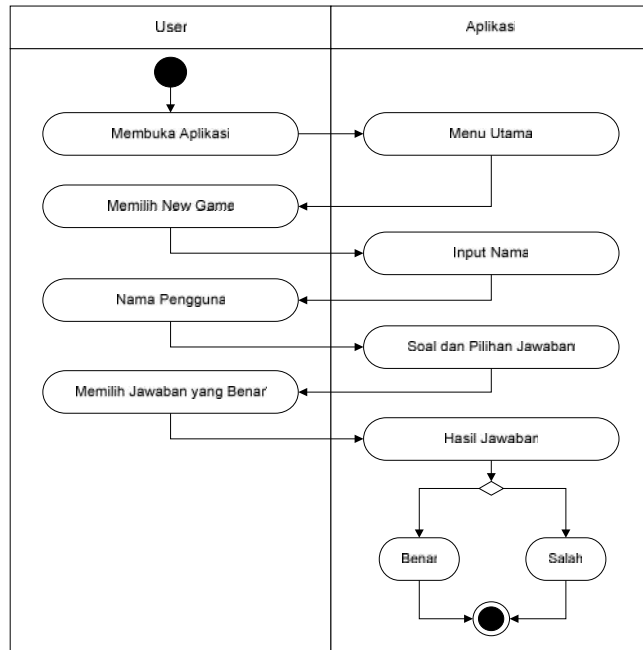
Untuk mengetahui aktor dan *use case* yang akan digunakan, maka dilakukan identifikasi *use case*. Setelah mendapatkan aktor dan *use case* maka diagram *use case* dapat digambarkan.



Gambar 3. Use Case Diagram Aplikasi Genius Quiz

Activity Diagram

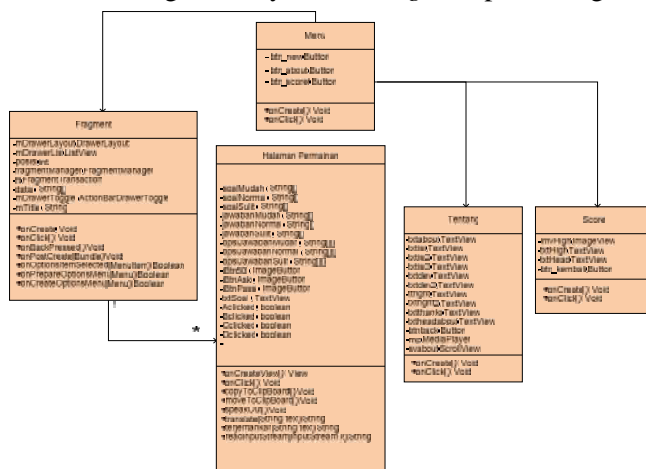
Activity diagram dirancang untuk menggambarkan berbagai alir aktivitas dalam aplikasi yang sedang dirancang, bagaimana masing – masing alir berawal hingga berakhir. Berikut adalah *activity diagram* pada perancangan aplikasi *Genius Quiz* :



Gambar 4. Activity Diagram Aplikasi

Class Diagram

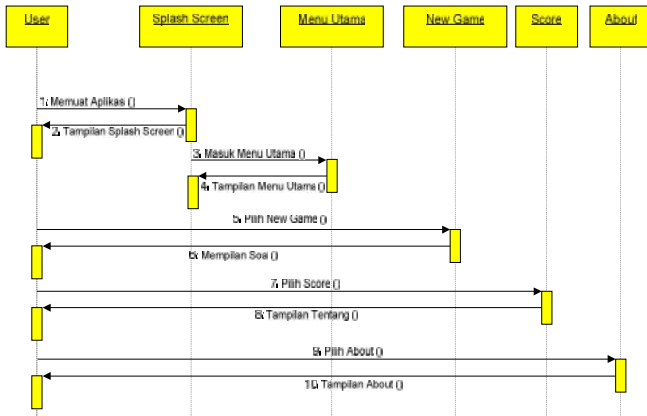
Class diagram dirancang untuk menentukan objek – objek yang dibutuhkan untuk perancangan aplikasi. Setiap kelas memiliki *attributes* dan *methods* masing – masing sesuai dengan kebutuhan kelas tersebut. Dengan adanya *class diagram*, perancangan aplikasi jelas berbasis Android.



Gambar 5. Class Diagram Aplikasi Genius Quiz

Sequence Diagram

Sequence diagram merupakan suatu diagram yang memperlihatkan atau menampilkan interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, display, dan sebagainya berupa pesan / *message*. Berikut adalah sequence diagram pada perancangan aplikasi *Genius Quiz* :



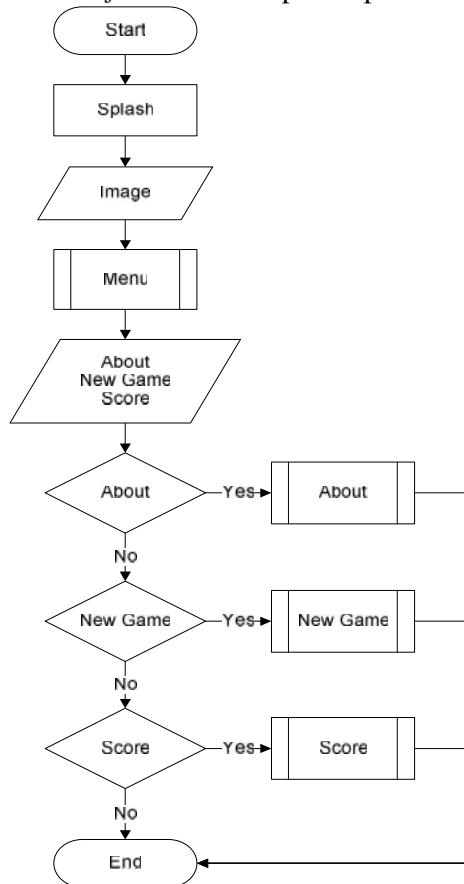
Gambar 6. Sequence Diagram Aplikasi Genius Quiz

Perancangan Flowchart

Perancangan *flowchart* merupakan proses awal hingga akhir dalam perancangan aplikasi *Genius Quiz* yang akan dibangun. Pada perancangan *flowchart* terbagi menjadi 5 bagian, yaitu proses perancangan desain, proses *new game*, proses cek jawaban, proses *about*, proses *score* dan proses metode.

Proses Perancangan Desain

Proses perancangan desain adalah awal yang harus dilakukan dalam membangun sebuah aplikasi, dimana desain dapat memberikan nilai tambah pada aplikasi tersebut. Dalam aplikasi ini memiliki desain yang sederhana, berikut adalah *flowchart* dari proses perancangan desain.

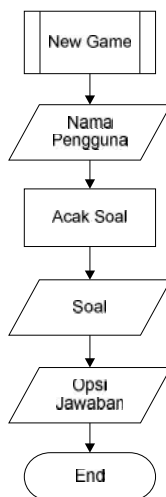


Gambar 7. Proses Perancangan Desain

Berikut adalah keterangan dari gambar 3.5 yaitu, proses awal aplikasi berjalan adalah munculnya *splash screen* yang memiliki durasi 3 detik, kemudian akan menampilkan *layout menu* yang berisi 3 menu pilihan, yaitu *new game*, *score*, dan *about*.

Proses New Game

Pada proses ini pengguna dapat melakukan permainan quiz tersebut, dimana pengguna akan memperoleh pertanyaan yang sebelumnya sudah diacak secara random. Berikut adalah *flowchart* dari proses *new game* :

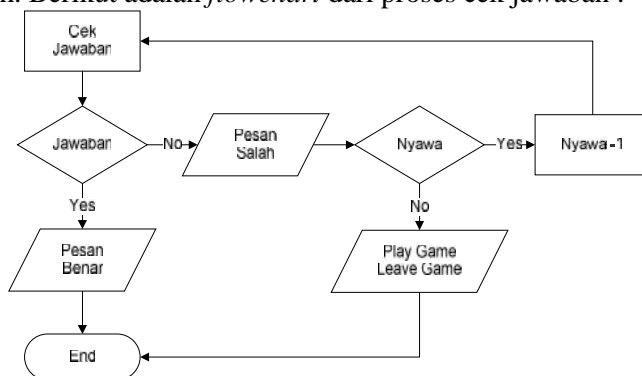


Gambar 8. Proses *New Game*

Berikut adalah keterangan dari gambar 3.6 yaitu, terlebih dahulu soal atau pertanyaan akan di acak, lalu soal dan pilihan jawaban akan di tampilkan ke *layout*.

Proses Cek Jawaban

Pada proses ini saat pengguna telah memilih dan menentukan jawaban yang benar dari keempat pilihan jawaban, maka jawaban yang dipilih akan di cek atau proses apakah jawaban yang dipilih tersebut benar atau salah. Berikut adalah *flowchart* dari proses cek jawaban :



Gambar 9. Proses Cek Jawaban

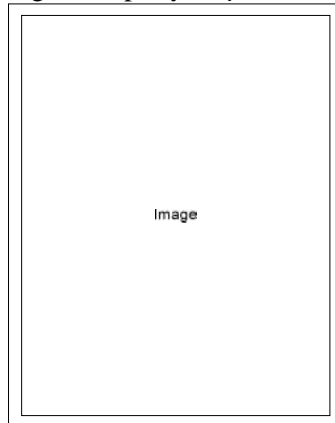
Berikut adalah keterangan dari gambar 3.7 yaitu, pada saat proses pengecekan jawaban terlebih dahulu jawaban yang dipilih akan diproses jika jawaban salah, maka akan muncul pesan salah dan nyawa akan -1 dan apabila sudah tidak memiliki nyawa lagi, maka akan muncul pesan pilihan “Play Game dan Leave Game”. Jika jawaban yang dipilih benar, maka akan muncul pesan benar, dan akan lanjut ke soal berikutnya.

Perancangan Interface

Perancangan *interface* adalah menjelaskan mengenai perancangan desain pada *layout* atau layar yang terdapat pada aplikasi. Diaman dalam perancangan *layout* pada aplikasi *Genius Quiz* terdiri dari 5 *layout* yaitu adalah sebagai berikut :

1. *Layout Splash Screen*

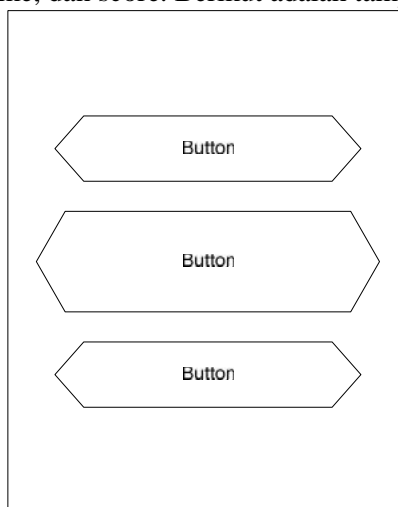
Sebagai halaman pembuka aplikasi akan tampil *layout splash screen* untuk memperindah tampilan desain. *Layout splash screen* ini akan tampil dengan interval waktu 5 detik dan akan secara otomatis akan disusul dengan tampilnya *layout menu*.



Gambar 10. *Layout Splash Screen*

2. *Layout Menu*

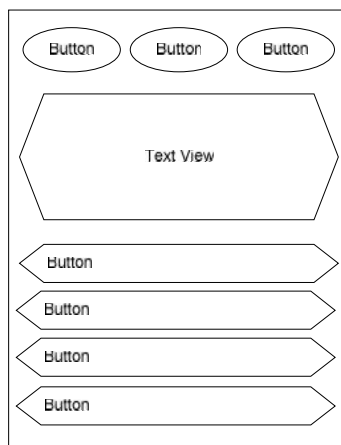
Sebagai menu utama pada aplikasi *Genius Quiz*, dimana pada *layout* ini terdapat 3 tombol button, yaitu about, new game, dan score. Berikut adalah tampilan desain pada *layout menu* :



Gambar 11. *Layout Menu*

3. *Layout Quiz*

Layout Quiz akan muncul setelah memilih *new game* pada *layout menu*. *Layout quiz* terdapat 1 *text view* yang berfungsi untuk menampilkan pertanyaan ke layar dan 7 tombol *button*, dimana 4 tombol *button* berfungsi untuk memilih jawaban yang benar, sedangkan sisahnya 3 tombol *button* untuk memilih bantuan dalam menjawab pertanyaan yang ada. Berikut adalah tampilan desain pada *layout quiz* :



Gambar 12. Layout Quiz

Implementasi

Adapun petunjuk umum dalam pengoperasian aplikasi *Genius Quiz* pada *smartphone* adalah sebagai berikut :

1. *Splash Screen*

Splash adalah tampilan gambar yang melambangkan aplikasi *game match*, yang pertama kali muncul pada saat *game match* dimainkan. Berikut adalah tampilan *splash*.



Gambar 13. Splash

2. *Menu*

Menu akan tampil secara otomatis setelah splash dengan durasi 3 detik. Dalam *layout menu* terdapat 3 tombol atau *button* yaitu *Play Game*, *About*, dan *Score*. Berikut adalah tampilan *menu*.



Gambar 14. Menu

3. *Input Nama Pengguna*

Ketika tombol *New Game* di pilih maka akan muncul *input* nama pengguna seperti pada gambar 4.9. Berikut adalah tampilan *New Game*.



Gambar 15. *Input Nama Pengguna*

4. *New Game*

Ketika tombol pengguna meng-input nama, maka akan tampil layar baru seperti pada gambar 4.10. Dimana pengguna dapat memilih menjawab pertanyaan – pertanyaan yang ada. Berikut adalah tampilan *New Game*.



Gambar 16. *New Game*

5. *About*

Pada *layout about* terdapat informasi – informasi yang merancang *Genius Quiz* dan yang ikut serta dalam membantu perancangan aplikasi sehingga *Genius Quiz* dapat diselesaikan sesuai dengan harapan.



Gambar 17. About

6. *Score*

Pada *score* terdapat 3 *score* tertinggi, dimana pengguna sejauh mana menjawab pertanyaan yang telah tersedia. Berikut adalah tampilan *score*.



Gambar 18. Score

KESIMPULAN DAN SARAN

Berdasarkan uraian dan pengujian yang telah dilakukan maka dapat di ambil kesimpulan yaitu Aplikasi *Quiz* ini dirancang dengan terfokus pada penyajian soal yang berupa soal pilihan ganda serta dengan tampilan *user interface* yang mudah dipahami sehingga pengguna akan mudah mengerti untuk menggunakan aplikasi ini. Metode *Linear Congruential* dapat melakukan pengacakan soal dengan kombinasi konstanta yang tepat maka akan dihasilkan pengacakan soal yang benar-benar acak.

DAFTAR PUSTAKA

- Suprianti Dan Agustina, 2012. Pemrograman Aplikasi Android, MediaKom.
Satyaputra Dan Aritonang, 2012 *Java For Beginners With Eclipse 4.2 Juno*, PT Elex Media Komputindo,.
Safaat, 2012. Pemrograman Aplikasi Mobile *Smartphone* dan *Tablet PC* Berbasis Android, Informatika Bandung.