



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1. Metodologi Penelitian

Dalam melakukan penelitian ini, terdapat tahap-tahap penelitian yang digunakan sebagai berikut.

1. Studi Literatur

Tahap pertama dalam penelitian ini adalah melakukan studi literatur mengenai Computer Based Test (CBT), Pseudo Random Number Generator, dan Linear Congruential Generator dengan referensi dari buku, jurnal, artikel dan berbagai sumber lainnya sebagai sumber untuk mendukung perancangan dan pembangunan aplikasi.

2. Pengumpulan Data

Bank soal yang digunakan adalah kumpulan soal teori yang sudah diseleksi oleh Course-Net Indonesia dari internet dan data hasil pengerjaan didapat dari pengalaman pengguna dalam mengerjakan berbagai variasi soal yang ditampilkan pada aplikasi. Nilai keberhasilan sistem didapat berdasarkan keberhasilan algoritma dalam mengacak soal tanpa terjadi pengulangan dan kondisi ambigu serta hasil pengumpulan tanggapan kuesioner kepuasan pengguna dalam menggunakan aplikasi.

3. Perancangan Desain Sistem

Perancangan dibuat dengan pemanfaatan algoritma *Linear Congruential Generator* untuk mengacak soal yang diambil dari database soal. Setelah mendapatkan semua data yang dibutuhkan, dilakukan perancangan desain sistem

untuk menentukan keberhasilan algoritma dalam melakukan pengacakan soal pada aplikasi.

4. Pembangunan Sistem dan Penerapan Algoritma

Merancang sebuah aplikasi berbasis android dan memanfaatkan algoritma *Linear Congruential Generator* untuk pengacakan soal yang akan ditampilkan pada aplikasi. Pendalaman algoritma dilakukan dengan melakukan pemilihan nilai konstanta a , b , dan m yang baik sehingga tidak ditemukan kondisi gagal dan ambigu dalam pengacakan soal. Jika hasil pengacakan soal yang ditampilkan tidak menghasilkan pengulangan dan nilai minus, maka dapat disimpulkan penerapan algoritma berhasil.

5. Uji Coba

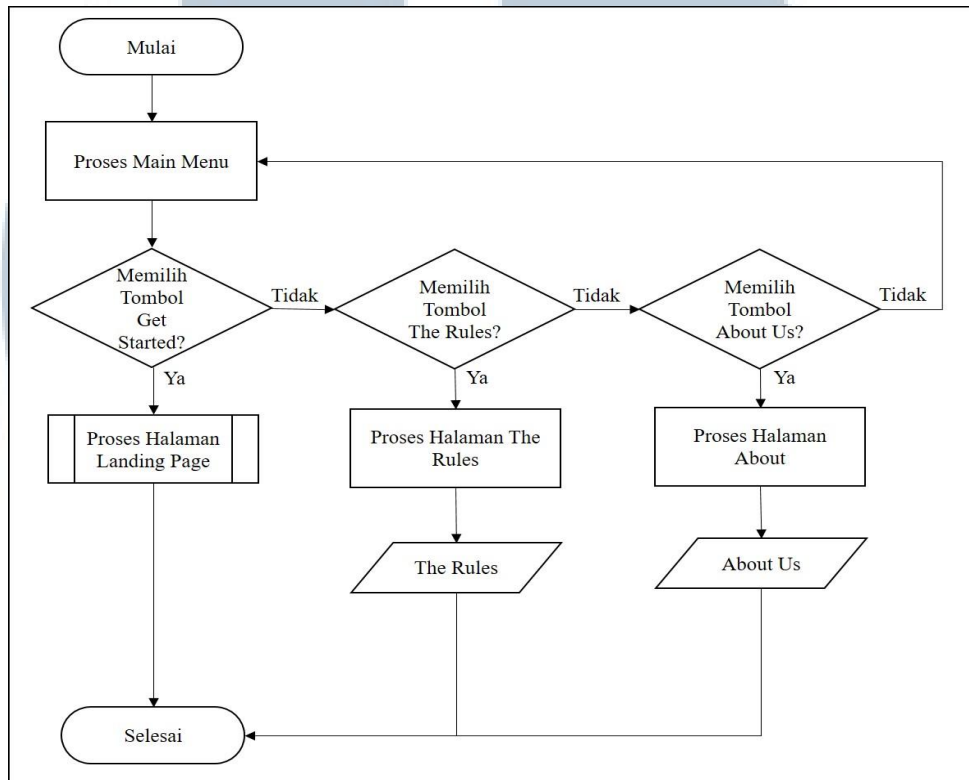
Setelah selesai dibangun, algoritma akan diuji keberhasilannya dalam mengacak 40 soal yang ditampilkan tidak terjadi pengulangan dan terjadi kondisi ambigu. Pengujian algoritma dilakukan melalui 3 cara, melalui perhitungan manual menggunakan *Microsoft Excel*, penerapan algoritma untuk menampilkan deret bilangan acak pada Dev C++, dan penerapan algoritma untuk menampilkan soal dari database pada aplikasi android. Setelah itu, aplikasi akan diujikan kepada pengguna untuk dinilai keberhasilan algoritma dalam mengacak soal.

3.2. Perancangan Sistem

Perancang sistem dalam penelitian ini dilakukan perancangan awal terlebih dahulu untuk memahami alur kerja dari sistem. Perancangan sistem yang dibuat adalah sebagai berikut.

3.2.1. Flowchart

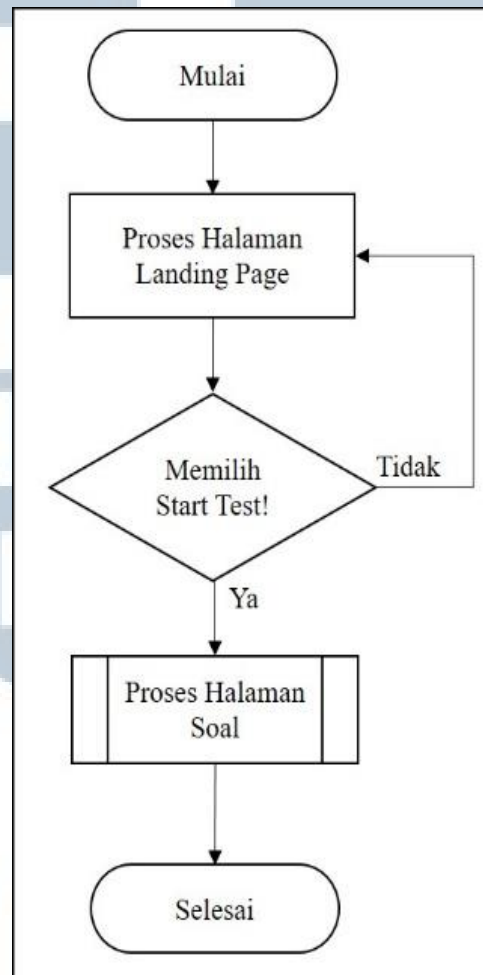
Flowchart digunakan untuk memberikan gambaran mengenai urutan proses yang terdapat pada aplikasi. Alur kerja aplikasi diawali pada *flowchart* berikut ini.



Gambar 3. 1 *Flowchart* proses halaman *main menu*

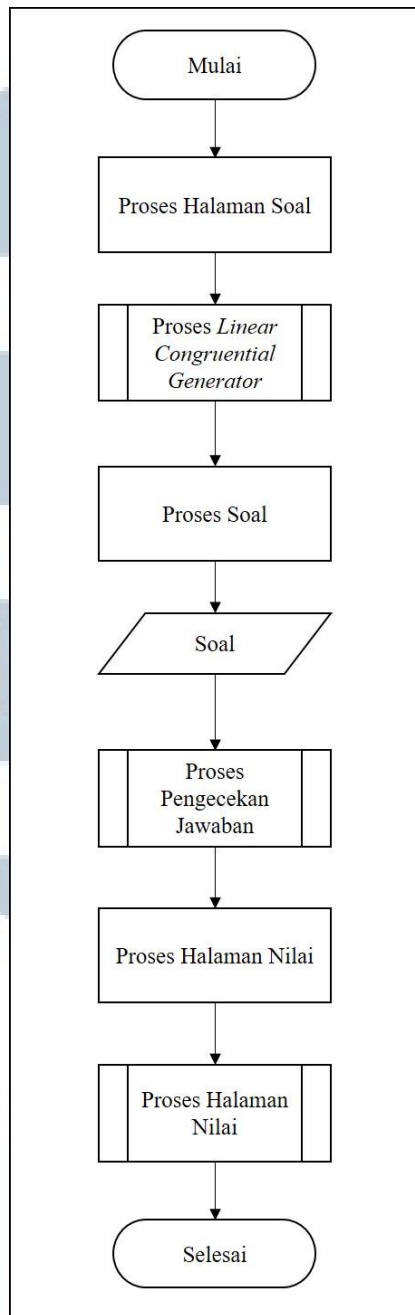
Gambar 3.1 menunjukkan alur yang dihadapi pengguna saat menjalankan aplikasi. Jika memilih tombol *Get Started*, user akan diarahkan pada proses halaman *landing page*. Jika memilih tombol *The Rules*, user akan diarahkan pada proses halaman *The Rules*. Jika user memilih tombol *About Us*, akan diarahkan pada proses halaman *About Us*. Halaman *The Rules* akan menampilkan aturan yang akan dihadapi dalam mengerjakan soal yang diberikan oleh aplikasi. Jika memilih *about us*, user akan dialihkan pada halaman *about us* yang berisi tentang data penyusun aplikasi.

Pilihan lain yang dapat diinput oleh *user* adalah *button get started*. Proses selanjutnya ketika *button get started* dipilih adalah menampilkan halaman *landing page*. Alur proses halaman *landing page* dijelaskan pada Gambar 3.2.



Gambar 3. 2 *Flowchart* proses halaman *landing page*

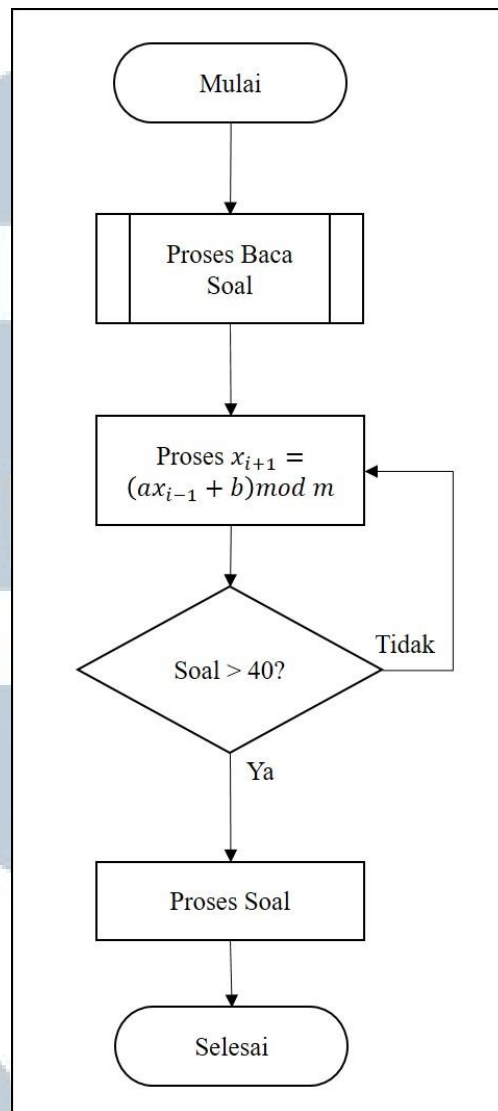
Halaman *landing page* menampilkan sedikit penjelasan mengenai jenis latihan soal sertifikasi yang akan ditampilkan oleh aplikasi. Ketika *user* memilih *button start test*, *user* akan dialihkan pada proses penampilan soal.



Gambar 3. 3 *Flowchart* proses halaman soal

Gambar 3.3 menunjukkan tahapan proses dari penampilan halaman soal.

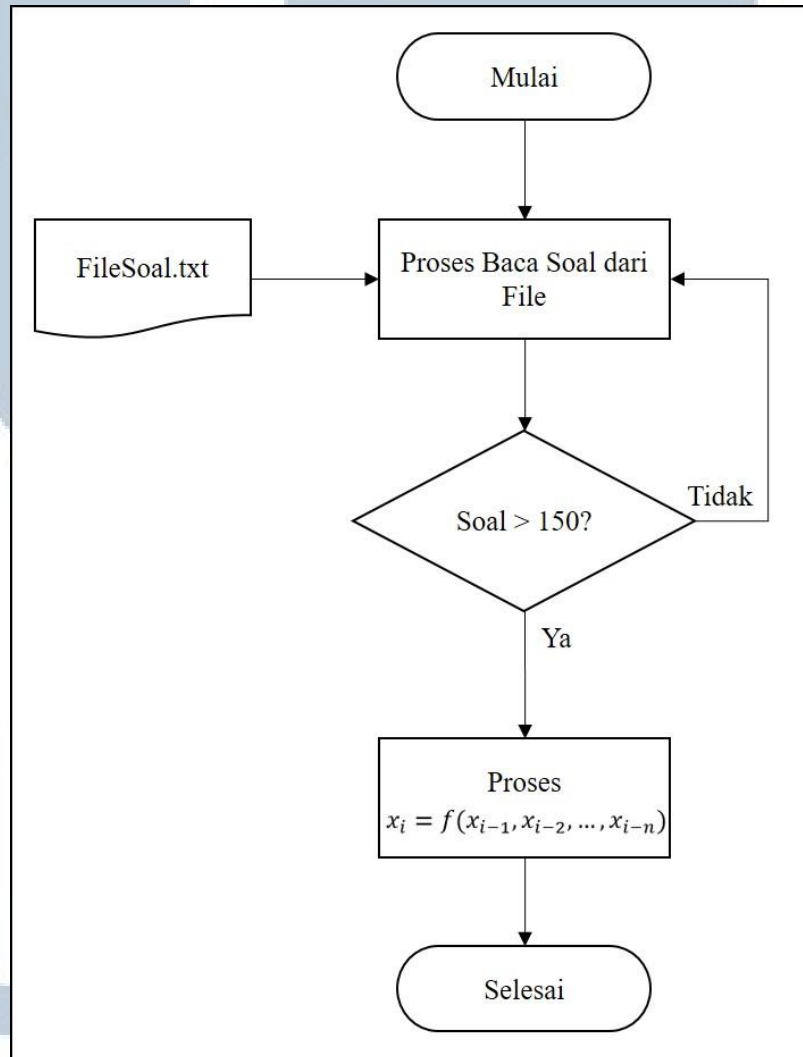
Terdapat beberapa modular seperti, proses pengacakan menggunakan algoritma *Linear Congruential Generator*, proses pengecekan jawaban, hingga proses halaman nilai.



Gambar 3. 4 *Flowchart* proses algoritma

Gambar 3.4 menunjukkan proses menghasilkan deret 40 soal pada aplikasi dengan algoritma *Linear Congruential Generator*. Sebelum nilai x_i dipanggil untuk dimasukkan ke dalam penghitungan rumus algoritma *Linear Congruential Generator*, sistem melakukan proses pembacaan soal pada *external file* untuk pemanggilan soal.

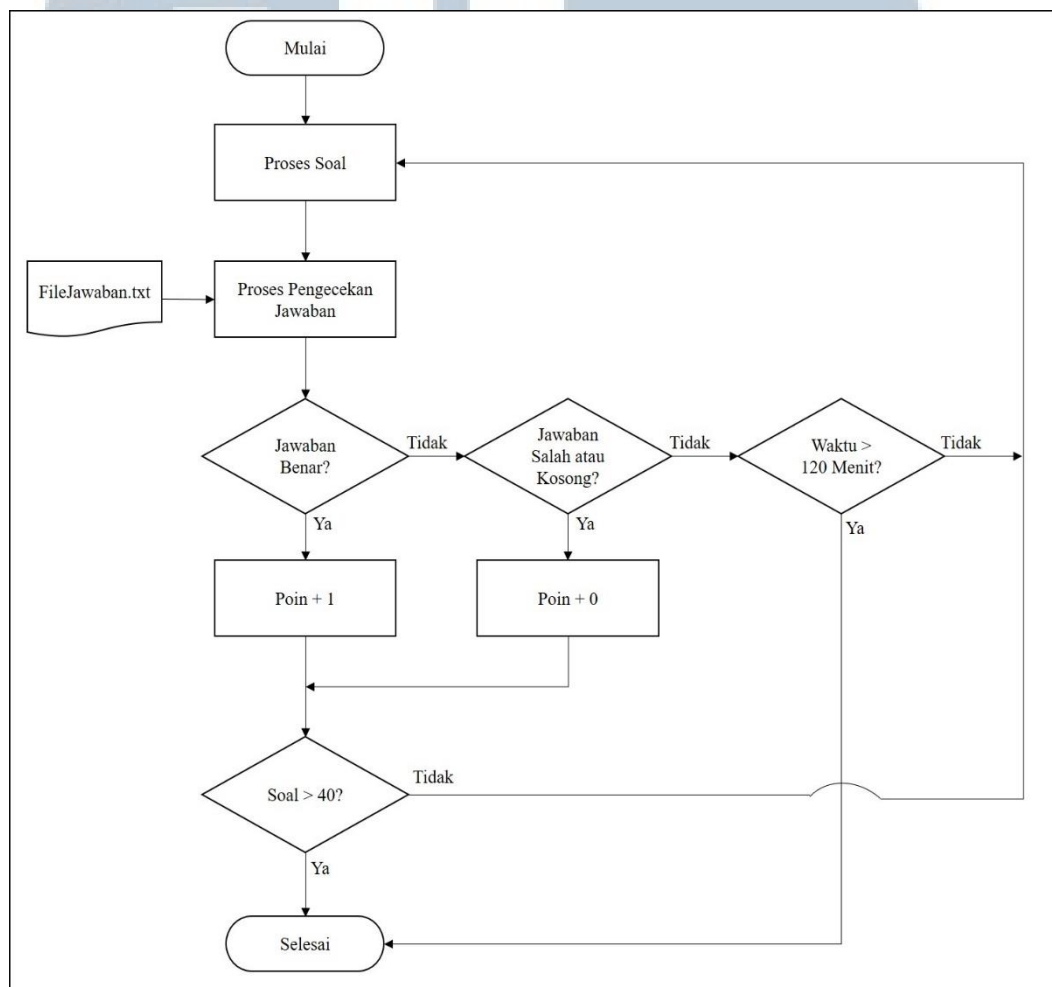
Gambar 3.5 berikut menunjukkan proses pembacaan semua soal dari *external file: FileSoal.txt*. Setelah seluruh soal terbaca oleh sistem, proses selanjutnya adalah proses pencarian nilai x_i menggunakan algoritma *Pseudo Random Number Generator*.



Gambar 3. 5 Flowchart proses baca soal

Gambar 3.6 berikut menunjukkan proses pengecekan jawaban yang di-input oleh pengguna. Jika pengguna memilih jawaban yang benar, maka sistem akan menandakan poin untuk soal yang terjawab dengan benar ditambah satu. Jika pengguna memilih jawaban yang salah atau tidak mengisi, maka sistem akan

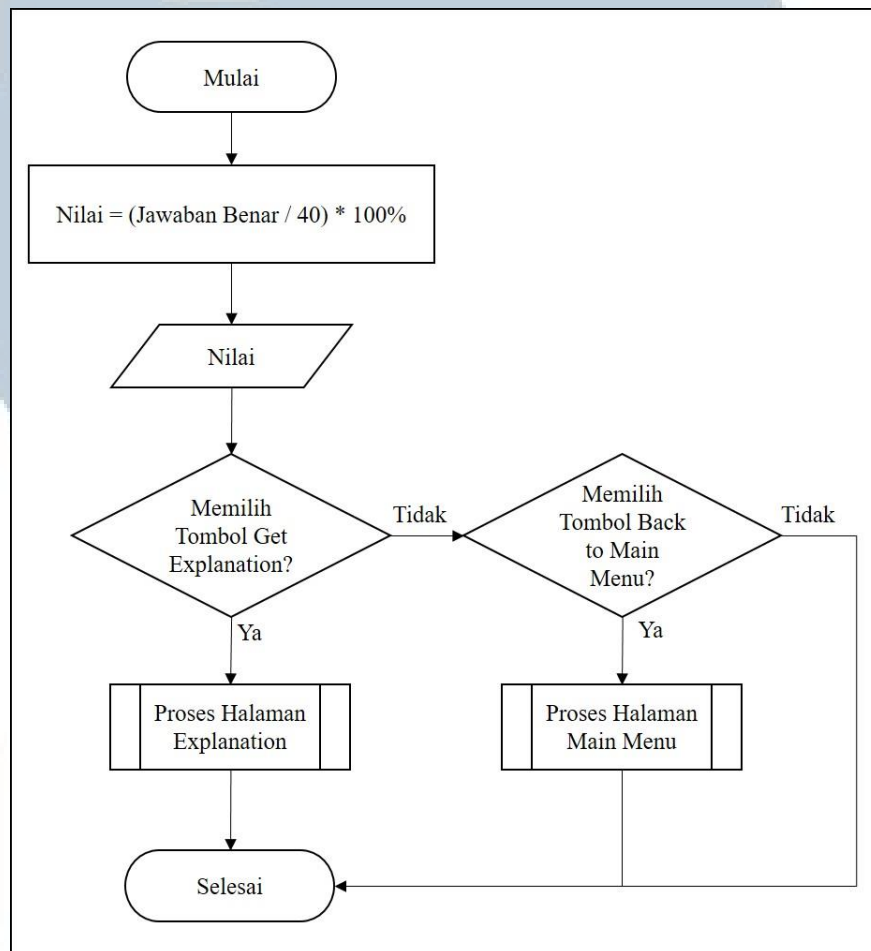
menandakan poin untuk soal tersebut ditambah nol atau kosong. Jika pengguna berada pada soal tersebut hingga melebihi batas waktu 120 menit, maka sistem akan menghitung semua poin yang tersimpan untuk mendapatkan nilai akhir pengerjaan. Lalu, jika soal yang ditampilkan belum melebihi soal ke 40, maka sistem akan memproses soal selanjutnya dan sebaliknya, sistem akan meneruskan pengguna ke halaman nilai.



Gambar 3. 6 Flowchart proses pengecekan jawaban

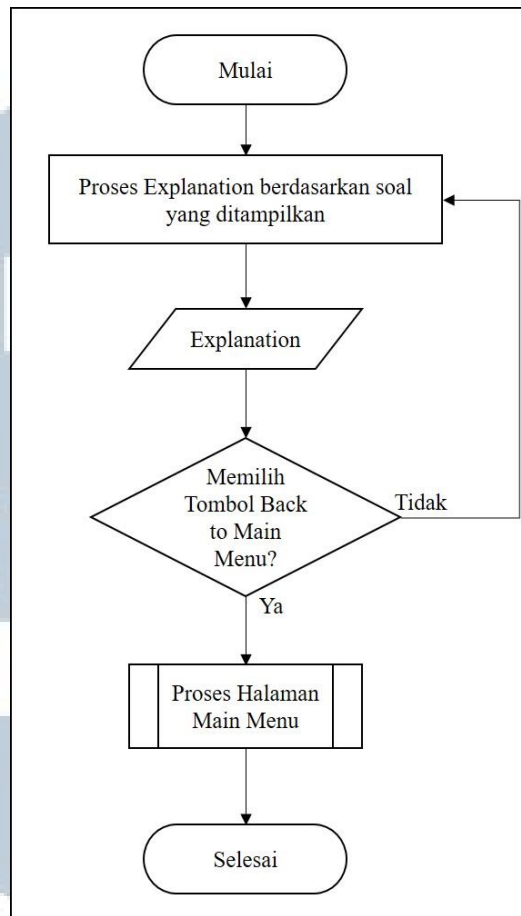
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.7 berikut menunjukkan proses penampilan halaman nilai yang diraih oleh pengguna setelah mengerjakan 40 soal yang ditampilkan. Pada halaman ini terdapat tombol *Get the explanation* untuk menampilkan penjelasan dari soal yang dikerjakan dan tombol *Back to Main Menu* untuk kembali ke halaman *main menu*.



Gambar 3. 7 Flowchart proses halaman nilai

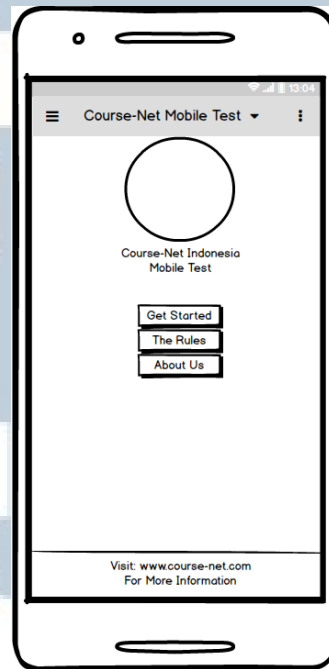
Gambar 3.8 berikut menunjukkan proses penampilan halaman *explanation* atau penjelasan dari soal yang sudah dikerjakan pengguna. Halaman ini berupa tampilan penjelasan dan tombol *Back to Main Menu* yang berfungsi sama seperti pada halaman nilai.



Gambar 3. 8 *Flowchart proses halaman explanation*

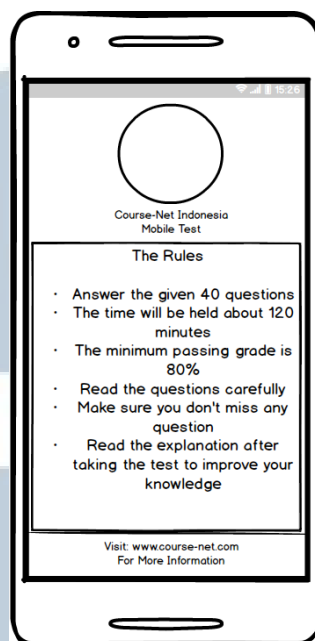
3.2.2. Rancangan *User Interface* Sistem

Proses selanjutnya adalah proses perancangan tampilan antarmuka aplikasi. Tujuannya adalah memberikan gambaran tampilan antarmuka untuk *user* yang menggunakan aplikasi. Gambar 3.9 menjelaskan tampilan antarmuka saat aplikasi dijalankan.



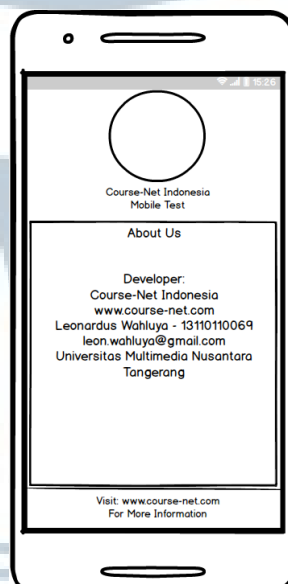
Gambar 3. 9 Rancangan antarmuka *main menu*

Terdapat beberapa pilihan navigasi bagi *user*. Tombol *Get Started* akan mengalihkan *user* ke halaman *Landing Page*. Tombol *The Rules* akan mengalihkan *user* ke halaman tampilan *The Rules*. Tombol *About Us* akan mengalihkan *user* ke halaman tampilan *About Us*.



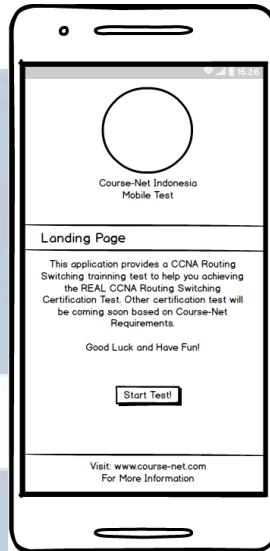
Gambar 3. 10 Rancangan antarmuka *the rules*

Gambar 3.10 adalah rancangan antarmuka pada halaman *The Rules*. Pada halaman ini hanya menampilkan logo dari aplikasi dan berisi teks singkat tentang aturan sederhana dalam menggunakan aplikasi.



Gambar 3. 11 Rancangan antarmuka *about us*

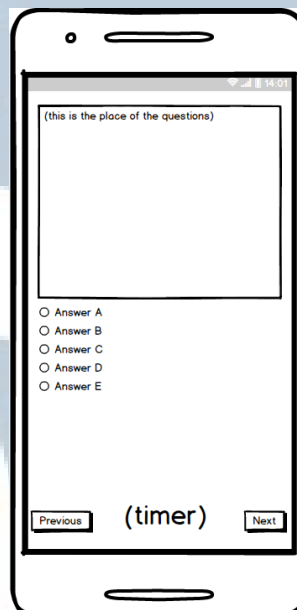
Gambar 3.11 adalah rancangan antarmuka pada halaman *about us*. Sama seperti halaman *the rules*, pada halaman ini hanya menampilkan logo dari aplikasi dan teks berisi data penyusun aplikasi.



Gambar 3. 12 Rancangan antarmuka *landing page*

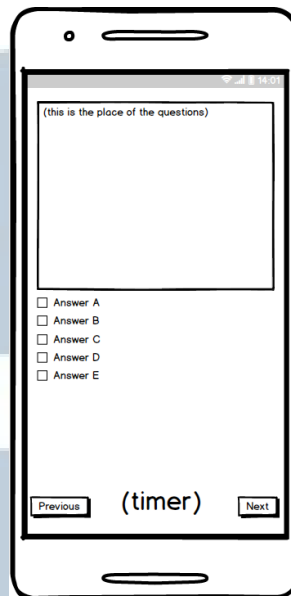
Gambar 3.12 menunjukkan rancangan antarmuka halaman *landing page*.

Halaman ini menampilkan sedikit informasi mengenai soal tes yang akan diberikan.



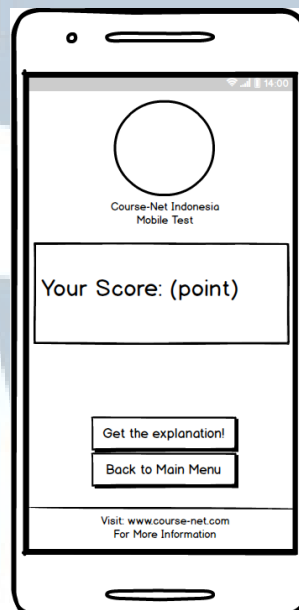
Gambar 3. 13 Rancangan antarmuka soal tipe *Radio Button*

Gambar 3.13 adalah rancangan antarmuka soal tipe pertanyaan dengan satu jawaban menggunakan navigasi *radio button*. Terdapat *timer*, untuk membatasi waktu pengerjaan serta tombol *next* dan *previous*.



Gambar 3. 14 Rancangan antarmuka halaman soal *Checkbox*

Gambar 3.14 adalah rancangan antarmuka soal tipe pertanyaan dengan beberapa jawaban menggunakan navigasi *check box*



Gambar 3. 15 Rancangan antarmuka halaman nilai

Gambar 3.15 adalah rancangan antarmuka untuk menampilkan hasil dari pengerjaan pada aplikasi. Tombol *Get the explanation*, yang berfungsi menampilkan jawaban yang benar dan penjelasannya, dan tombol *Back to Main Menu*, yang berfungsi mengembalikan *user* ke halaman *main menu*.



Gambar 3. 16 Rancangan antarmuka halaman *explanation*

Gambar 3.16 adalah rancangan antarmuka untuk menampilkan jawaban yang benar beserta penjelasannya. Tampilan diambil berdasarkan soal yang diberikan sebelumnya dan ditampilkan dalam bentuk paragraf. Terdapat tombol *Back to Main Menu* yang fungsinya sama dengan yang terdapat pada halaman nilai.

