

BAB IV

PERANCANGAN

Dalam penulisan ini penulis mencoba untuk merancang aplikasi Kebutaan warna berbasis android yaitu sebuah aplikasi yang menampilkan soal-soal tes isihara dengan menggunakan *smartphone* Android

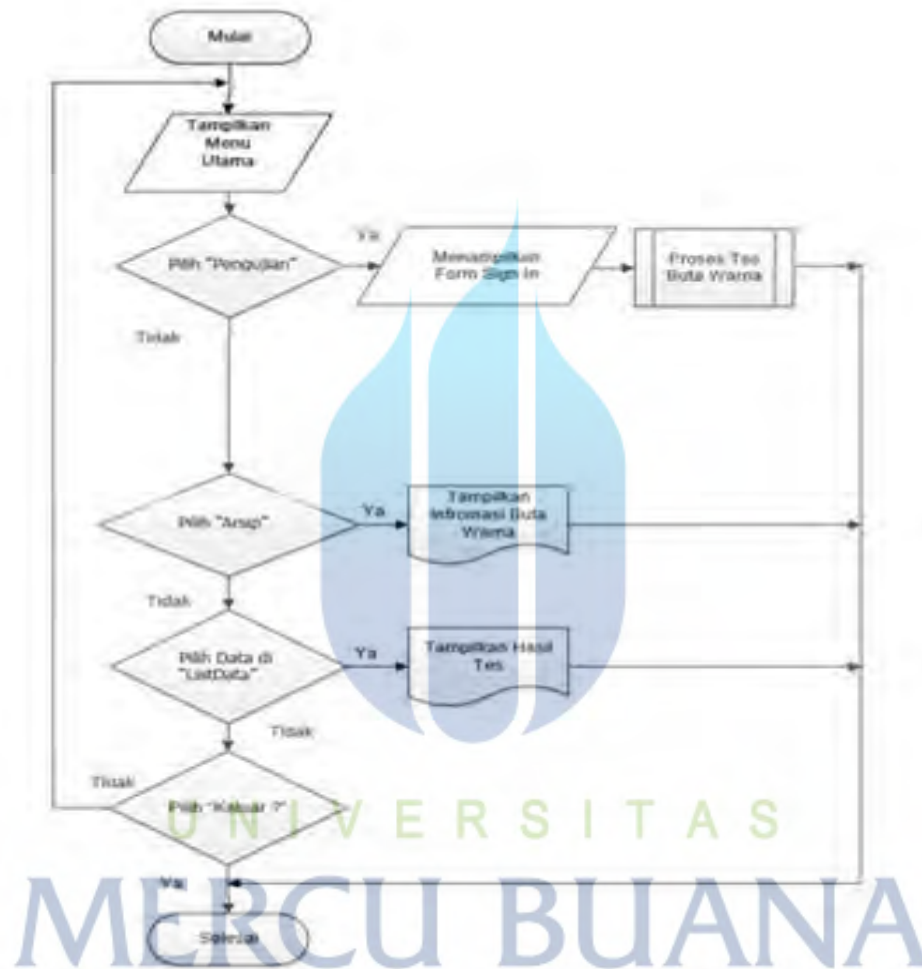
4.1. PERTIMBANGAN PERANCANGAN SISTEM

Sebelum membangun aplikasi ada beberapa hal yang harus diperhatikan dan dipertimbangan terlebih dahulu agar pada saat implementasi aplikasi berhasil dibangun, mudah digunakan dan sesuai tujuan, dan analisa yang telah dilakukan. Beberapa yang harus dipertimbangkan sebelum implementasi adalah diagram alur, *use case*, *activity diagram*.

4.1.1. Diagram Alur Aplikasi

Flowchart merupakan diagram yang menggambarkan aliran dokumen pada suatu prosedur kerja di organisasi dan memperlihatkan diagram alir yang menunjukkan arus dari dokumen, aliran data fisik, entitas-entitas sistem informasi dan kegiatan operasi yang berhubungan dengan sistem informasi. Penggambaran biasanya diawali dengan mengamati dokumen apa yang menjadi media data atau informasi. Selanjutnya ditelusuri bagaimana dokumen tersebut terbentuk, kebagian atau entitas mana dokumen tersebut mengalir, perubahan apa yang terjadi pada dokumen tersebut, proses apa yang terjadi terhadap dokumen tersebut, dan seterusnya (Rosa.A.S, 2011).

Diagram ini bisa memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut. *Flowchart* menu utama aplikasi menggambarkan alur proses menu utama aplikasi yang akan dilakukan aplikasi.



Gambar 4.1. *Flowchart* Aplikasi Tes Buta Warna

Berikut akan dijelaskan secara detail mengenai *Flowchart* aplikasi tes buta warna sebagai berikut :

Memilih menu tes buta warna, sebelum *User* melakukan tes buta warna *User* akan diperintahkan untuk melakukan proses sign in.

Setelah *User* melakukan sign in *User* dapat melakukan sebuah tes buta warna, dimana sistem akan menampilkan soal tes buta warna dan *User* menginputkan angka yang yang terlihat pada soal *plate* pada layar, setelah melakukan tes buta warna maka *User* akan menerima hasil diagnosa. Hasil dari tes buta warna metode isihara ini meliputi hasil presentase dari mata normal, buta warna parsial dan buta warna total.

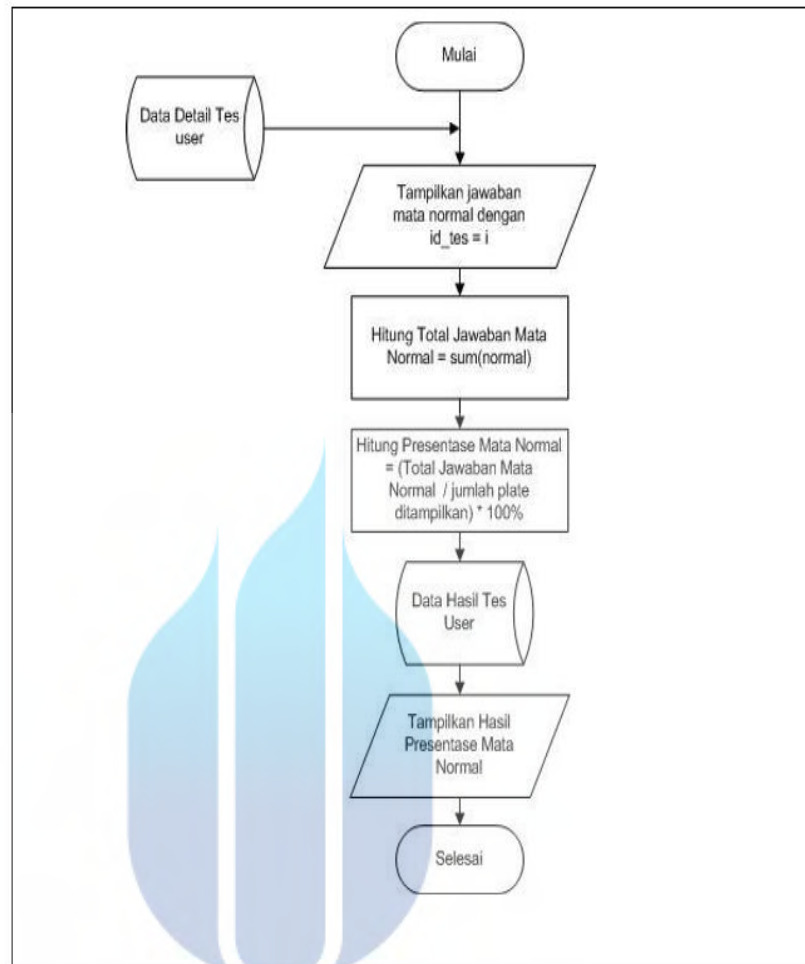
Memilih menu informasi tes buta warna, sistem akan menampilkan informasi buta warna. Informasi yang ditampilkan meliputi informasi buta warna normal, parsial dan buta warna total. Memilih menu hasil tes, sistem akan menampilkan hasil tes buta warna dari setiap *User* (*id_User*). Hasil tes yang ditampilkan berupa presentase dari mata normal, buta warna parsial dan buta warna total. Memilih keluar dari aplikasi, maka sistem akan mengeluarkan *User* dari aplikasi tes buta warna isihara tersebut.

4.1.2. Flowchart Hitung Presentase

Diagnosa *Flowchart* Hitung Presentase Diagnosa akan menggambarkan alur untuk menghitung presentase dari setiap diagnose yaitu presentase mata normal, presentase buta warna normal, buta warna parsial dan buta warna total. 1. *Flowchart* Menghitung

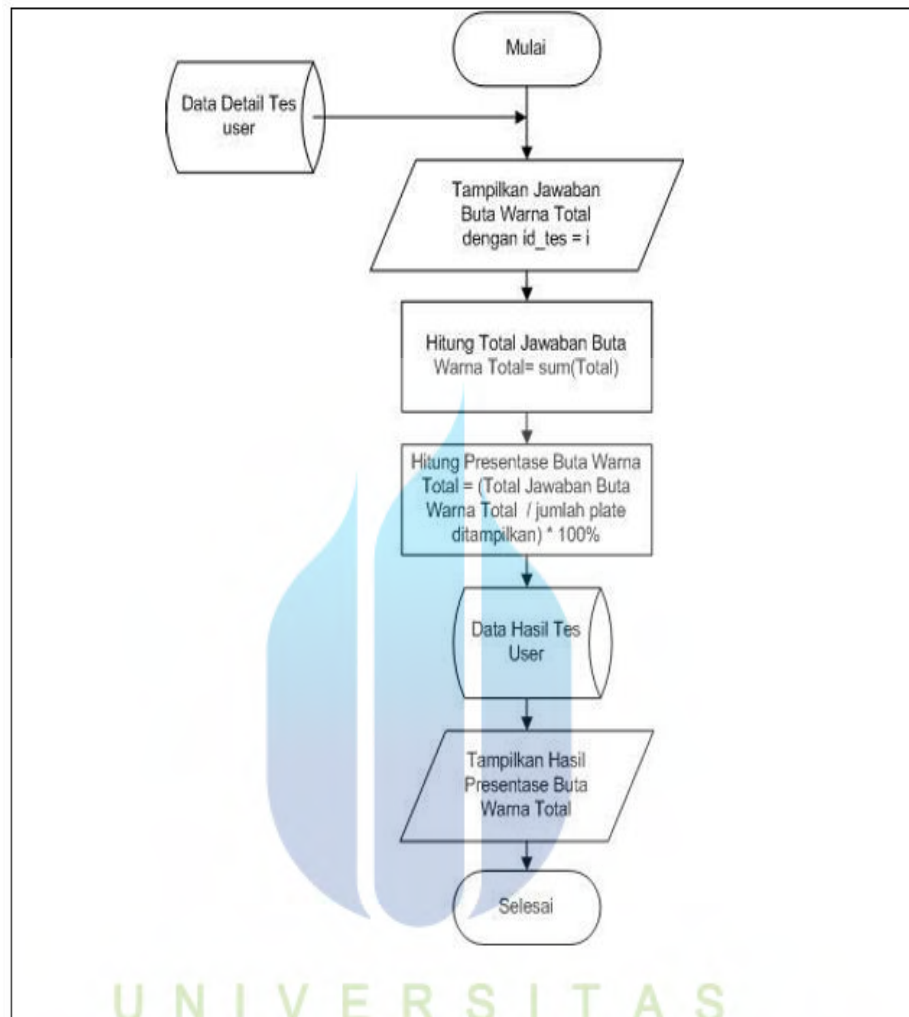
Presentase Mata Normal *Flowchart* menghitung presentase mata normal adalah proses untuk menghitung nilai presentase mata normal *User*. *Flowchart* Menghitung Presentase Mata Normal sebagai berikut:

- A. Menampilkan semua nilai pada *Tabel tb_detail_tes* kolom normal berdasarkan $id_tes = a$.
- B. Menghitung jumlah jawaban mata normal pada kolom normal (*Tabel tb_detail_tes*), berdasarkan $id_tes = a$.
- C. Menghitung presentase mata normal dengan cara menghitung jumlah mata normal, di bagi jumlah *plate*, kemudian di kalikan 100%. Simpan hasil presentase mata normal pada *Tabel tb_tes*



Gambar 4.2. *Flowchart* Menghitung Presentase Mata Normal

Flowchart Menghitung Presentase Buta Warna Total *Flowchart* menghitung presentase buta warna total adalah proses untuk menghitung nilai presentase buta warna total *User*. Alur kerja *Flowchart* Menghitung Presentase Buta Warna Total.

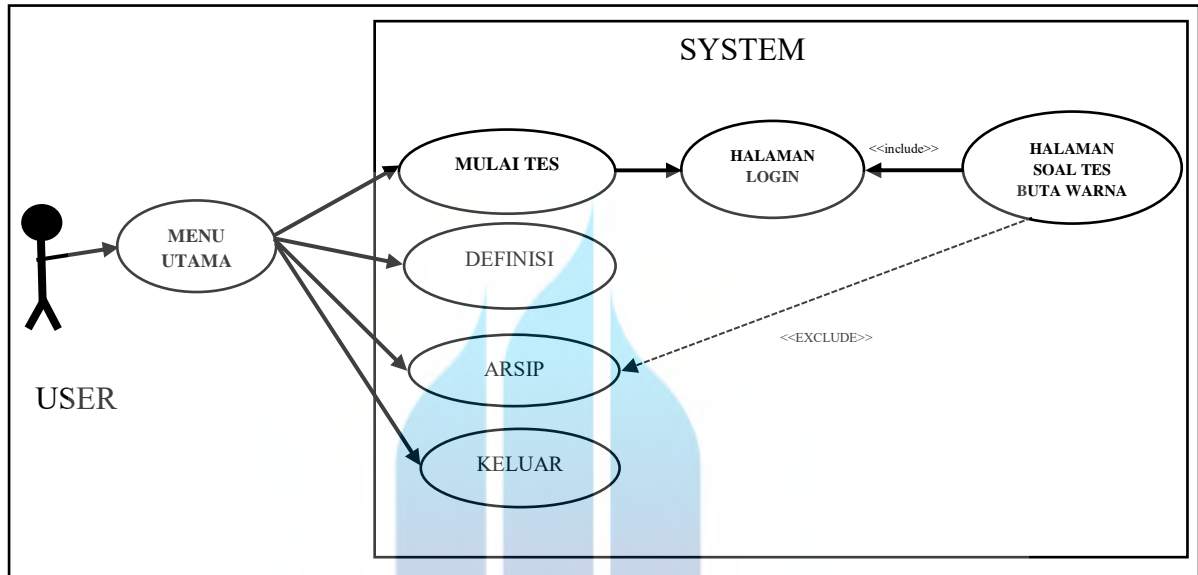


Gambar 4.3. Flowchart Menghitung Presentase Buta Warna Total

4.1.3. Use Case Diagram

Use case mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan. Selain mendeskripsikan *use case* secara langsung, kita juga bisa menjabarkan melalui skenario. Skenario adalah rangkaian langkah-langkah yang menjabarkan sebuah interaksi antara seorang pengguna dengan sebuah sistem. *Use case* merupakan

sebuah piranti yang berharga untuk membantu memahami persyaratan fungsional sebuah sistem (Speerman Roberts, 2009) .



Gambar 4.4. Diagram Use Case Aplikasi Tes Buta Warna Ishihara

Fitur yang dapat dipilih oleh *User* diantaranya saat *User* memilih menu Tes Buta Warna selanjutnya *User* diharapkan melakukan Sign Up untuk register *User* dan Sign In untuk dapat masuk ke halaman tes buta buta warna. *User* dapat memilih menu Informasi Buta Warna untuk melihat jenis buta warna, *User* dapat memilih menu Hasil Tes untuk menampilkan hasil tes buta warna dari setiap *User* dan memilih menu Keluar untuk keluar dari aplikasi tes buta warna.

4.1.4. Activity Diagram

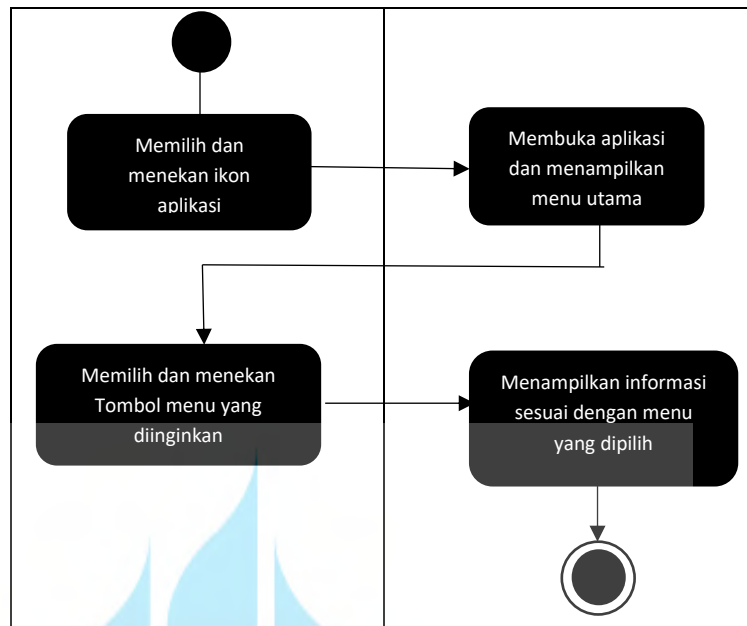
Activity Diagram merupakan suatu penggambaran tentang berbagai alur aktivitas dalam suatu system yang dirancang, bagaimana

alur berawal, *decision* yang terjadi dan bagaimana sistem berakhir. Pada dasarnya *activity diagram* memiliki kemiripan dengan *flowchart diagram*. *Activity diagram* memodelkan event –event yang terjadi pada *use case* dan digunakan untuk permodelan aspek dinamis dari sistem.

4.1.4.1. Activity Diagram Menu

Activity diagram menu menggambarkan aktivitas ketika pengguna memilih menu. Pengguna memilih salah satu menu, selanjutnya sistem akan menampilkan konten yang sesuai dengan jenis menu yang dipilih. Terdapat beberapa tombol menu yang akan dibangun pada tampilan utama GUI (*Graphical User Interface*), hal ini bertujuan untuk mempermudah pengguna dalam pemakaian aplikasi. Berbagai informasi disediakan untuk memberikan wawasan tentang cara penggunaan dan tujuan aplikasi. Berbagai informasi disediakan untuk memberikan wawasan tentang cara penggunaan dan tujuan aplikasi ini dibuat. *Activity diagram* memilih menu dinyatakan pada gambar berikut ini

Pengguna	Sistem
----------	--------



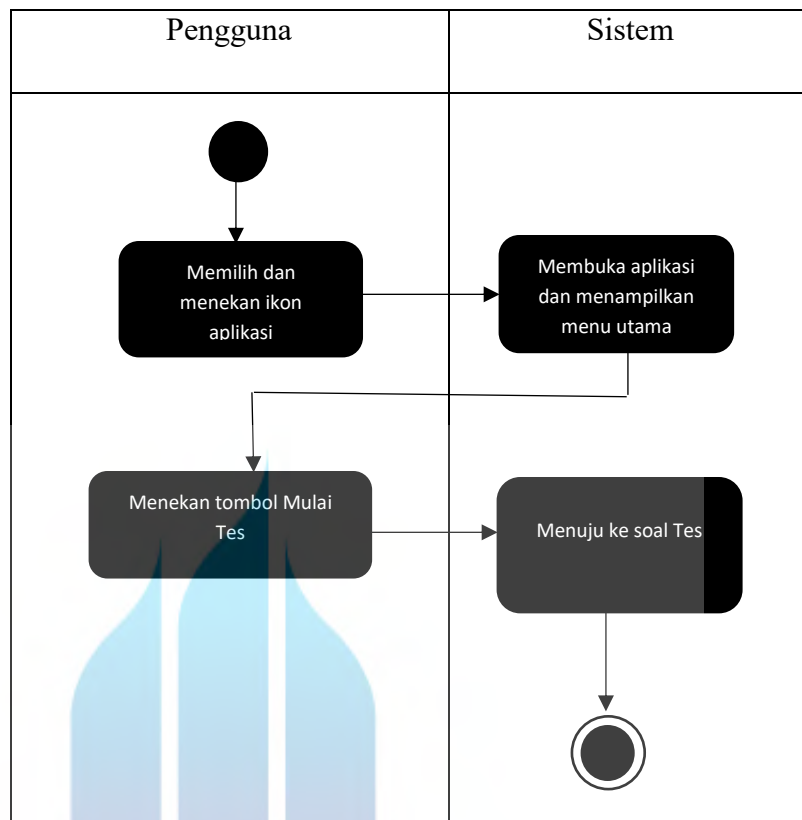
Gambar 4.5. Activity Diagram menu

Pada gambar menunjukkan aktifitas ketika pengguna menggunakan aplikasi dan masuk ke menu utama. Setelah pengguna masuk ke dalam menu utama selanjutnya adalah melakukan pemilihan salah satu menu yang diinginkan pengguna kemudian sistem akan menampilkan konten pada masing - masing menu tersebut.

UNIVERSITAS MERCU BUANA

4.1.4.2. Activity Diagram Menu Mulai Tes

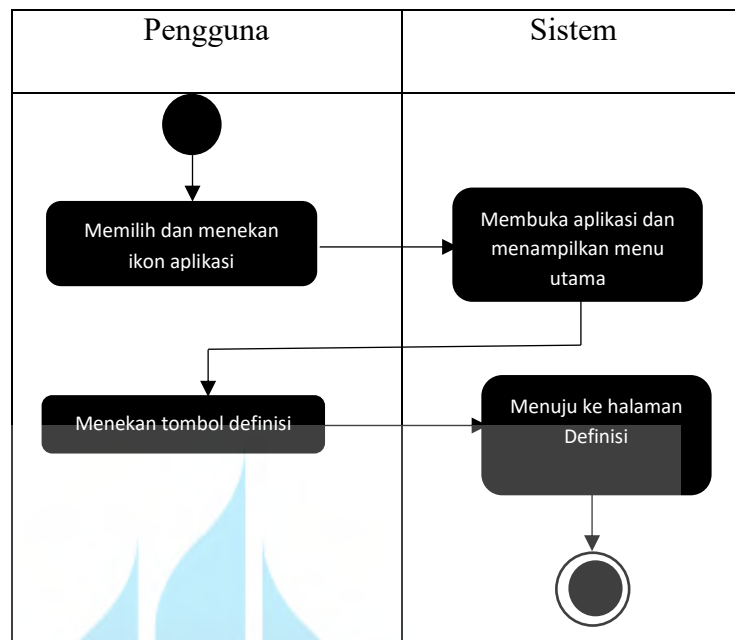
Activity diagram memilih menu Mulai menggambarkan aktivitas ketika pengguna memilih menu mulai. Pengguna memilih menu mulai selanjutnya sistem akan menampilkan dan mengarahkan pengguna ke pemindai kamera. *Activity diagram* memilih menu mulai dinyatakan pada gambar.



Gambar 4.6. Activity Diagram menu Mulai Tes

4.1.4.3. Activity Diagram Menu Definisi

Activity diagram memilih menu definisi menggambarkan aktivitas ketika pengguna memilih menu definisi. Pengguna memilih menu definisi selanjutnya sistem akan menampilkan ke halaman yang memberikan informasi tentang aplikasi dan informasi buta warna. *Activity diagram* memilih menu definisi dinyatakan pada gambar.

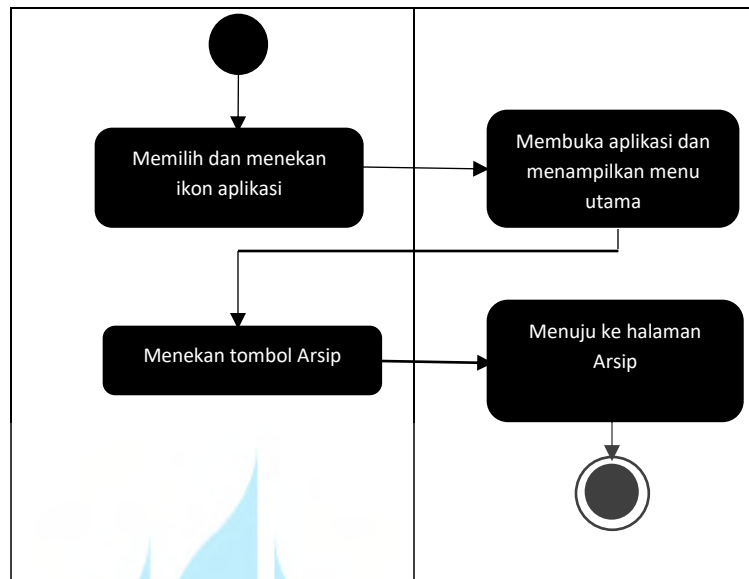


Gambar 4.7. Activity Diagram menu definisi

4.1.4.4. *ACTIVITY DIAGRAM MENU ARSIP*

Activity diagram memilih menu arsip menggambarkan aktivitas ketika pengguna memilih menu arsip. Pengguna memilih menu arsip selanjutnya sistem akan menampilkan ke halaman yang memberikan informasi hasil tes dari menu pengujian. *Activity diagram* memilih menu arsip dinyatakan pada gambar.

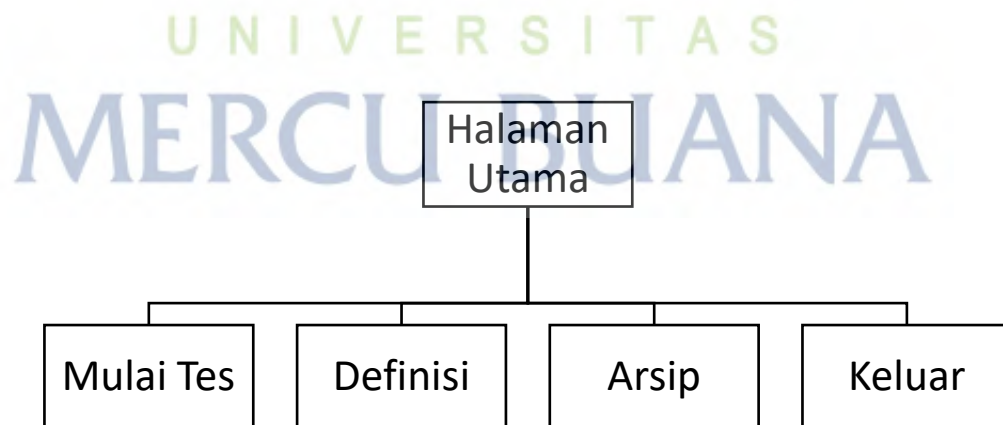




Gambar 4.8. Activity diagram menu *Arsip*

4.2. METAFORA DESAIN STRUKTUR MENU UTAMA

Struktur menu merupakan gambaran umum dari suatu rancangan aplikasi untuk memudahkan pengguna dalam menjalankan aplikasi. Dalam pembuatan struktur menu ini diharapkan dapat mempermudah pengguna dalam menggunakan aplikasi. Berikut ini perancangan struktur menu pada *aplikasi kebutaan warna* berbasis *android* sebagai berikut:



Gambar 4.9. Struktur menu aplikasi

Pada gambar menunjukkan menu utama aplikasi sebagai penunjang kemudahan dalam pemakaian pengguna. Menu mulai sebagai akses masuk ke dalam menu selanjutnya sebelum memasuki menu pengujian. Menu definisi memberikan informasi tentang buta warna. Menu arsip merupakan menu untuk melihat kembali *history* hasil tes di menu pengujian. Menu keluar membantu pengguna apabila ingin keluar dari aplikasi.

4.3. METAFORA DESAIN ANTAR MUKA

Perancangan antarmuka dibuat untuk memberikan gambaran terhadap keadaan yang sebenarnya dari aplikasi yang akan dibangun. Berikut ini adalah desain antar muka yang akan diterapkan pada aplikasi yang akan dibangun.

4.3.1. Rancangan Tampilan Menu Utama

Rancangan antarmuka utama yang akan diterapkan pada aplikasi yang akan diterapkan pada aplikasi yang akan dibangun dengan resolusi tampilan 800 x 600 *pixels*. Tampilan utama didukung dengan *background* yang menarik, berwarna dan desain diambil dari berbagai logo pembangun aplikasi *buta warna*. Desain antar muka menu utama dinyatakan:



Gambar 4.10. Tampilan Menu Utama

Berdasarkan gambar terdapat beberapa tombol sebagai pendukung kebutuhan pengguna. Berikut fungsi beberapa tombol pada menu utama dinyatakan pada table

Tabel 4.1. Tombol pada menu

No	Tombol	Fungsi
1	Mulai Tes	Menuju ke halaman pengujian soal-soal
2	Definisi	Menampilkan informasi tentang buta warna
3	Arsip	Menampilkan informasi hasil tes buta warna
4	Keluar	Keluar dari aplikasi

4.3.2. Rancangan tampilan menu Definisi

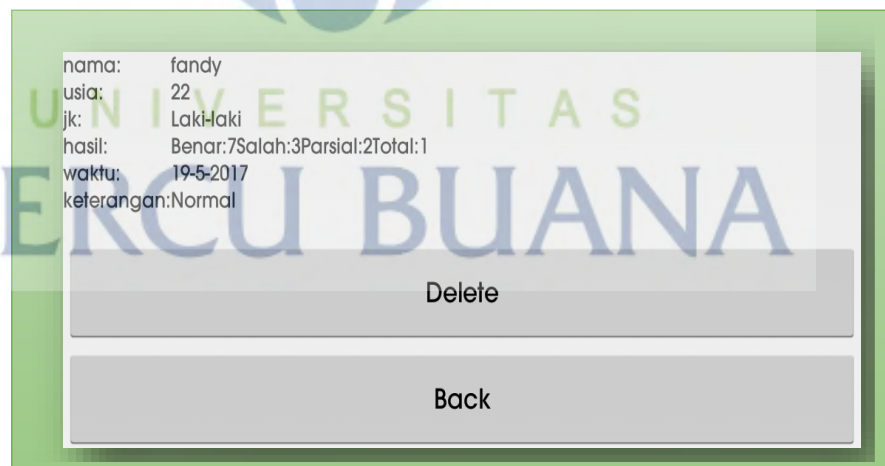
Perancangan halaman menu definisi yang berisi tentang fungsi dan kegunaan aplikasi yang telah dibuat, sehingga memberikan informasi tentang buta warna kepada pengguna. Perancangan halaman menu desfinisi :



Gambar 4.11. Tampilan Menu Tentang

4.3.3. Rancangan Tampilan Menu Arsip

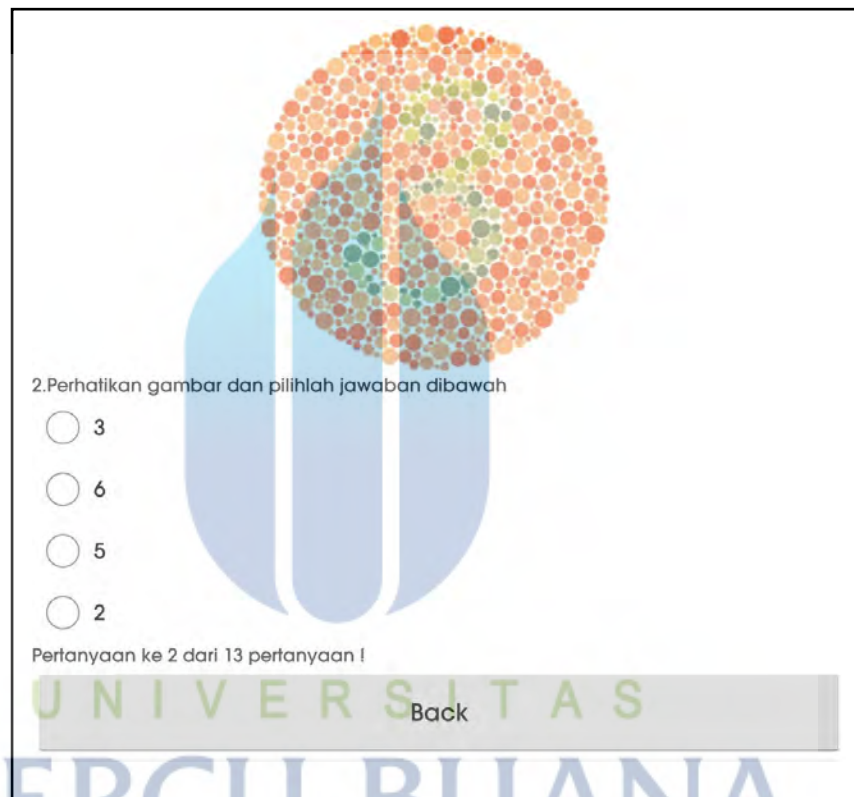
Perancangan halaman menu arsip yang berisi hasil tes buta warna, di dalam menu ini di maksudkan agar pengguna dapat melihat hasil tes yang sudah pernah di lakukan sebelumnya ;



Gambar 4.12. Halaman Arsip

4.3.4. Rancangan Tampilan Halaman Soal

Setelah pengguna menekan tombol mulai tes, selanjutnya akan muncul tampilan untuk mengisikan nama dan umur akan. Setelah itu pengguna dapat melakukan tes buta warna di halaman soal-soal tes buta warna. Berikut ini adalah rancangan tampilannya.



Gambar 4.13. Tampilan Halaman Soal tes

4.1.5. Rancangan Database

Database pada Aplikasi Tes Buta Warna ini menggunakan database SQLite. Database berfungsi untuk menyimpan data yang berkaitan dengan proses pada aplikasi tersebut. Rancangan database db_butawarna menggunakan 8 buah tabel yaitu tabel tb_User,

tb_jenis_tes, tb_tes, tb_det_tes, tb_plate, tb_grup, tb_det_grup, dan tb_pasien, berikut adalah penjelasan dari setiap tabel tersebut.

a) Tabel User

Tabel *User* dengan nama *tb_User* merupakan tabel yang berfungsi untuk menyimpan nama *User* beserta password *User* berdasarkan *id_User*, yang nantinya digunakan saat proses login. Susunan field pada Tabel *tb_User* akan dijelaskan pada Tabel 3.1

Tabel 4.2. Susunan Tabel *tb_User*

No	Nama Field	Tipe Data	Keterangan
1	<i>id_User</i>	Integer	Menyimpan id user id ini berupa umur <i>user</i>
2	<i>nama_User</i>	Varchar	Menyimpan <i>Username</i> yang akan digunakan untuk proses login dari setiap <i>User</i> .

Tabel *tb_User* memiliki 2 buah field yaitu *id_User*, *nama_User*, *nama_user*. Setiap *id_User* memiliki nama *User* yang disimpan pada kolom *nama_User*,

b) Tabel Tes

Tabel tes dengan nama *tb_tes* merupakan tabel yang berfungsi untuk menyimpan hasil tes dari setiap tes yang dilakukan oleh *User*. Hasil tes yang disimpan meliputi hasil tes

mata normal, hasil tes buta warna parsial dan hasil tes buta warna total.

Tabel 4.3. Susunan Tabel tb_tes

No.	Nama Field	Tipe Data	Keterangan
1	id_tes	Integer	Menyimpan id dari setiap tes.
2	id_User	Integer	Menyimpan id <i>User</i> yang menandakan bahwa id tes tersebut dimiliki oleh <i>User</i> dengan id ini.
3	tgl_tes	Date	Menyimpan tanggal dilakukannya tes terhadap id_tes tersebut.
4	hasil_normal	Text	Menyimpan nilai presentase mata normal berdasarkan id_tes tersebut.
5	hasil_parsial	Text	Menyimpan nilai presentase buta warna parsial berdasarkan id_tes tersebut.
8	hasil_total	Text	Menyimpan nilai presentase buta

			warna total berdasarkan id_tes tersebut.
--	--	--	--

Tabel *tb_tes* memiliki 8 buah field yaitu *id_tes*, *id_User*, *tgl_tes*, Setiap *id_tes* memiliki 4 hasil tes yaitu presentase mata normal yang disimpan pada kolom *hasil_normal*, presentase buta warna parsial, dan presentase buta warna total yang disimpan pada kolom *hasil_total*.

c) *Tabel Detail Tes*

Tabel detail tes dengan nama *tb_det_tes* merupakan tabel yang berfungsi untuk menyimpan nilai detail tes setiap *User* saat melakukan tes buta warna, berikut adalah susunan field pada Tabel *tb_det_tes*.

Tabel 4.4. Susunan Tabel *tb_det_tes*

No.	Nama Field	Tipe Data	Keterangan
1	<i>id_det_tes</i>	Integer	Menyimpan id dari masingmasing detail tes.
2	<i>id_tes</i>	Integer	Menyimpan id_tes yang menandakan bahwa detail tes ini dimiliki oleh id_tes tersebut.

3	jawaban_User	Text	Menyimpan nilai jawaban User yang di-input saat proses tes buta warna
4	normal	Text	Menyimpan nilai untuk jawaban mata normal.
5	parsial	Text	Menyimpan nilai untuk jawaban buta warna parsial
6	Total	Text	Menyimpan nilai untuk buta warna total.

Kolom jawaban_User berfungsi untuk menampung nilai dari jawaban yang di input oleh User, pada kolom normal akan diisi nilai 1 jika nilai antara kolom jawaban_User sesuai dengan nilai pada kolom nilai_plate_normal(tb_plate). sedangkan jika nilai antara kolom jawaban tidak sesuai dengan nilai pada kolom nilai_plate_normal (tb_plate) akan ditulis 0 pada kolom normal, untuk kolom parsial dan kolom total akan diisi dengan nilai 1.

d) Tabel Plate

Tabel plate dengan nama tb_plate merupakan tabel yang berfungsi untuk menyimpan gambar plate dengan menggunakan id_plate, nilai dari setiap plate seperti nilai plate untuk mata normal, nilai plate untuk buta warna parsial dan nilai plate untuk buta warna total.

Tabel 4.5. Susunan Tabel tb_plate

No.	Nama Field	Tipe Data	Keterangan
1	id_plate	Integer	Menyimpan id dari masing-masing plate.
2	nilai_plate_normal	Text	Menyimpan nilai jawaban plate untuk mata normal dari setiap id_plate.
3	nilai_plate_parsial	Text	Menyimpan nilai jawaban plate untuk buta warna parsial
6	nilai_plate_total	Text	Menyimpan nilai jawaban plate untuk mata parsial dari setiap id_plate.

Tabel tb_plate berfungsi untuk menyimpan jawaban setiap plate, setiap plate memiliki nilai atau jawaban yang berbeda dari setiap penderita, misalnya pada id_plate 2, mata normal akan melihat angka pada kolom nilai_plate_normal yaitu angka 8, sedangkan buta warna parsial akan melihat angka pada kolom nilai_plate_merahhijau yaitu angka 3, penderita buta warna protanopia akan melihat angka pada kolom nilai_plate_parsial yaitu angka 0, dan penderita buta warna total akan melihat angka pada kolom nilai_plate_parsial.

e) **Tabel Grup**

Tabel grup dengan nama `tb_grup` merupakan tabel yang berfungsi untuk menyimpan identitas dari setiap grup. Setiap grup memiliki beberapa plate yang akan ditampilkan saat tes buta warna. Berikut adalah susunan field pada Tabel `tb_grup`.

Tabel 4.6. Susunan Tabel `tb_grup`

No.	Nama Field	Tipe Data	Keterangan
1	<code>id_grup</code>	Integer	Menyimpan id dari masingmasing grup.
2	<code>no_grup</code>	Text	Menyimpan <code>no_grup</code> dari setiap <code>id_grup</code>

Terdapat 2 field yang terdapat pada tabel `tb_grup` yaitu field `id_grup` dan field `no_grup`. Setiap `id_grup` mewakili 1 grup, misalnya `id_grup` 1 memiliki `no_grup` 1.

f) **Tabel Detail Grup**

Tabel detail grup dengan nama `tb_det_grup` merupakan tabel yang berfungsi untuk menyimpan isi detail grup dari setiap grup dan menyimpan plate yang digunakan pada grup tersebut. Berikut adalah susunan field pada Tabel `tb_det_grup`.

Tabel 3.1. Susunan Tabel `tb_det_grup`

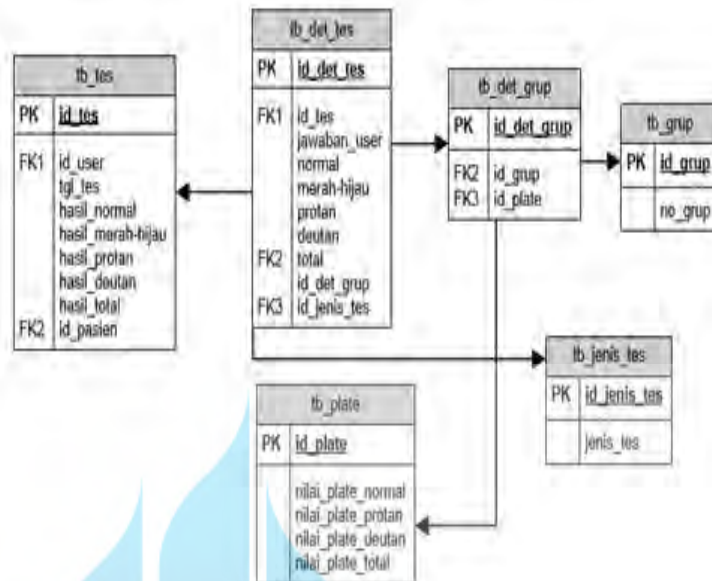
No.	Nama Field	Tipe Data	Keterangan
1	<code>id_det_grup</code>	Integer	Menyimpan id dari masingmasing detail grup.

2	id_grup	Integer	Menyimpan id grup dari masing-masing grup.
3	id_pate	Integer	Menyimpan id plate dari setiap plate yang akan ditampilkan.

Terdapat 3 field yang terdapat pada Tabel tb_det_grup yaitu field id_det_grup, id_grup dan id_plate. Tabel tb_det_grup akan menyimpan id_plate dari setiap id_grup, misalnya id_grup 1 menyimpan id_plate 1, 3, 5, 6, dan 2. Saat proses tes buta warna berlangsung aplikasi akan melakukan random terhadap id_grup, setelah id_grup didapat maka setiap id_plate yang memiliki id_grup tersebut akan di random. Tabel tb_det_grup memiliki relasi foreign key terhadap Tabel tb_grup dan Tabel tb_plate. Tabel tb_tes memiliki relasi foreign key terhadap Tabel tb_pasien sehingga 1 id_User bisa memiliki banyak id_tes. Tabel tb_det_tes memiliki hubungan relasi foreign key terhadap Tabel tb_det_grup dan tb_tes.

g) *Relasi Database*

Aplikasi ini memiliki 7 buah tabel yaitu Tabel tb_User, tb_tes, tb_det_tes, tb_grup, tb_det_grup, dan tb_plate.



Gambar 4.14. Relasi Database Aplikasi Tes Buta Warna