BAB 3
ANALISIS DAN PERANCANGAN SISTEM

Analisis dan perancangan sistem memerlukan tahapan yang sistematis untuk mendapatkan aplikasi yang baik dan sesuai dengan kegunaan dan tujuannya. Tahapan awal dari analisis adalah menganalisis sistem yang meliputi analisis masalah, prosedur yang sedang berjalan, aplikasi yang akan dibangun, metode yang digunakan, alat bantu yang sudah ada, menganalisis kebutuhan-kebutuhan pengguna, kebutuhan non-fungsional, kebutuhan fungsional, kebutuhan perangkat keras, dan kebutuhan perangkat lunak. Sedangkan untuk tahapan perancangan sistem membahas tentang perancangan sistem dan perancangan antar muka yang akan dibangun.

3.1 Analisis Sistem

Analisis Sistem adalah suatu cara penguraian sistem yang akan dibangun. Tahapan punguraian yang dilakukan dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan tujuan untuk megidentifikasikan dan mengevaluasikan permasalahan pada sistem yang lama sehingga nantinya dapat dilakukannya perbaikan. Analisis dapat juga diartikan sebagai penelitian atas sistem yang telah ada dengan tujuan untuk merancang sistem yang baru. Dalam membangun sebuah sistem sangat perlu dilakukan tahapan tentang analisis sistem yang akan dibangun.

3.1.1 Analisis Masalah

Game ber-*genre* ritme merupakan *game* yang menyenangkan karena untuk memainkannya pengguna harus mengikuti gerakan yang muncul sesuai dengan irama musik yang mengiringi permainan. *Game* ber-*genre* ritme yang dibangun adalah *game* dance touch. Cara memainkan *game* dance touch adalah pengguna akan mengikuti gerakan jari tangan yang tampil pada layar sesuai dengan musik yang dimainkan. Gerakan – gerakan yang muncul mensimulasikan bentuk jari tangan penguna. Jika gerakan – gerakan tersebut dapat diikuti dengan baik maka akan dapat mengendalikan isi dari *game* dance touch. Untuk dapat memainkan

game sejenis *game* dance touch pengguna biasanya mengunakan sebuah alat *controller* berbentuk fisik seperti *joystik* atau *gamepad, keyboard*, dan *mouse.* Namun terdapat kekurangan saat menggunakan *controller* fisik yaitu pengguna cepat merasakan bosan karena dengan *controller* fisik tidak mensimulaikan gerakan yang tampil pada layar sehingga membuat *game* sejenis *game* dance touch menjadi tidak menyenangkan untuk dimainkan. *Controller* fisik juga membuat pemain menjadi tidak aktif karena pengguna hanya menikmati *game* dengan duduk terdiam sedangkan *game* sejenis *game* dance touch lebih menyenangkan dimainkan dengan menggerakan tangan atau jari tangan penggguna. Dengan kekuragan yang telah disebutkan di atas bahwa interaksi dengan *controller* fisik yang digunakan untuk bermain *game* sejenis *game* dance touch hanya membuat pengguna menjadi cepat bosan dan tidak aktif serta terkesan monoton.

Game dance touch dibangun dengan memanfatkan teknologi *gesture recognition*. Teknologi *gesture recognition* memungkinkan pengguna untuk berinteraksi langsung dengan permainan tanpa menggunakan *controller* fisik dengan cara bergerak dan juga dapat memberikan pengalaman bermain *game* lebih menarik karena melibatkan gerakan organ tubuh pengguna saat memainkanya contohnya gerakan tangan atau jari tangan penguna. Terlebih saat memainkan *game* sejenis *game* dance touch.

3.1.2 Analisis *Game* Sejenis

Tahap ini akan dilakukan analisis terhadap *game* sejenis yang telah dibuat. Tujuan dari analisis ini yaitu untuk membandingkan konsep dari *game* yang sudah ada untuk kemudian dikembangkan sesuai dengan kebutuhan dari *game* yang akan dibangun. Ada 2 *game* yang akan dianalisis diantaranya *game* Just Dance dan OSU!.

3.1.2.1 *Game* Just Dance

Game Just Dance merupakan *game* ritme yang mengajak pengguna untuk ikut menari dengan memanfaatkan sebuah *device* sebagai *controller*. Just dance pertama kali dikembangkan oleh *developers* Ubisoft Paris dan Ubisoft Milan.

Game ini dirilis pertama kali pada tahun 2009 khusus pada *console game* Wii, setelah sukses pada *console* Wii Ubisoft menggembangkan kembali *game* ini ke beberapa *console game* seperti Xbox 360, Playstation 3, Playstation 4 dan terakhir muncul untuk pengguna PC.

Untuk mengontrol just dance pada PC menggunakan *smartphone* sebagai *remote control* untuk mendeteksi gerakan tubuh pengguna. Dengan cara gerakan pengguna di rekam secara visual melalui *sensor* yang terdapat pada *smartphone* , rekaman gerakan tersebut di kirim ke dalam *game* menggunakan koneksi jaringan wifi yang sama dengan PC. Berikut ini merupakan gambar logo *game* just dance dapat dilhat pada gambar 3.1.



Gambar 3.1 *Game* Just Dance

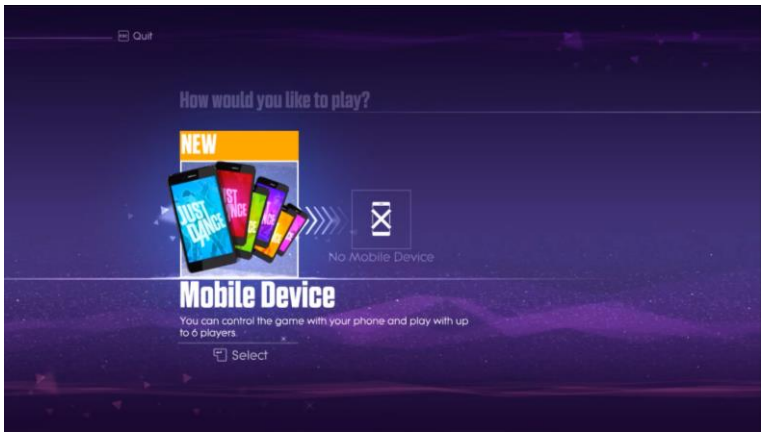
3.1.2.1.1 Cara Menggunakaan *Game* Just Dance

Pada *game* just dance ada beberapa menu yang disediakan yaitu : menu untuk mengkoneksikan *smartphone* dengan pc, menu utama, menu memilih musik, menu profil pengguna dan menu *options*.

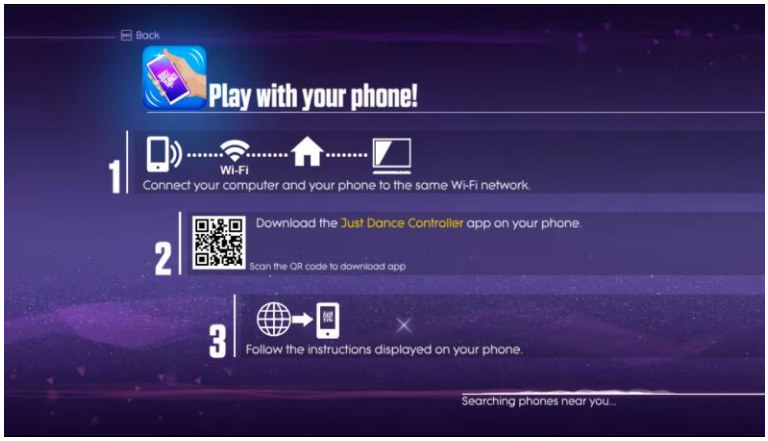
1. Menu Mengkoneksikan *smartphone* dengan PC

   Pada menu ini akan tersedia 2 pilihan menu yaitu: *mobile device* dan *not mobile device*. Jika memilih submenu *mobile device* maka akan muncul tampilan proses koneksi dengan *smartphone*. Jika memilih submenu *not mobile device* maka pengguna masuk kedalam *game* hanya untuk menonton *gameplay* dari pemain lain. Berikut ini merupakan gambar tampilan antar

muka untuk mengkoneksikan *smartphone* dan gambar proses koneksi pc dengan komputer dapat dilihat pada gambar 3.2 dan 3.3.



Gambar 3.2 Menu Koneksi Pc Dengan *Smartphone*



Gambar 3.3 Tampilan Proses Koneksi Pc Dengan *Smartphone*

2. Menu utama

   Pada menu utama terdapat beberapa submenu yaitu : *just dance* merupakan menu untuk memilih *rivals* yang berarti pemain bermain sendiri atau *coop* yang berarti bermain bersama pengguna lain dalam 1 tim setelah memilih maka masuk kedalam menu memilih musik, *just dance machine* merupakan menu untuk memilih musik yang dilakukan secara acak, *world dance floor* merupakan menu yang menyediakan *gameplay* yang unik yaitu pengguna akan disiapkan untuk melawan bos dengan musik yang pilih secara acak, *just dance tv* merupakan menu yang berisi video tentang just dance, *dance quest* merupakan menu yang berisi misi-misi khusus yang harus dimainkan oleh pengguna dengan musik yang pilih secara acak, *sweet and playlist*

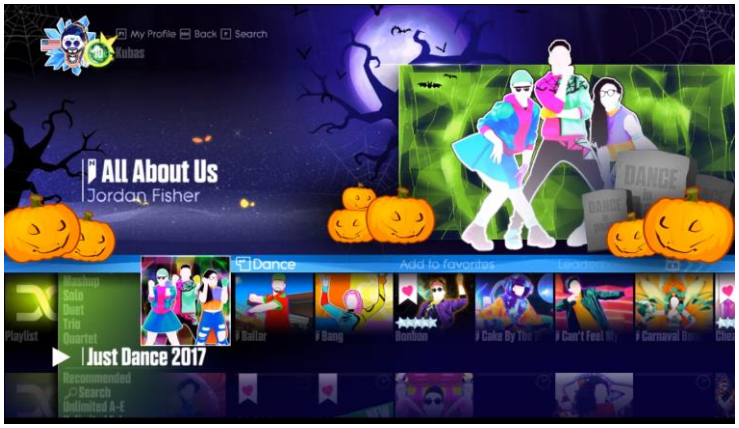
merupakan menu yang menyediakan submenu untuk memilih *playlist, all song, new platlist* dan *non-stop shuffle* menu-menu tersebut disediakan agar pengguna dapat bermain sambil berolahraga dengan musik yang disukai sesuai dengan pilihan musik yang sesuai dengan selera pengguna, *video challenge* merupakan menu yang menyediakan submenu *create a challenge* yaitu pengguna akan membuat sebuah *video gameplay* yang dimainkan oleh pengguna dan diupload ke *server* just dance dan *challangers* merupakan submenu yang disediakan untuk menantang lawan yang telah membagikan *video challage*-nya ke *server* just dance, *ubisoft club* merupakan menu yang berisi kumpulan *video gameplay* yang dimainkan oleh semua pengguna, dan *just dance unlimated* merupakan menu yang menyedikan 200 lebih musik eksklusif yang dapat dimainkan pengguna. Berikut adalah gambar tampilan menu utama dari *game* just dance dapat dilihat pada gambar 3.4.



Gambar 3.4 Menu Utama Just Dance

3. Menu memilih musik

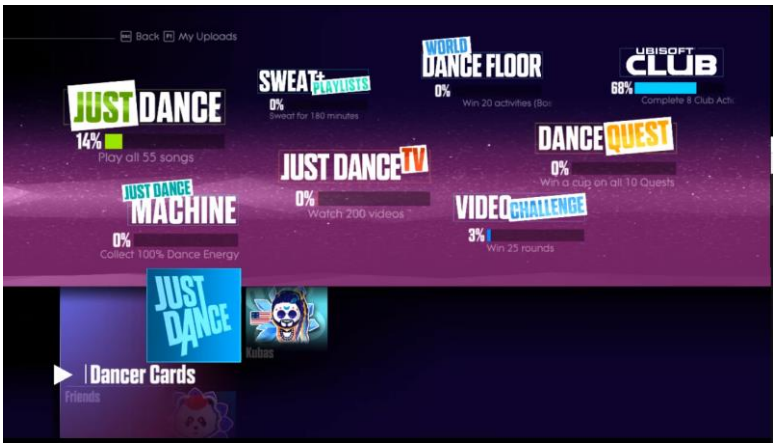
   Pada menu ini menyedikan tampilan untuk memilih musik. Ada beberapa *playlist* musik yang dapat dipilih yaitu *just dance 2017* berisi musik yang populer 2017, *recommended* berisi musik yang direkomendasikan, *search* yaitu pengguna dapat mencari musik yang ingin dimainkan dan *unlimited a-z* berisi musik yang diurutkan berdasarkan nama judul musik. Terdapat juga beberapa kategori musik yang dapat dipilih yaitu : *mashup, solo, duet, trio* dan *quartet*. Berikut ini adalah tampilan antarmuka dari menu memilih musik dapat dilihat pada gambar 3.5.



Gambar 3.5 Menu Memilih Musik

4. Menu profil pengguna

   Pada menu ini pengguna dapat melihat statistik perkembangan permainan pengguna dari berbagai submenu yang terdapat pada menu utama dan pengguna dapat mengganti *username* maupun *profile picture*. Berikut ini adalah gambar tampilan antarmuka menu profil pengguna dapat dilihat pada gambar 3.6.

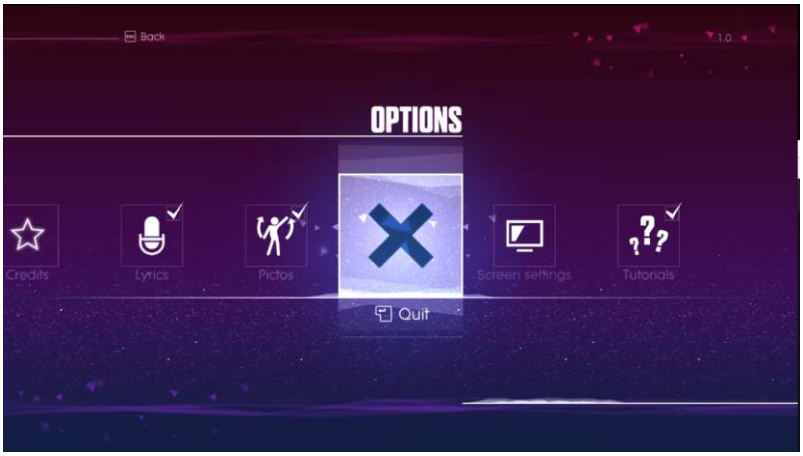


Gambar 3.6 Menu Profil Pengguna

5. Menu *options*

   Pada menu options terdapat beberapa pengaturan yang dapat diaktifkan atau dinonaktifkan yaitu autodance yaitu pengaturan untuk menampilkan atau tidak seorang peraga pada *gameplay*, *creadits* yaitu isi tentang just dance, *lyrics* yaitu pengaturan untuk menampilkan lirik atau tidak, *pictos* yaitu pengaturan untuk menampilkan gambar peraga kecil yang muncul di bagian bawa kanan dengan tujuan menuntun pengguna bergerak kearah yang benar,

quit untuk keluar dari *game*, *screen setting* yaitu pengaturan untuk mengubah tampilan antara *full screen* atau *windowed mode*, *tutorial* yaitu pengaturan untuk menampilkan atau tidak tutorial sebelum memulai *gameplay*. Berikut ini adalah gambar tampilan antarmuka dari menu *options* dapat dilihat pada gambar 3.7.



Gambar 3.7 Menu *Options*

3.1.2.1.2 Komponen *Game* Just Dance

Pada *game* just dance terdapat beberapa komponen antara lain menu, grafis, tombol, dan fungsi apikasi dapat dilihat pada tabel 3.1. terdapat komponen *game* just dance.

Tabel 3.1 Komponen *Game* Just Dance

No.	Komponen	Keterangan
1	Menu	Pada *game* just dance terdapaat beberapa menu dan sub menu sebagai berikut : 1. Menu memgkoneksikan *smartphone* dengan pc. Sub menu *mobile device*, dan sub menu *not mobile device* 2. Menu utama. Sub menu *just dance*, sub menu *just dance machine*, sub menu *world dance floor*, sub menu *just dance tv*, sub menu *dance quest*, sub menu *sweet and playlist*, sub menu *video challenge*, sub menu *ubisoft club*, dan sub menu *just dance unlimited*. 3. Menu memilih musik. 4. Menu *profile* pengguna. Sub menu *dancer card*, dan sub menu *profile* 5. Menu *options*. Sub menu *autodance*, sub menu *credit*, sub menu *lyrics*, sub menu *pictos*, sub menu *quit*, sub menu *screen setting*, dan sub menu *tutorials*
2	Grafis	*Game* ini menggunkan grafis 2 dimensi
3	Tombol	*Game* just dance memiliki tombol yang digunakan untuk mengarahkan

		penguna di dalam *game*. berikut tombol yang digunakan beserta fungsinya antara lain : 1. Tombol *mobile device*, digunakan untuk mengkoneksikan pc dengan *smartphone* sebagai *controller* 2. Tombol *not mobile device* , digunakan untuk masuk ke menu utama tanpa terkoneksi ke *smartphone* 3. Tombol *just dance*, digunakan untuk memilih mode bermain *rival* atau *coop* 4. Tombol *rival*, digunakan untuk masuk ke dalam menu memilih musik sebagai *single player* *5.* Tombol *coop*, digunakan untuk masuk ke dalam menu memlih musik sebagai *multiplayer* 6. Tombol *just dance machine*, digunakan untuk mengacak mode bermain dan memlih lagi secara acak sesuai mode yang terpilih 7. Tombol *icon* musik, digunkan untuk memilih musik yang ingin dimainkan 8. Tombol *world dance floor*, digunakan untuk memilih mode permainan bermain bersama dengan komunitas melawan boss 9. Tombol *just dance tv*, digunakan untuk masuk menu just dance tv *10.* Tombol *icon video*, digunakan untuk untuk memutar video pada *just dance tv* 11. Tombol *dance quest*, digunakan untuk masuk menu *dance quest* 12. Tombol *icon quest*, digunakan untuk memilih quest yang akan dimainkan 13. Tombol *dance skills*, digunakan untuk memilih level *beginer*, *just dance*, *dance master* dan *super start* pada menu *dance quest* 14. Tombol memilih karakter, digunakan untuk memilih karakter yang dapat digunakan oleh pengguna. 15. Tombol *sweet and playlist*, digunakan untuk masuk menu sweet and playlist dan untuk memilih tombol *playlist*, *all song*, *new playlist* dan *non-stop shuffle* 16. Tombol *playlist*, digunakan untuk masuk ke menu daftar playlist yang disediakan oleh *game* 17. Tombol *all song*, digunakan untuk masuk menu yang nemapilkan semua musik yang terdapat di dalam *game* 18. Tombol *new playlist*, digunakan untuk membuat playlist baru yang berisi musik yang sesuai dengan selera pengguna 19. Tombol *non-stop shuffle*, digunakan untuk memutar musik secara acak dan akan terus diputar secara otomatis selama waktu yang diinginkan oleh pengguna *20.* Tombol *video challenge*, digunakan untuk untuk memilih tombol *create a challenge* dan *challengers* 21. Tombol *create a challenge*, digunakan untuk masuk menu masuk menu *video challenge* dan memilih musik untuk membuat *video challenge* 22. Tombol *challenge*, digunakan untuk masuk menu masuk menu *video challenge* dan memilih *video challenger* yang telah di upload *23.* Tombol *ubiscoft club*, digunakan untuk masuk menu *ubisoft club* 24. Tombol *just dance unlimited*, digunakan untuk menu *just dance unlimited* 25. Tombol M atau F1, digunakan untuk membuka menu profile pengguna 26. Tombol F1 (menu *profile* pengguna), digunakan untuk melihat *video gameplay* yang di upload oleh penguna 27. Tombol *dancer card*, digunakan untuk mengganti foto *profile*

		pengguna 28. Tombol *friends*, digunakan untuk melihat teman pada *game* just dance 29. Tombol F, digunakan untuk search musik yang ingin dimainkan 30. Tombol F3, digunakan untuk masuk menu options dan memilih tombol *autodance*, *credits*, *lyrics*, *pictos*, *quit*, *screen setting*, dan *tutorials* 31. Tombol *autodance*, digunakan untuk mengaktifkan fitur autodance peraga yang terdapat di dalam *game* 32. Tombol *credits*, digunakan untuk melihat detail dari developers 33. Tombol *lyrics*, digunakan untuk menampilkan lirik lagu pada saat *gameplay* 34. Tombol *pictos*, digunakan untuk menampilkan icon peraga pada sisikanan bawah layar dalam *gameplay* 35. Tombol *quit*, digunakan untuk keluar dari *game* 36. Tombol *screen setting*, digunakan untuk memilih tombol *full screen* dan *windowed mode* 37. Tombol *full screen*, digunakan untuk mengaktifkan mode *full screen* saat bermain *just dance* 38. Tombol *windowed mode*, digunakan untuk mengaktifkan mode *window* saat bermain just dance 39. Tombol *tutorials*, digunakan untuk 40. Tombol *esc* atau *backspace*, digunakan untuk kembali ke menu sebelumnya
4	Fungsi Aplikasi	Aplikasi ini berguna untuk menghibur dan juga mengajak pengguna dapat bermain sambil berolahraga dengan memanfaatkan fitur dari *smartphone* untuk menjadi *remote controller* pada *game* ini membuat pengguna harus mengerakan seluruh badannya untuk bergerak. Aplikasi ini juga dapat meningkatkan daya konsetrasi dari pemain.

3.1.2.1.3 Hasil Analisis *Game* Just Dance

Hasil analisis *game* Just Dance adalah sebagai berikut :

1. Just Dance merupakan *game* ritme yang dapat dimainakn sendiri atau bersama pengguna lain secara *multiplayer online*.
2. Just Dance menyediakan fitur unik. Fitur ini mengharuskan pengguna untuk ikut bergerak saat memainkan *game* ini.
3. Fitur yang disediakan merupakan fitur *remote control* menggunakan *smartphone* untuk mengontrol isi *game* dengan gerakan tubuh dari penggun. Gerakan dari pengguna di rekam melalui *sensor* yang terdapat pada *smartphone* .
4. Dengan memanfaatkan fitur dari *smartphone* membuat *game* ini unik dan menarik untuk dimainkan serta dapat menjadi sarana untuk berolahraga dengan menggerakan seluruh tubuh dari pengguna.

3.1.2.2 *Game* OSU!

Game OSU! dikembangkan oleh Dean Herber yang dirilis pada tanggal 16 September 2007. *Game* ini pertama kali dikembangkan untuk Microsoft Windows dengan bahasa pemograman C# pada .NET Framework.

Osu! adalah sebauh *game* ritme dengan kosep *game* musik yang mengangkat jenis musik anime yang dibangun dengan tampilan simpel dan mudah untuk di mengerti dan dioperasikan oleh pengguna. Berikut ini adalah gambar logo dari *game* OSU! dapat dilihat pada gambar 3.8.



Gambar 3.8 *Game* OSU!

3.1.2.2.1 Cara Menggunakan *Game* OSU!

Pada *game* OSU! terdapat beberapa menu yang disediakan yatu menu utama, menu *play*, menu *options*, dan menu edit.

1. Menu utama OSU!

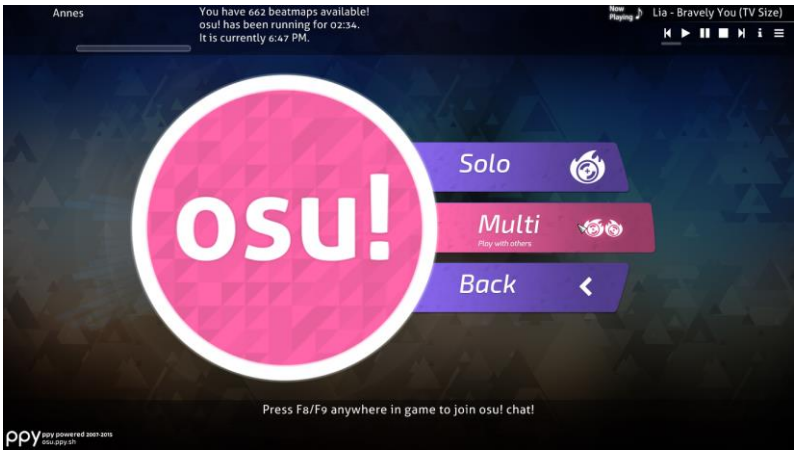
   Pada menu utama terdapat 5 menu yaitu *play* digunakan untuk memilih apakah pemain mau untuk bermain *solo* (satu pemain) atau *multiplayer* (banyak pemain), menu memilih musik digunakan untuk memilih musik yang akan dimainkan, edit digunkan untuk memilih menu edit, *options* untuk memilih menu *options*, dan exit digunakan untuk keluar dari permainan. Berikut ini adalah gambar tampilan menu utama *game* OSU! dapat dilihat pada gambar 3.9.



Gambar 3.9 Menu Utama

2. Menu *play*

   Pada menu *play* osu! terdapat 2 menu di dalamnya yaitu : menu *solo* dan menu *multi*. Kedua menu merupakan pilihan bagi pengguna apakah pengguna akan bermain *solo* (satu pemain) atau pengguna ingin bermain secara bersama banyak pengguna secara *multiplayer*. Setelah memilih salah satu menu maka akan masuk menu memlih musik. Berikut ini adalah gambar tampilan menu *play* dari *game* OSU! dapat dilihat pada gambar 3.10.

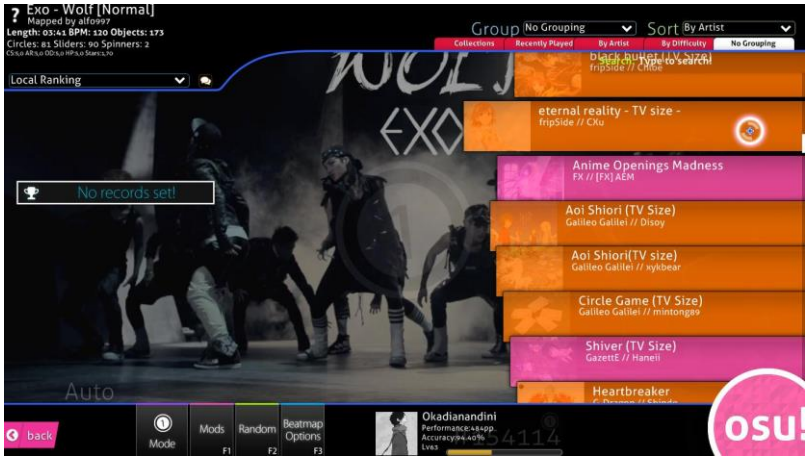


Gambar 3.10 Menu *Play*

3. Menu memilih musik

   Pada menu *play* osu! terdapat beberapa musik dan level yang bisa di pilih. Namun hanya beberapa musik saja yang dapat dimainkan jika pemain merupakan pemain pemula. Pada menu ini juga terdapat beberapa tombol

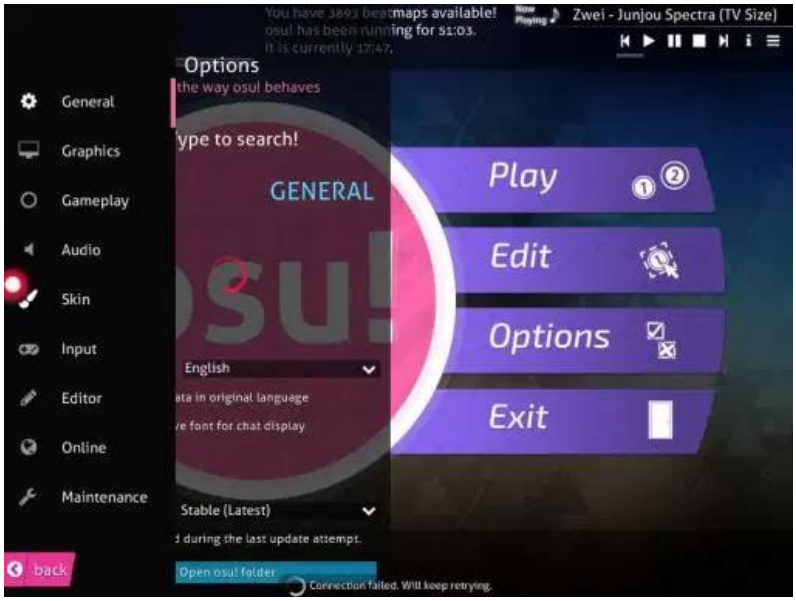
yaitu Mode untuk memilih mode *gameplay* yang terdiri dari 4 mode yaitu osu!, osu!taiko, osu!catch dan osu!mania, *Mods* untuk memiliih *mods* yang dibutuhkan pengguna saat berada di dalam *gameplay*, *Randoms* untuk memilih acak musik yang akan dimainkan, dan *Bitmap Option* untuk *manage collections*, *delete*, *cleare local score*, dan edit. Berikut ini adalah gambar tampilan menu memilih musik dari *game* OSU! dapat dilihat pada gambar 3.11.



Gambar 3.11 Menu Memilih Musik

4. Menu *options*

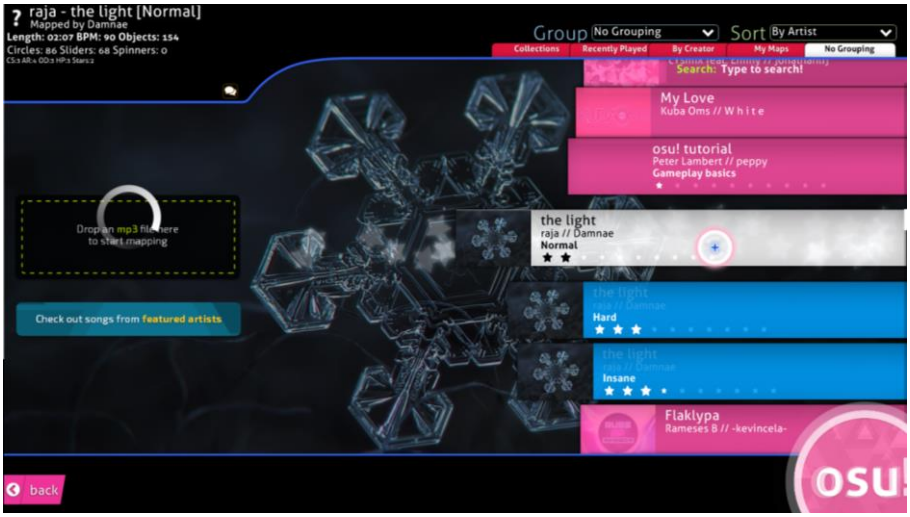
   Pada menu *options* berisi berbagai macam jenis pengaturan yang disediakan antara lain: *general* untuk mengatur *user login* dan pengaturan bahasa, *grahpics* untuk mengatur tampilan dari *game*, *gampelay* untuk mengatur tampilan antarmuka dari *gameplay*, *audio* untuk mengatur suara *game*, *skin* untuk mengatur *background game* dan *background gameplay*, input untuk mengatur *input* apa yang digunakan yaitu menggunakan *mause* atau *keyboard*, *editor* untuk mengatur tampilan dari *editor* pada menu edit, *online* dan *maintenance*. Berikut ini adalah gambar tampilan menu *options* dari *game* OSU! padat dilihat pada gambar 3.12.



Gambar 3.12 Menu *Options*

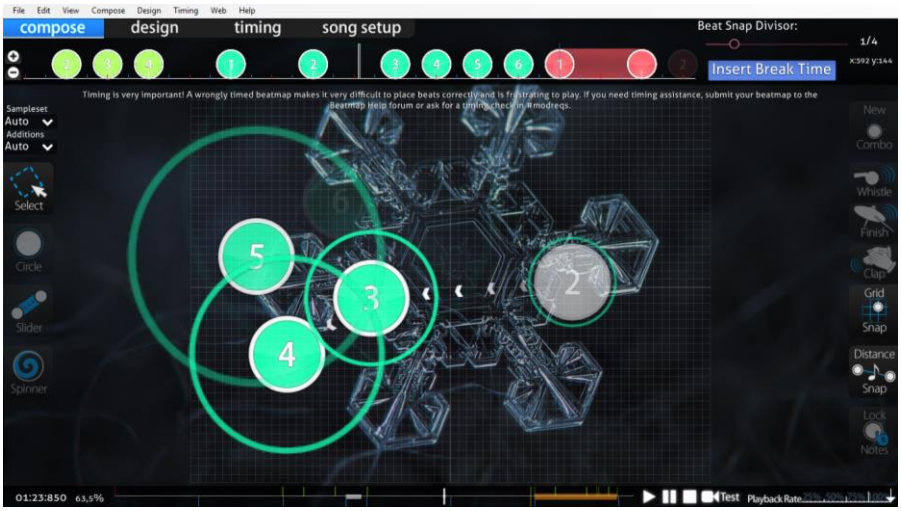
5. Menu edit

Pada menu editor tampilannya menyerupai tampilan menu untuk memilih musik. Seperti pada gambar 3.13 berikut :



Gambar 3.13 Menu Edit

Pada saat pengguna memilih salah satu musik maka akan muncul tampilan editor, seperti pada gambar 3.14 berikut :



Gambar 3.14 Tampilan Editor

Pengguna dapat dengan bebas mengubah tamppilan perintah yang muncul pada saat *gameplay* sesuai dengan musik yang mau di edit oleh pengguna.

3.1.2.2.2 Komponen *game* OSU!

Pada *game* OSU! terdapat beberapa komponen antara lain menu, grafis, tombol, dan fungsi apikasi dapat dilihat pada tabel 3.2. terdapat komponen *game* OSU!.

Tabel 3.2 Komponen *Game* OSU!

No.	Komponen	Keterangan
1	Menu	Pada *game* OSU! terdapat beberapa menu sebagai berikut : 1. Menu utama Sub menu *play*, sub menu *edit*, sub menu *option*, dan sub menu *exit* 2. Menu *play* Sub menu memilih *solo* atau *multi*, sub menu memilih musik, sub menu *start game*, sub menu *mode*, sub menu *mods* dan sub menu *bitmaps option* 3. Menu *edit* Sub menu memilih musik dan sub menu *start editor* 4. Menu *option* Sub menu *general*, sub menu *graphics*, sub menu *gameplay*, sub menu *audio*, sub menu *skin*, sub menu *input*, sub menu *editor*, sub menu *online*, dan sub menu *maintenance* 5. Menu *mode* Sub menu osu!, sub menu osu!taiko, sub menu osu!*catch* dan sub menu osu!mania. 6. Menu *mods* Sub menu memilih mods yang tersedia dan sub menu *reset* semua *mods* 7. Menu *bitmaps options* Sub menu *manage* collections, sub menu *delete*, sub menu *remove from unplayer*, sub menu *clear local score* dan sub menu *edit*

2	Grafis	*Game* ini menggunkan grafis 2 dimensi
3	Tombol	*Game* OSU! memiliki tombol yang digunkan untuk mengarahkan penguna di dalam *game*. berikut tombol yang digunakan beserta fungsinya antara lain : 1. Tombol osu! (pada menu utama), digunakan untuk memilih menu utama 2. Tombol osu! (pada menu memilih musik), digunkan untuk memulai *game* 3. Tombol osu! (pada menu edit), digunkan untuk masuk ke antarmuka *editor* 4. Tombol *play*, digunakan untuk memilih tombol *solo*, *multi*, dan *back* 5. Tombol *solo* atau *multi*, digunakan untuk masuk ke menu memilih musik 6. Tombol *mode*, digunakan untuk memlih *mode game* 7. Tombol *mods*, digunakan untuk memilih *mods* yang digunakan 8. Tombol *reset all mods*, digunakan untuk mereset kembali *mods* yang telah digunakan. 9. Tombol *random*, digunakan untuk mengacak musik yang ingin dimainkan *10.* Tombol *bitsmaps*, digunakan untuk memilih tombol *manage collections*, *delete*, *remove from unplayer*, *clear local score*, *edit, dan close* 11. Tombol *manage collections*, digunakan untuk memenage list musik dalam *game* dan untuk memilih tombol *delete the collections* 12. Tombol *delete* atau *delete the collections*, digunakan untuk menghapus musik 13. Tombol *remove from unplayer*, digunakan untuk menghapus musik yang tidak dimainkan 14. Tombol *clear local score*, digunakan untuk mereset *score* yang telah dicapai 15. Tombol *edit*, digunakan untuk masuk ke menu *edit* 16. Tombol *profile*, digunakan untuk melihat profil pengguna 17. Tombol *back* dan *close*, digunakan untuk kembali ke menu sebelumnya 18. Tombol *exit*, digunakan untuk keluar dari *game*
4	Fungsi Aplikasi	Aplikasi ini sangat menghibur pengguna dengan beberapaa bentuk mode *gameplay* yang berbeda, setiap *gameplay* memberikan antarmuka yang berbeda dengan tingkat kesulitan yang juga berbeda-beda. Dengan beberapa mode *gameplay* yang disediakan dapat meningkatkan konsentrasi dari pengguna.

3.1.2.2.3 Hasil analisis *Game* OSU!

Hasil analisis *game* OSU! adalah sebagai berikut :

1. *Game* OSU! merupkan *game* yang menarik dengan konsep yang unik dimana pemain diberikan beberapa mode yang ingin untuk dimainkan. Terdapat 4 mode yang dapat dimainkan. Ke empat mode tersebut memiliki *gameplay* yang berbeda dengan tingkat kesulitan yang berbeda pula. Setiap mode memberikan tantangan tersendiri untuk para pengguna.

2. *Game* OSU! ini dapat dimainkan bersama orang lain via online *game* yang dapat bermain dengan beberapa orang sekaligus dalam satu room yang dibuat oleh pengguna lain.
3. *Game* OSU! memberikan suatu fungsi yang unik yaitu pengguna diberikan kebebasan unttuk mengedit tampilan *game*, dapat menambahkan musik dari local storage dan dapat mengubah mengedit tampilan *gameplay* dari musik yang diinginkan.

3.1.2.3 Perbandingan Komponen pada *Game* Sejenis

Perbandingan komponen pada *game* sejenis adalah cara membandingkan komponen yang dimiliki oleh masing-masing *game* yang telah dianalisis dengan tujuan dapat membantu dalam pembangunan aplikasi. Tabel 3.3. menjelaskan perbandingan komponen *game* sejenis.

Tabel 3.3 Perbandingan Komponen *Game* Sejenis

No.	Komponen	*Game* Just Dance	*Game* OSU!
1	Menu	Pada *game* just dance terdapaat beberapa menu sebagai berikut : 5 Menu dan 20 Sub menu.	Pada *game* OSU! terdapat beberapa menu sebagai berikut : 7 Menu dan 32 Sub menu.
2	Grafis	*Game* ini menggunkan grafis 2 dimensi.	*Game* ini menggunkan grafis 2 dimensi.
3	Tombol	*Game* just dance memiliki tombol yang digunakan untuk mengarahkan penguna di dalam *game*. berikut tombol yang digunakan beserta fungsinya antara lain : 1. Tombol *mobile device*, digunakan untuk mengkoneksikan pc dengan *smartphone* sebagai *controller*. 2. Tombol *not mobile device*, digunakan untuk masuk ke menu utama tanpa terkoneksi ke *smartphone* . 3. Tombol *just dance*, digunakan untuk memilih mode bermain *rival* atau *coop*. 4. Tombol *rival*, digunakan untuk masuk ke dalam menu memilih musik sebagai *single player* *5.* Tombol *coop*, digunakan untuk masuk ke dalam menu memlih musik sebagai	*Game* OSU! memiliki tombol yang digunkan untuk mengarahkan penguna di dalam *game*. berikut tombol yang digunakan beserta fungsinya antara lain : 1. Tombol osu! (pada menu utama), digunakan untuk memilih menu utama 2. Tombol osu! (pada menu memilih musik), digunkan untuk memulai *game* 3. Tombol osu! (pada menu edit), digunkan untuk masuk ke antarmuka *editor* 4. Tombol *play*, digunakan untuk memilih tombol *solo*, *multi*, dan *back* 5. Tombol *solo* atau *multi*, digunakan untuk masuk ke menu memilih musik 6. Tombol *mode*, digunakan untuk memlih *mode game* 7. Tombol *mods*, digunakan untuk memilih *mods* yang digunakan 8. Tombol *reset all mods*,

		multiplayer 6. Tombol *just dance machine*, digunakan untuk mengacak mode bermain dan memlih lagi secara acak sesuai mode yang terpilih 7. Tombol *icon* musik, digunkan untuk memilih musik yang ingin dimainkan 8. Tombol *world dance floor*, digunakan untuk memilih mode permainan bermain bersama dengan komunitas melawan boss 9. Tombol *just dance tv*, digunakan untuk masuk menu just dance tv *10.* Tombol *icon video*, digunakan untuk untuk memutar video pada *just dance tv* 11. Tombol *dance quest*, digunakan untuk masuk menu *dance quest* 12. Tombol *icon quest*, digunakan untuk memilih quest yang akan dimainkan 13. Tombol *dance skills*, digunakan untuk memilih level *beginer, just dance, dance master* dan *super start* pada menu *dance quest* 14. Tombol memilih karakter, digunakan untuk memilih karakter yang dapat digunakan oleh pengguna. 15. Tombol *sweet and playlist*, digunakan untuk masuk menu sweet and playlist dan untuk memilih tombol *playlist, all song, new playlist* dan *non-stop shuffle* 16. Tombol *playlist*, digunakan untuk masuk ke menu daftar playlist yang disediakan oleh *game* 17. Tombol *all song*, digunakan untuk masuk menu yang nemapilkan semua musik yang terdapat di dalam *game* 18. Tombol *new playlist*, digunakan untuk membuat playlist baru yang berisi musik yang sesuai dengan selera pengguna 19. Tombol *non-stop shuffle*,	digunakan untuk mereset kembali *mods* yang telah digunakan. 9. Tombol *random*, digunakan untuk mengacak musik yang ingin dimainkan *10.* Tombol *bitsmaps*, digunakan untuk memilih tombol *manage collections, delete, remove from unplayer, clear local score, edit, dan close* 11. Tombol *manage collections*, digunakan untuk memenage list musik dalam *game* dan untuk memilih tombol *delete the collections* 12. Tombol *delete* atau *delete the collections*, digunakan untuk menghapus musik 13. Tombol *remove from unplayer*, digunakan untuk menghapus musik yang tidak dimainkan 14. Tombol *clear local score*, digunakan untuk mereset *score* yang telah dicapai 15. Tombol *edit*, digunakan untuk masuk ke menu *edit* 16. Tombol *profile*, digunakan untuk melihat profil pengguna 17. Tombol *back* dan *close*, digunakan untuk kembali ke menu sebelumnya 18. Tombol *exit*, digunakan untuk keluar dari *game*

		digunakan untuk memutar musik secara acak dan akan terus diputar secara otomatis selama waktu yang diinginkan oleh pengguna *20.* Tombol *video challenge*, digunakan untuk untuk memilih tombol *create a challenge* dan *challengers* 21. Tombol *create a challenge*, digunakan untuk masuk menu masuk menu *video challenge* dan memilih musik untuk membuat *video challenge* 22. Tombol *challenge*, digunakan untuk masuk menu masuk menu *video challenge* dan memilih *video challenger* yang telah di upload *23.* Tombol *ubiscoft club*, digunakan untuk masuk menu *ubisoft club* 24. Tombol *just dance unlimited*, digunakan untuk menu *just dance unlimited* 25. Tombol M atau F1, digunakan untuk membuka menu profile pengguna 26. Tombol F1 (menu *profile* pengguna), digunakan untuk melihat *video gameplay* yang di upload oleh penguna 27. Tombol *dancer card*, digunakan untuk mengganti foto *profile* pengguna 28. Tombol *friends*, digunakan untuk melihat teman pada *game* just dance 29. Tombol F, digunakan untuk search musik yang ingin dimainkan 30. Tombol F3, digunakan untuk masuk menu options dan memilih tombol *autodance*, *credits*, *lyrics*, *pictos*, *quit*, *screen setting*, dan *tutorials* 31. Tombol *autodance*, digunakan untuk mengaktifkan fitur autodance peraga yang terdapat di dalam *game* 32. Tombol *credits*, digunakan untuk melihat detail dari developers 33. Tombol *lyrics*, digunakan	

		untuk menampilkan lirik lagu pada saat *gameplay* 34. Tombol *pictos*, digunakan untuk menampilkan icon peraga pada sisikanan bawah layar dalam *gameplay* 35. Tombol *quit*, digunakan untuk keluar dari *game* 36. Tombol *screen setting*, digunakan untuk memilih tombol *full screen* dan *windowed mode* 37. Tombol *full screen*, digunakan untuk mengaktifkan mode *full screen* saat bermain *just dance* 38. Tombol *windowed mode*, digunakan untuk mengaktifkan mode *window* saat bermain just dance 39. Tombol *tutorials*, digunakan untuk menampilkan tutorial, 40. Tombol *esc* atau *backspace*, digunakan untuk kembali ke menu sebelumnya	
4	Fungsi Aplikasi	Aplikasi ini berguna untuk menghibur dan juga mengajak pengguna dapat bermain sambil berolahraga dengan memanfaatkan fitur dari *smartphone* untuk menjadi *remote controller* pada *game* ini membuat pengguna harus mengerakan seluruh badannya untuk bergerak. Aplikasi ini juga dapat meningkatkan daya konsetrasi dari pemain.	Aplikasi ini sangat menghibur pengguna dengan beberapaa bentuk mode *gameplay* yang berbeda, setiap *gameplay* memberikan antarmuka yang berbeda dengan tingkat kesulitan yang juga berbeda-beda. Dengan beberapa mode *gameplay* yang disediakan dapat meningkatkan konsentrasi dari pengguna.

3.1.2.4 Kesimpulan Hasil Analisis *Game* Sejenis

Dari hasil analisis *game* sejenis serta perbedaan komponens pada *game* sejenis dapat disimpulkan terdapat fitur dan mode yang dapat dikembangkan dan digunakan dalam pembangunan aplikasi. Berikut adalah kesimpulan hasil analisis *game* sejenis yang dibuat:

1. *Game* Just Dance merupakan *game* ritme yang menyediakan fitur untuk memanfaatkan *sensor* pada *smartphone* untuk dapat mentracking gerakan tubuh pengguna dan mengubah hasil tracking menjadi *controller* dalam *game* ini. Terdapat pula menu untuk mengintegrasi *smartphone* dengan pc. Dari

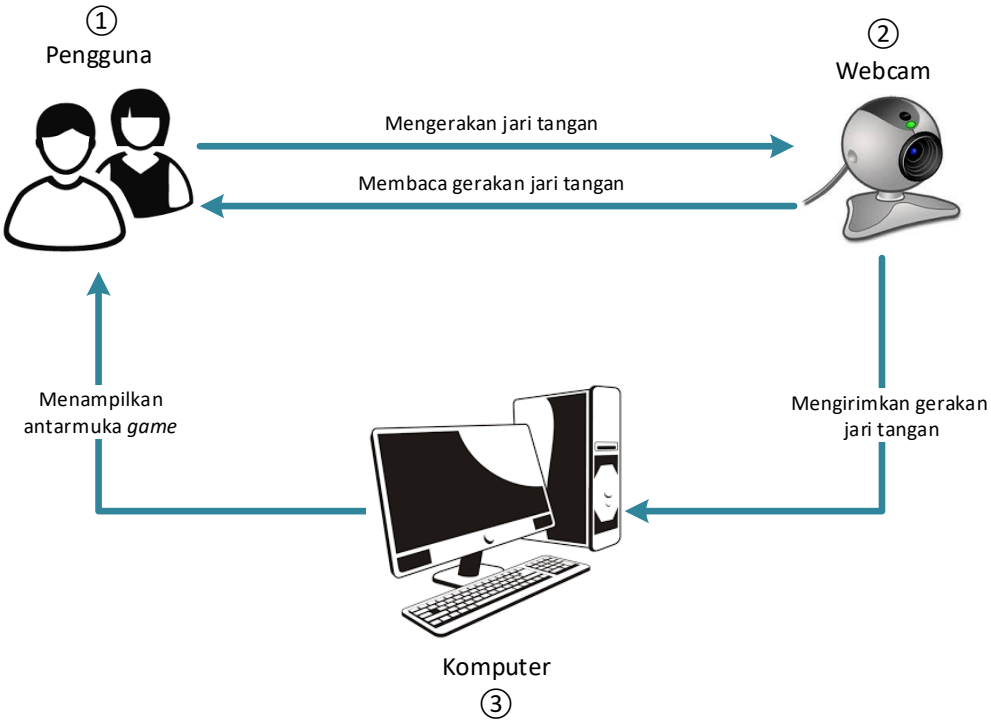
konsep seperti ini dapat memberikan ide cara mengontrol *game* dengan menggunakan *gesture recognition*.

2. *Game* OSU! merupakan *game* ritme yang menyediakan konsep mode *gameplay* yang berbeda-beda sehingga membuat pengguna tertarik mencoba semua jenis *gameplay*. Dari beberapa *gameplay* yang disediakan dapat diambil sebagai referensi dalam pembuatan *gameplay* yang menarik dan unik.

Dari konsep yang disediakan ke dua *game* Just Dance dan OSU! dibangun dengan tujuan untuk menghibur pengguna dengan cara menikmati musik sambil menyelesaikan permainan yang ditampilkan. Kedua *game* mempunyai kelebihan dan kekurangan masing – masing. *Game* Just Dance menyediakan *gameplay* yang menarik dan mempunyai sistem *motion control* namun memiliki *gameplay* yang sama disetiap mode permainannya, dan *game* OSU! mempunyai keunikan dari mode *gameplay* yang disediakan karena setiap mode memiliki bentuk dan antarmuka yang berbeda serta dapat diedit sesuai dengan keinginan pengguna namun belum mendukung *motion control* didalamnya. Dengan konsep yang terdapat pada kedua *game* tersebut dapat dikembangkan *game* yang menyenangkan dan menghibur dengan memanfaatkan *gesture recognition* ke dalam *game* ritme.

3.1.3 Analisis Sistem yang akan Dibangun

Dari masalah yang ada, solusi yang diberikan adalah membangun *game* dance touch yang dapat mengimplementasikan sistem *gesture recognition* dengan memanfaatkan webcam untuk medeteksi pergerakan jari tangan, melacak gerakan jari tangan dan mengubah inputan gerakan jari tangan tersebut menjadi alat untuk mengontrol *game* yang dibangun. Untuk mengimplementasikan *gesture recognition* menggunakan *library* OpenCV yang telah tersedia, *library* ini berfungsi untuk membuat sensor kamera dapat mengenali tangan dan gerakan tangan dari pengguna. Berikut ini adalah gambar analisis sistem yang akan di bangun dapat dilihat pada gambar 3.15.



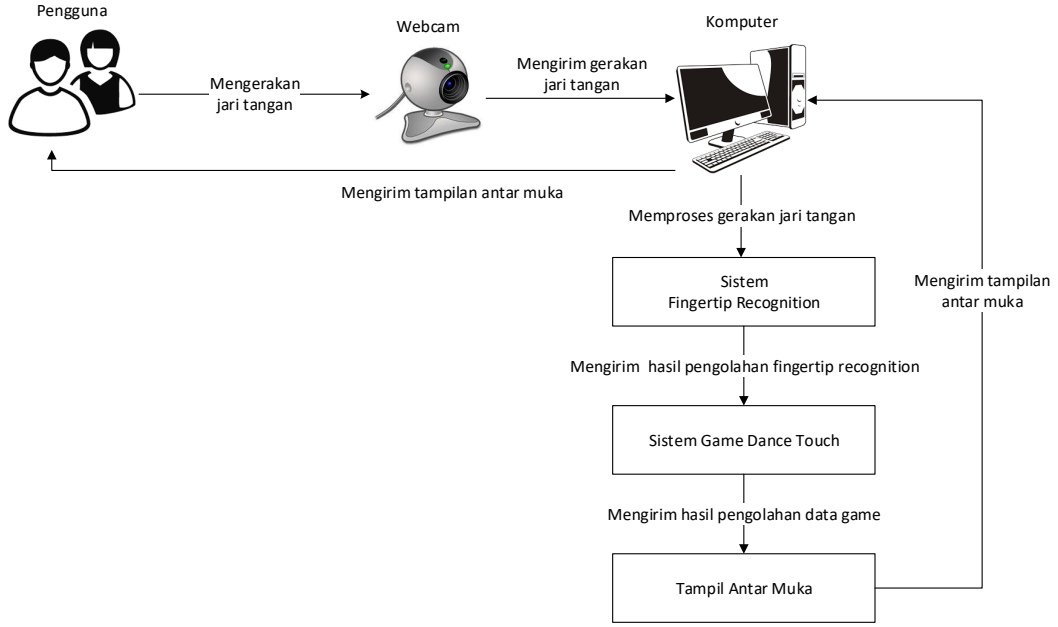
Gambar 3.15 Analisis Sistem Yang Akan Dibangun

Berikut ini adalah alur pada sistem *game* dance touch yang akan dibangun :

1. Pengguna menggerakan jari tangan di depan webcam.
2. Webcam merekam dan melacak gerakan jari tangan dari pengguna.
3. Hasil rekaman gerakan jari tangan pengguna dikirim ke komputer.
4. Komputer akan memproses gerakan jari tangan yang di terima menjadi inputan ke dalam *game*.
5. Proses gerakan jari tangan yang ada pada *game* dance touch akan dibuat dalam 2 mode yang dapat digunakan yaitu mode *cursor* dan mode *gesture*.
6. Mode *cursor* akan digunakan untuk bernavigasi pada menu *game* dance touch. Pengguna menggerakan jari tangan kiri atau jari tangan kanan di depan webcam untuk memilih menu yang berada dalam *game* dance touch yang akan di bangun.
7. Mode *gesture* akan digunakan untuk memainkan isi dari *game* dance touch. Pengguna menggerakan jari tangan kiri atau jari tangan kanan sesuai *gesture* yang akan muncul pada *gameplay* yang ada di dalam *game* dance touch yang akan di bangun.

3.1.4 Analisis Arsitektur Sistem

Arsitektur sistem yang akan dibangun terdiri dari beberapa bagian yaitu mengenai analisis sistem *fingertip tracking* dan sistem *game* dance touch. Arsitektur pembangunan *game* dance touch ditunjukan pada gambar 3.16 berikut :



Gambar 3.16 Arsitektur sistem

Beriikut ini adalah penjelasan alur dalam arsitektur sistem yang akan di bangun :

1. Pengguna diharuskan untuk meletakan jari tangan tepat di depan kamera yang tersedia.
2. Kamera akan menangkap inputan jari tangan dari pengguna.
3. Hasil dari inputan yang di tangkap akan dikirimkan kedalam komputer untuk di proses.
4. Komputer menerima inputan dari kamera dan memproses inputan gerakan tangan tersebut melalui dua tahapan yaitu sistem *fingertip recognition* dan sistem *game* dance touch.
5. Dalam sistem *fingertip recognition* akan diperiksa apakah objek yang diinputkan merupakan jari atau bukan.

6. Setelah objek telah berhasil di deteksi sebagai jari tangan maka akan di lakukan proses pencocokan bentuk jari tangan.
7. Setelah itu bentuk jari tangan yang telah terdeteksi tersebut akan dilacak pergerakan sehingga dari jari tangan dapat digunakan pada sistem *game* danece touch.
8. Dalam sistem *game* dance touch, inputan jari tangan yang telah diolah di dalam sistem *fingertip recognition* akan di fungsikan untuk mengontrol kursor dalam menu dan dapat digunakan untuk memainkan permainan.
9. Setelah proses di atas berhasil dilakukan maka akan tampilan pada antar muka akan di kirim ke komputer.
10. Komputer akan menampilkan antarmuka dari *game* dance tiuch kepada pengguna antarmuka yang telah di proses.

3.1.5 Analisis Metode Fingertip Tracking

Analisis metode *fingertip recognition* menggambarkan kebutuhan sistem untuk dapat menyediakan sebuah fungsi pendeteksi suatu objek dan fungsi pelacakan gerakan objek dalam *game* dance touch. Dalam penelitian ini metode *fingertip recognition* dibangun sebuah sistem yang diberinama sistem *fingertip recognition* yang dibangun berpusat pada pengenalan jari tangan dan pelacakan gerakan jari tangan dari pengguna. Sistem *fingertip recognition* dapat digunakan sebagai kursor untuk bernavigasi di dalam menu *game* selayaknya *mouse* pada mode *gesture* pada menu dan sebagai alat mengotrol *gameplay* dari *game* dance touch pada mode *gesture* pada *gameplay*.

Prinsip kerja dari sistem *fingertip recognition* dalam *game* dance touch adalah mendeteksi objek jari tangan manusia yang di tangkap oleh kamera atau webcam. Objek yang ditangkap oleh kamera akan dipetakan dan dilacak pergerakannya. Setelah dilakukan pemetaan objek akan di deteksi bentuk dan pergerakan dari objek itu sendiri. Oleh sebab itu untuk memudahkan pembuatan sistem dibutuhkan sebuah pendefinisian bentuk dan pola jari tangan apa saja yang akan digunakan pada sistem *fingertip recognition*. Berikut adalah pendefinisian

fingertip recognition yang akan digunakan pada *game* dance touch pada tabel berikut:

Tabel 3.4 Fingertip Recogntion

Gesture		**Interaksi**
Mode bentuk jari tangan pada menu	Jari telunjuk	Menggerakan kursor
	Mengepal tangan	Mengklik tombol yang dipilih menggunakan jari telunjuk
Mode bentuk jari tangan dalam permaian	Merapatkan kelima jari tangan	Merupakan gestur ini tidak akan dinilai oleh sistem dan digunakan hanya sebagai lepas tombol yang di klik
	Jari telunjuk dan ibu jari	Mengaktifkan salah satu tombol yang tersedia
	Jari telunjuk, jari tengah dan jari manis	Mengaktifkan salah satu tombol yang tersedia
	Jari telunjuk dan jari tengah	Mengaktifkan salah satu tombol yang tersedia
	Jari telunjuk, jari tengah dan ibu jari	Mengaktifkan salah satu tombol yang tersedia
	Jari telunjuk dan jari tengah	Mengaktifkan salah satu tombol yang tersedia
	Lima jari	Membuka menu *pouse* pada *gameplay*

Fingertip recognition tersebut digunakan untuk memilih tombol - tombol yang akan muncul pada layar *game* dance touch.

3.1.5.1 Proses Deteksi Jari Tangan

Proses deteksi jari tangan menggunakan haar cascade yang terdapat di dalam OpenCV. Untuk melakukan deteksi jari tangan diperlukan data latih yang di dalamnya berisi gambar – gambar yang berisi jari tangan dan latar belakang tanpa jari tangan. Setelah data latih di buat maka akan menhasilkan file dengan format .xml yang akan digunakan pada program.

3.1.5.1.1 Data Latih

Pada proses latih ini membutuhkan data latih berupa gambar positif dan negatif. Gambar positif merupakan gambar yang di dalamnya terdapat objek yang akan di deteksi, dalam penelitian ini objek yang akan dideteksi yaitu gambar jari tangan. Sedangkan gambar negatif merupakan data gambar latar belakang yang berwarna hitam putih.

Berikut ini adalah gambar 3.17 contoh data positif dan gambar 3.18 contoh data negatif dapat dilihat berikut ini :



Gambar 3.17 Contoh data positif



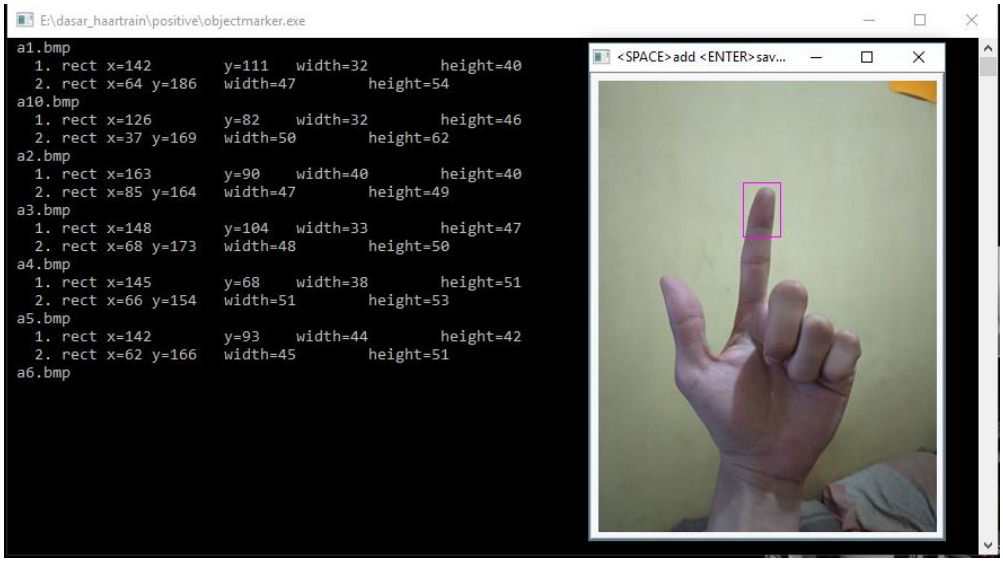
Gambar 3.18 Contoh data negatif

Untuk data positif membutuhkan 150 sampel gambar, sedangkan untuk data negatif membutuhkan 600 *sampel* gambar. Dari seluruh data positif dan negatif yang dimiliki akan dilatih dan menghasilkan sebuah file .xml yang berisi tentang hasil olahan data latih. Untuk proses data latih dilakukan dengan beberapa cara yaitu menandai data positif, membuat sample dataset positif, melatih haar-cascade dan mengubah hasil data latih ke dalam file berformat .xml.

3.1.5.1.1.1. Menandai data positif

Untuk dapat menandai data positif membutuhkan *tool* dari opencv bernama objectmarker.exe. Program ini akan digunakan unutuk menandai data positif secara manual. Hasil yang didapatkan yaitu data posisi x dan y serta besar

ukuran dari pixel yang ditandai, berikutnya akan di simpan dengan format .txt. berikut adalah gambar dari proses menandai objek jari tangan pengguna menggunakan objectmarker.exe dapat dilihat pada gambar 3.19.



Gambar 3.19 Menandai objek jari tangan pengguna menggunakan objectmarker.exe

3.1.5.1.1.2. Membuat sampel data positif.

Hasil dari objectmarker.exe yang di simpan dalam file berformat .txt akan digunakan untuk membuat datalist vektor dengan format .vec. Hasil dari data vektor yang dibuat akan gunakan dalam tahap berikutnya yaitu melatih data haar-cascade. Untuk proses membuat *sample* data positif pada program objectmarker.exe dapat dimasukan perintah sebagai berikut.

```
createsamples.exe -info positive/info.txt -vec vector/vector.vec -num 150 -w 40 -h 40
```

Gambar 3.20 Perintah createsample.exe.

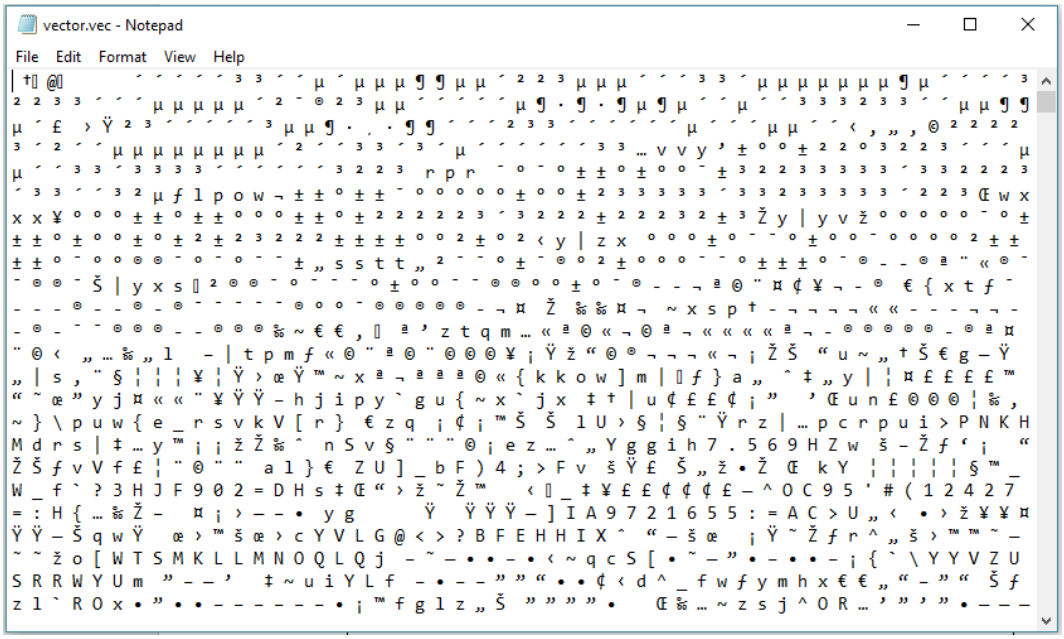
Keterangan dari perintah createsampel.exe dapat dilihat pada tabel 3.5. berikut ini:

Tabel 3.5 Keterangan dari perintah createsample.exe

Komponen	Keterangan
-info	merupakan datalist dari hasil objectmarker.exe yang di simpan dalam sebuah file berformat .txt
-num	merupakan jumlah sample positif yang akan dibuatkan dengan gambar latar belakang.
-vec	merupakan file yang berisi vektor dari data positif yang telah dibuat.

-w	merupakan lebar dari pixel pada gambar positif yang akan diolah
-h	merupakan tinggi dari pixel pada gambar positif yang akan diolah

File .vec yang telah dibuat akan dibutuhkan pada tahap berikutnya yaitu tahap data latih cascade. Berikut adalah gambar 3.21 yaitu isi dari file vektor yang telah dibuat.



Gambar 3.21 File vektor.vec.

3.1.5.1.1.3. Data latih haar-classifier

Untuk melatih haar-classifier membutuhkan *tools* bawaan dari OpenCV yang bernama haartraining.exe. Program ini digunakan untuk membuat data latih haar-cascade dan menyimpan hasil data latih ke dalam file yang bernama AdaBoostCARTHaarClassifier.txt. Data tersebut disimpan dalam sebuah folder bernama cascade dan dibagi lagi dalam beberapa subfolder yang diberi nama sesuai jumlah *stage* yang telah ditetapkan.

Untuk proses membuat data latih haar-cascade pada program haartraining.exe dapat dimasukan perintah sebagai berikut.

```
haartraining.exe -data cascades -vec vector/vector.vec -bg negative/bg.txt -npos 150 -nneg 150 -nstages 20 -mem 2046 -mode ALL -w 40 -h 40
```

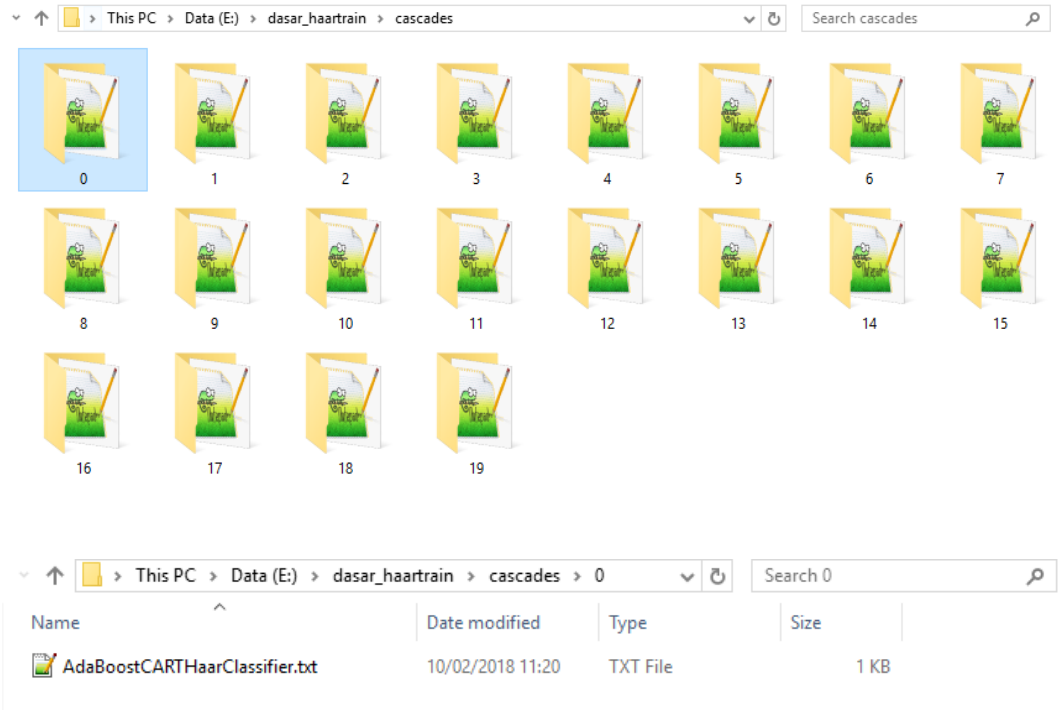
Gambar 3.22 Perintah pada haartraining.exe.

Keterangan dari perintah train cascade dapat dilihat pada tabel 3.6. berikut ini:

Tabel 3.6 Keterangan dari perintah haartraining.exe.

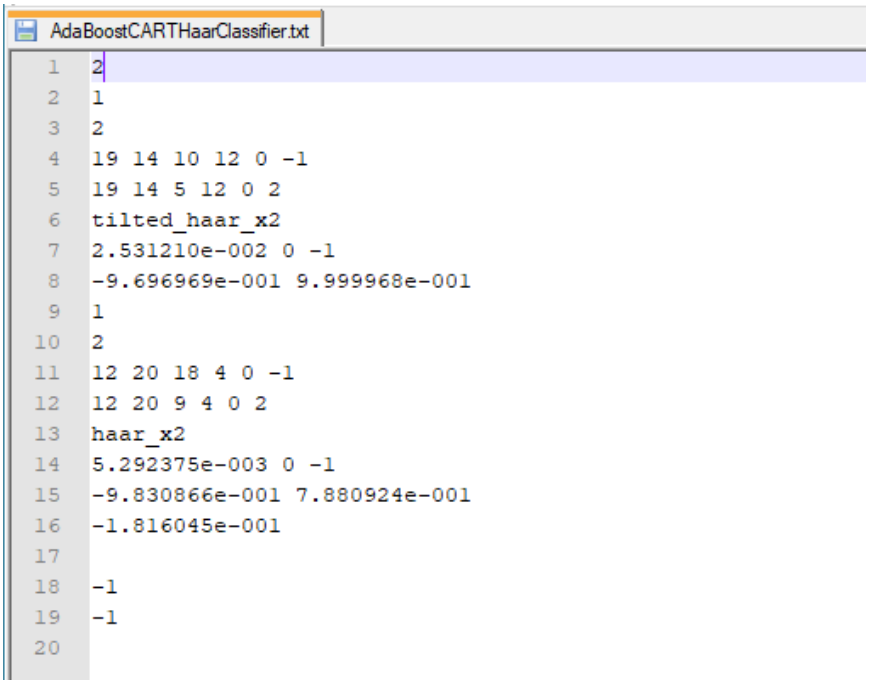
Komponen	Keterangan
-data	merupakan lokasi penyimpanan dari latih cascade
-bg	merupakan datalist dari data negatif yang diperlukan yang berformat .txt
-vec	merupakan data vektor dari data positif yang telah dibuat pada program createsample.exe.
-mode	merupakan pilihan mode yang digunakan untuk proses latih yaitu basic atau all.
-numStages	merupakan jumlah *stage* yang di lakukan untuk melatih data latih haar-cascade.
-w	merupakan lebar dari pixel pada gambar positif yang akan diolah
-h	merupakan tinggi dari pixel pada gambar positif yang akan diolah

Berikut ini adalah gambar dari isi folder cascade dan subfolder sesuai jumlah stage yang telah di tentukan dapat dilihat pada gambar 3.23.



Gambar 3.23 Isi folder cascade dan isi subfolder *stage*.

Berikut merupakan salah satu contoh isi dari file AdaBoostCARTHaarClassifier.txt dapat dilihat pada gambar 3.24 berikut.



Gambar 3.24 Isi file AdaBoostCARTHaarClassifier.txt.

3.1.5.1.1.4. Mengubah data haar-classifier ke file .xml

Hasil data latih haar-classifier yang telah dilakukan perlu untuk di ubah ke dalam bentuk format file .xml. Untuk dapat merubah hasil data latih haar-cascade akan menggunakan sebuah *tools* dari OpenCV yang bernama haarconv.exe. Program ini akan digunakan untuk mengubah semua file yang terdapat pada folder cascade kedalam file berformat .xml. File dengan format .xml inilah yang dibutuhkan dalam program utama, didalam file ini tersimpan data yang berisi informasi tentang bentuk jari tangan pengguna yang telah dilatih sebelumya.

Untuk proses mengubah data latih haar-classifier ke dalam file .xml dapat dilakukan dengan memasukan perintah sebagai berikut :

```
haarconv.exe cascades finger.xml 40 40
```

Gambar 3.25 Perintah pada traincascade.exe.

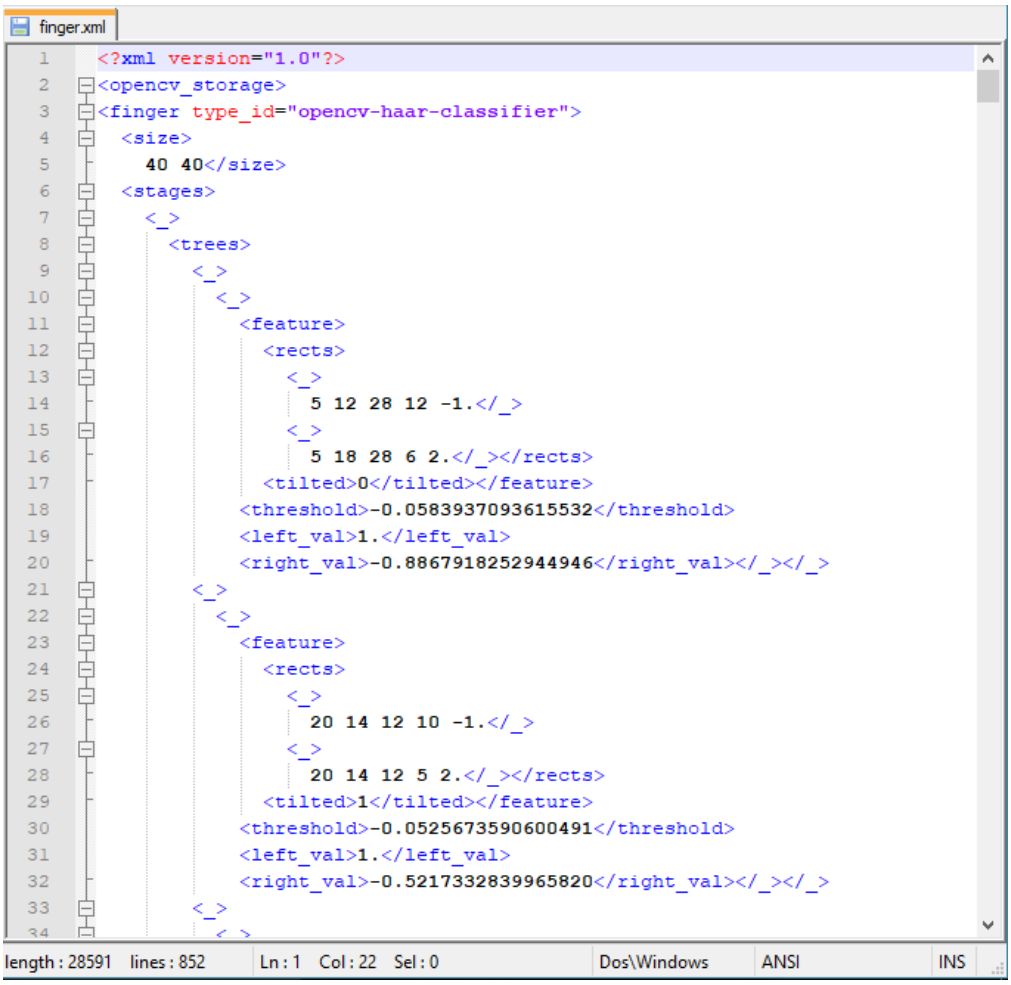
Keterangan dari perintah train cascade dapat dilihat pada tabel 3.7. berikut ini:

Tabel 3.7 Keterangan dari perintah train cascade.

Komponen	Keterangan
cascades	merupakan lokasi penyimpanan dari latih cascade
finger.xml	merupakan file yang akan dibuat dari file yang berada pada folder cascade ke dalam bentuk file .xml yang diberi nama finger.xml

40 40	merupakan lebar dan tinggi dari pixel pada gambar positif yang akan diolah

Beriku merupakan isi dari file .xml yang dapat dilihat pada gambar 3.26 berikut ini:



Gambar 3.26 Isi file .xml

3.1.5.2 Cara Menggunakan Metode Fingertip Tracking

Berikut adalah cara kerja dari masing-masing gerakan jari tangan yang di bagi menjadi 2 mode yaitu dalam mode bentuk jari tangan pada menu dan bentuk jari tangan dalam permaian :

3.1.5.2.1 Mode bentuk jari tangan pada menu

3.1.5.2.1.1 Jari telunjuk

Jari telunjuk akan digunakan sebagai kursor di dalam *game* dance touch. Untuk membaca pergerakan jari telunjuk dalam *game* dance touch berikut ini adalah fase untuk membentuk *gesture* jari telunjuk sebagai kursor :

a. Mendeteksi jari telunjuk pengguna.
b. Memeriksa kesesuaian inputan bentuk jari telunjuk pengguna dalam file telunjuk.xml.
c. Jika yang terdekteksi jari telunjuk pengguna maka akan diberikan sebuah penanda pada ujung jari telunjuk yang berbentuk lingkaran, penanda tersebut akan menjadi titik pointer kursor dalam *game*.
d. Hasil deteksi jari telunjuk akan di proses sebagai pointer kursor.
e. Jika jari tangan pengguna digerakan maka dilakukan proses pelacakan gerakan jari tangan pengguna sehingga pointer akan bergerak mengikuti gerakan jari tangan pengguna.
f. Hasil pelacakan gerakan tersebut akan dikirim ke dalam sistem *game* sebagai kursor yang ada di dalam *game* dance touch.

Berikut ini adalah gambar 3.27 bentuk jari telunjuk yang akan di deteksi.



Gambar 3.27 Bentuk jari telunjuk

3.1.5.2.1.2 Mengepal tangan

Mengepal tangan merupakan *gesture* yang digunakan untuk mengklik atau memiilih sebuah tombol atau menu dalam *game* dance touch. Untuk membaca gerakan mengepalkan tangan yaitu dengan cara berikut :

a. Mendeteksi kepalan tangan pengguna.
b. Memeriksa kesesuaian inputan bentuk kepalan tangan pengguna dalam file berformat mengepal.xml.
c. Jika yang terdeksi merupakan kepalan tangan penguna maka sistem akan memberi penanda pada jari tangan yang terdeteksi dan memproses *gesture* tersebut menjadi fungsi klik tombol.

Berikut ini adalah gambar 3.28 bentuk mengepalkan tangan yang akan di deteksi.



Gambar 3.28 Bentuk kepalan jari tangan

3.1.5.2.2 Mode bentuk jari tangan dalam permaian

3.1.5.2.2.1 Jari telunjuk, jari tengah dan jari manis

Jari telunjuk, jari tengah, dan jari manis merupakan *gesture* untuk mengklik tombol yang berbentuk jari telunjuk, jari tengah, dan jari manis. Untuk membaca gerakan jari telunjuk, jari tengah, dan jari manis yaitu dengan cara berikut :

a. Mendeteksi bentuk jari telunjuk, jari tengah dan jari manis pengguna.
b. Memeriksa kesesuaian inputan bentuk jari telunjuk, jari tengah dan jari manis pengguna dalam file telunjuk_tengan_manis.xml.

c. Jika yang terdekteksi jari telunjuk, jari tengah dan jari manis maka sistem akan memberi penanda pada jari tangan yang terdeteksi dan mengirim hasil deteksi jari tersebut ke dalam *game* sebagai fungsi klik tombol *gesture* yang berbentuk jari telunjuk, jari tengah dan jari manis.
d. Jika tidak terdeteksi maka sistem tidak akan memberikan respon apapun kedalam *game*.

Berikut ini adalah gambar 3.29 bentuk jari telunjuk, jari tengah dan jari manis yang akan di deteksi.



Gambar 3.29 Bentuk jari telunjuk, jari tengah, dan jari manis

3.1.5.2.2.2 Jari telunjuk dan jari tengah

Jari telunjuk dan jari tengah merupakan *gesture* untuk mengklik tombol yang berbentuk jari telunjuk dan jari tengah. Untuk membaca gerakan jari telunjuk dan jari tengah yaitu dengan cara berikut :

a. Mendeteksi bentuk jari telunjuk dan jari tengah pengguna.
b. Memeriksa kesesuaian inputan bentuk jari tangan pengguna dalam file berformat telunjuk_tengah.xml.
c. Jika yang terdekteksi jari telunjuk dan jari tengah maka sistem akan memberi penanda pada jari tangan yang terdeteksi dan mengirim hasil deteksi jari tersebut ke dalam *game* sebagai fungsi klik tombol *gesture* yang berbentuk jari telunjuk dan jari tengah.
d. Jika tidak terdeteksi maka sistem tidak akan memberikan respon apapun kedalam *game*.

Berikut ini adalah gambar 3.30 bentuk jari telunjuk dan jari tengah yang akan di deteksi.



Gambar 3.30 Bentuk jari telunjuk dan jari tengah.

3.1.5.2.2.3 Jari telunjuk, jari tengah dan ibu jari

Jari telunjuk, jari tengah dan ibu jari merupakan *gesture* untuk mengklik tombol yang berbentuk jari telunjuk, jari tengah dan ibu jari. Untuk membaca gerakan jari telunjuk, jari tengah dan ibu jari yaitu dengan cara berikut :

a. Mendeteksi jari telunjuk, jari tengah dan ibu jari pengguna.
b. Memeriksa kesesuaian inputan bentuk jari tangan pengguna dalam file telunjuk_tengah_ibujari.xml.
c. Jika yang terdekteksi jari telunjuk, jari tengah dan ibu jari maka sistem akan memberi penanda pada jari tangan yang terdeteksi dan mengirim hasil deteksi jari tersebut ke dalam *game* sebagai fungsi klik tombol *gesture* yang berbentuk jari telunjuk, jari tengah dan ibu jari.
d. Jika tidak terdeteksi maka sistem tidak akan memberikan respon apapun kedalam *game*.

Berikut ini adalah gambar 3.31 bentuk jari telunjuk, jari tengah dan ibu jari yang akan di deteksi.



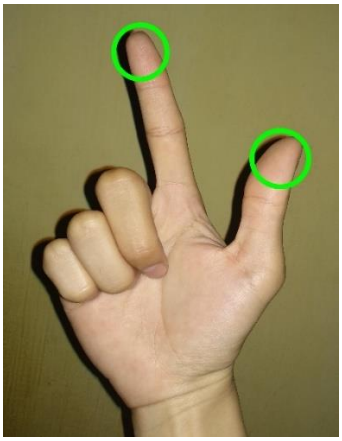
Gambar 3.31 Bentuk jari telunjuk, jari tengah dan ibu jari

3.1.5.2.2.4 Jari telunjuk dan ibu jari

Jari telunjuk dan ibu jari merupakan *gesture* untuk mengklik tombol yang tersedia sesuai dengan bentuk jari tangan. Untuk membaca gerakan jari telunjuk dan ibu jari yaitu dengan cara berikut :

a. Mendeteksi jari telunjuk dan ibu jari pengguna.
b. Memeriksa kesesuaian inputan bentuk jari tangan pengguna dalam file telunjuk_ibujari.xml.
c. Jika yang terdekteksi jari telunjuk dan ibu jari maka sistem akan memberi penanda pada jari tangan yang terdeteksi dan mengirim hasil deteksi jari tersebut ke dalam *game* sebagai fungsi klik tombol *gesture* yang berbentuk jari telunjuk dan ibu jari.
d. Jika tidak terdeteksi maka sistem tidak akan memberikan respon apapun kedalam *game*.

Berikut ini adalah gambar 3.32 bentuk jari telunjuk dan ibu jari yang akan di deteksi.



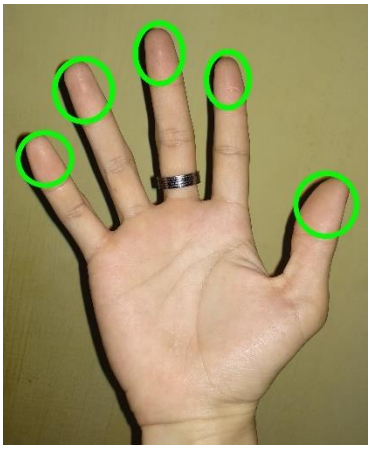
Gambar 3.32 Bentuk jari telunjuk dan ibu jari

3.1.5.2.2.5 Lima jari tangan terbuka

Lima jari tangan terbuka merupakan *gesture* untuk mengklik tombol jedah dalam permaianan. Untuk membaca gerakan lima jari yaitu dengan cara berikut :

a. Mendeteksi lima jari tangan pengguna.
b. Memeriksa kesesuaian inputan bentuk jari tangan pengguna dalam file tgnopen.xml.
b. Jika yang terdekteksi kelima jari maka sistem akan memberi penanda pada jari tangan yang terdeteksi dan mengirim hasil deteksi jari tersebut ke dalam game sebagai fungsi klik tombol jedah yang ada di dalam permaianan.

Berikut ini adalah gambar 3.33 bentuk kelima jari yang akan di deteksi.



Gambar 3.33 Bentuk kelima jari

Bahasa pemrograman yang digunakan adalah C# dengan perangkat lunak Unity 3D dan menggunakan *library* OpenCV 2.1. Dari seluruh insialisasi bentuk

jari tangan diatas, diubah sebagai inputan *game* dance touch. Sebagai contoh jika pengguna ingin menekan tombol yang berbentuk jari telunjuk dan jari tengah terangkat maka penggun dapat melakukan gerakan jari telunjuk dan jari tengah terangkat. Langkah-langkahnya proses pengenalan bentuk jari tangan pengguna dapat dilihat pada gambar berikut :

```
/*
perintah OpenCV untuk membaca file .xml yang berisi informasi
tentang jari tangan pengguna yang terdapat dalam directory
komputer salah satu contoh memanggil filel telunjuk.xml
*/
cascade =
CvHaarClassifierCascade.FromFile("./Assets/telunjuk.xml");

/*
perintah untuk menggunakan file .xml yang telah dipanggil
sebelumnya salah satu contoh memanggil filel telunjuk.xml
*/
CvSeq<CvAvgComp> finger = Cv.HaarDetectObjects(smallImg,
cascade, storage, ScaleFactor, MinNeighbors, 0, new CvSize(64,
64));

/*
perintah untuk mendapatkan nilai Rect dari jari tangan yang di
deteksi yang kemudian digunakan untuk menggambar lingkaran atau
kotak
*/
for (int i = 0; i < finger.Total; i++){
 CvRect r = finger[i].Value.Rect;
 if (ImageTracker == imgTracker.Circle){
  CvPoint center = new CvPoint{
  X = Cv.Round((r.X + r.Width * 0.5) * Scale),
  Y = Cv.Round((r.Y + r.Height * 0.5) * Scale)
 };
 int radius = Cv.Round((r.Width + r.Height) * 0.3 * Scale);
 img.Circle(center, radius, TrackerColor, 1,
 LineType.AntiAlias, 0);
 } else {
  CvPoint pt1 = new CvPoint{
  X = r.X,
  Y = r.Y
 };
  CvPoint pt2 = new CvPoint{
  X = r.Width,
  Y = r.Height
 };

 img.Rectangle(r, TrackerColor, 1);
 }
}
```

```
/*
perintah untuk mengkopi gambar yang dihasilkan oleh OpenCV ke
variabel image
*/
texture.SetPixels(cols);

/*
memasangkan gambar hasil akhir ke variabel image yang telah
dibuat dan menampilkan hasil setiap framenya
*/
texture.Apply();
GetComponent<Renderer>().material.mainTexture = texture;
```

Gambar 3.34 Pseudocode mendeteksi jari tangan

3.1.6 Analisis *Game* Yang Akan Di Bangun

Game dance touch yang akan dibangun dalam grafis 2D, *game* ini mempunyai *gameplay* yang mirip dengan *game* sejenisnya seperti osu! namun yang membedakan adalah cara mengontrol *gameplay* seperti *game* just dance. Berikut adalah fitur yang ditawarkan ke dalam *game* ini :

1. *Single player*
2. *Game* dibangun dengan grafis 2D.
3. Pengguna dapat menggunakan gerakan untuk berinteraksi dengan sistem.

3.1.6.1 Deskripsi Konsep *Game* Dance Touch

Game dance touch yang dibangun bersifat *single player*. *Game* ini dibangun agar pemain dapat menikmati *game* musik sambil ikut bergerak untuk menyelesaikan *gameplay* tersebut. Sebelum memulai *game* pada tamplan awal akan disediakan tampilan untuk membaca tangan dari pengguna. Setelah proses tersebut selesai maka masuk ke dalam menu utama yang di dalamnya terdapat beberapa menu lain seperti *start*, *option* dan *exit*. Untuk dapat bermain pengguna tingal masuk ke menu memilih musik dengan mengklik start sedangkan untuk pengaturan dapat mengklik options. Terdapat pula antarmuka untuk melihat skor yang didapat oleh pemain pada antarmuka skoring. Untuk dapat memainkan ini pengguna harus menggerakan tangannya untuk memilih menu dan Pada tabel 3.8. akan menjelaskan deskripsi konsep *game* dance touch.

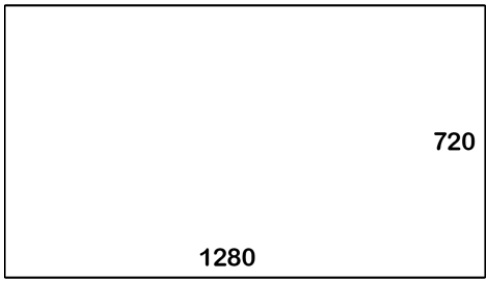
Tabel 3.8 Deskripsi Konsep *Game*

Unsur yang dibangun	Deaskripsi
Judul	*Game* Dance Touch.
Audio	Audio yang digunakan merupakan musik yang telah di edit dan dimasukan dalam *game*.
Grafis	Menggunakan grafis 2 dimensi.
Gambar	Gambar yang ada dalam *game* terdiri dari gambar yang mendeskripsikan *game* musik.
Animasi	Animasi yang digunakan dalam *game* ini merupakan animasi sederhana seperti membuka memperbesar sedangkan kembali memperkecil. Animasi juga di tambahkan ada saat menekan tombol note dalam permainan.
Interaktivitas	Interaktivitas dalam *game* dance touch ini adalah tombol lanjut, kembali dan ulangi.

Untuk menghasilkan *game* ritme yang sesuai dengan tujuan utama yaitu dapat mengimplementasikan *gesture reconition*, dibutuhkan konsep secara visual untuk mendukung tampilan *game* dan fungsionalitas yang terdapat di dalam *game*. Berikut ini adalah konsep visual *game* dance touch :

1. Ukuran Layar

   Game dance touch dirandan dengan ukuran 1280×720px.



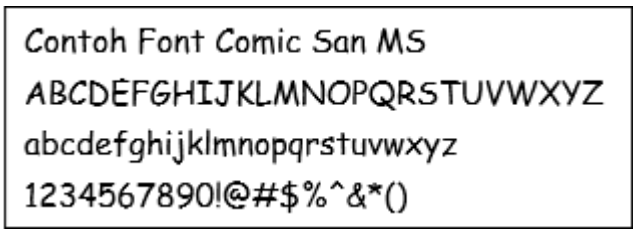
Gambar 3.35 Gambar ukuran layar

2. Layout

   Layout atau tata letak merupakan bagian dari desain visual yang penting sebab layout yang tertata rapi dapat sebagai sarana untuk menyampaikan informasi kepada pengguna agar tidak terjadi kesalahan penyampaian informasi.

3. Huruf

   Perancangan huruf yang digunakan dalam *game* dance touch adalah comic san. Comic san digunakan untuk judul, deskripsi musik dan navigasi.



Gambar 3.36 Font comic san ms

4. Warna

   Warna yang digunakan dalam *game* dance touch adalah warna kombinasi warna biru dan warna coklat.



Gambar 3.37 Kombinasi warna

3.1.6.2 *Gameplay Game* Dance Touch

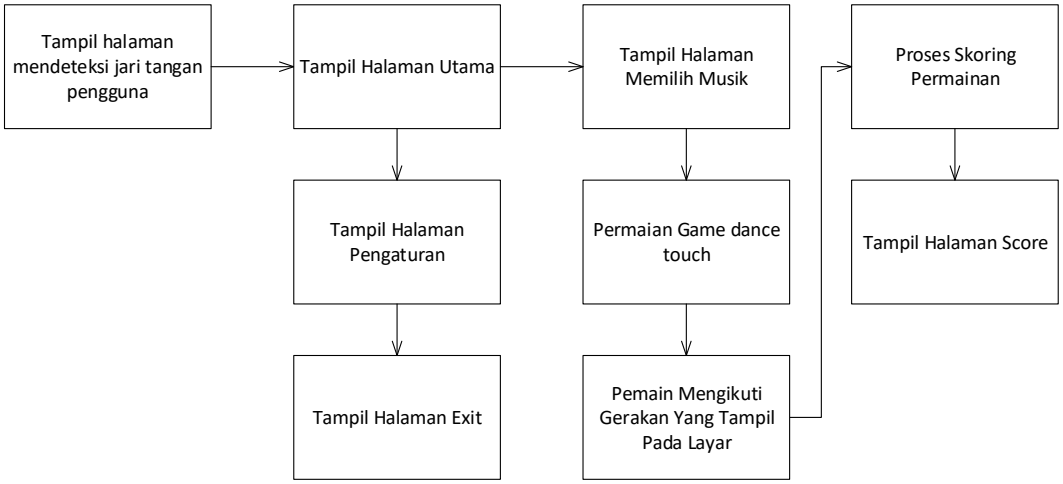
Game dance touch yang dibangun bersifat *single player*. *Game* ini dibangun agar pemain dapat menikmati *game* musik sambil ikut bergerak untuk menyelesaikan *gameplay* tersebut. Pada tampilan awal disediakan antarmuka untuk mengidentifikasi jari tangan yang akan digunakan sebagai *controller*. Proses pengidentifikasian dengan cara pemain meletakan tangan tepat di depan kamera, setelah itu tunggu beberapa saat untuk membaca jari tangan pengguna. Setelah selesai dari proses identifikasi maka pengguna dapat mengerakan kursor untuk bernavigasi dan dapat memilih beberapa tombol di dalam menu utama antraa lain :

1. Tombol *play* berfungsi untuk masuk ke menu memilih musik.
2. Tombol *option* berfungsi masuk ke menu option.
3. Tombol *exit* berfungsi untuk keluar dari permainan.

Pada halaman memilih musik pemain di minta untuk memilih sebuah musik dan tingkatan levelnya. Pada menu memilih musik terdapat deskripsi dari musik tersebut serta diputar cuplikan musik yang dimainkan selama beberapa

detik. Terdapat beberapa tombol pada menu Setelah beres memilih masuk ke dalam antarmuka *gameplay* ritme.

Sebelum memulai *gameplay* akan ada panduan cara memainkan permainan. Setelah itu muncul antarmuka *gameplay* ritme pemain akan di minta untuk mengikuti gerakan yang muncul di layar, jika pemain gagal mengikuti gerakan tangan yang diminta maka tidak akan mendapat point untuk skoring sebaliknya jika pemain berhasil mengikuti gerakan maka akan mendapat 100 point setiap berhasil mengikuti gerakan yang tampil di layar. Pemain juga bisa mendapatkan skoring tambahan jika berhasil melakukan gerakan secara berturut – turut. Setelah selesai *gampelay* akan muncul antarmuka *result* yang berisi detail skor yang di dapat pemain. Berikut adalah gambar 3.38 *gameplay game* dance touch



Gambar 3.38 *Gameplay game* dance touch

3.1.6.3 Menghitung skor

Pada setiap *gameplay* pemain harus bisa menyelesaikan semua bentuk gerakan yang tampil pada layar, pemain akan mendapat penilaian berupa angka. Berikut tabel – tabel yang menjelaskan rincian perhitungan skor dalam *game* dance touch:

Tabel 3.9 Skor Permainan

Aksi yang dilakukan	Point yang di dapat
Gagal mengikuti gerakan	0
Mengikuti gerakan	100

Tabel 3.10 Skor jumlah gerakan yang diikuti secara berturut-turut

Banyak gerakan yang diikuti tanpa kesalahan	Point yang di dapat
0 - 7	100
8 - 15	100 × 2
16 - 23	100 × 3
≥ 24	100 × 4

Pada antarmuka skor akhir akan ada beberapa detail yang muncul selama satu kali memainkan isi dari *gameplay* seperti total *score*, *highscore*, *note hit*, *highest streak* (banyak gerakan yang diikuti tanpa kesalahan).

3.2 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang akan dibangun sesuai dengan perangkat keras, serta perangkat lunak. Analisis kebutuhan non fungsional ini bertujuan sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

3.2.1 Analisis Kebutuhan Perangkat Keras

Berikut ini adalah spesifikasi perangkat keras yang digunakan dalam pembangunan aplikasi pada tabel 3.11. Dan spesifikasi minimum perangkat keras yang dibutuhkan untuk menggunakan aplikasi pada tabel 3.12.

a. Perangkat keras pembangunan aplikasi.

Tabel 3.11 Spesifikasi Perangkat Keras Pembangunan Aplikasi

No.	Perangkat Keras	Spesifikasi
1	*Processor*	Quad Core – 1.80 GHz
2	*Graphic Card* (*VGA*)	2 GB + 4GB
3	*Memori*	4 GB DDR3
4	*Monitor*	12” HD
5	*Webcam*	Standar
6	*Harddisk*	500 GB
7	*Speaker*	Standar

b. Kebutuhan perangkat keras yang dibutuhkan untuk menjalankan aplikasi.

Tabel 3.12 Spesifikasi Minimum Perangkat Keras Yang Dibutuhkan

No.	Perangkat Keras	Spesifikasi

1	*Processor*	Dual Core 1.80 GHz
2	*Graphic Card* (*VGA*)	4GB
3	*Memori*	4 GB
4	*Monitor*	12” HD
5	*Webcam*	Standar Port USB 2.0
6	*Harddisk*	125 GB
7	*Speaker*	Standar

3.2.2 Analisis Kebutuhan Perangkat Lunak

Berikut ini adalah perangkat lunak yang digunakann dalam pembangunan aplikasi pada tabel 3.13.

Tabel 3.13 Perangkat Lunak

No	Perangkat Lunak	Spesifikasi
1	Sistem Operasi	Windows 10 64bit
2	Program	a. Microsoft Visual Studio 2010
		b. Microsoft Visio 2013
		c. Unity3D
		d. *Library* OpenCV

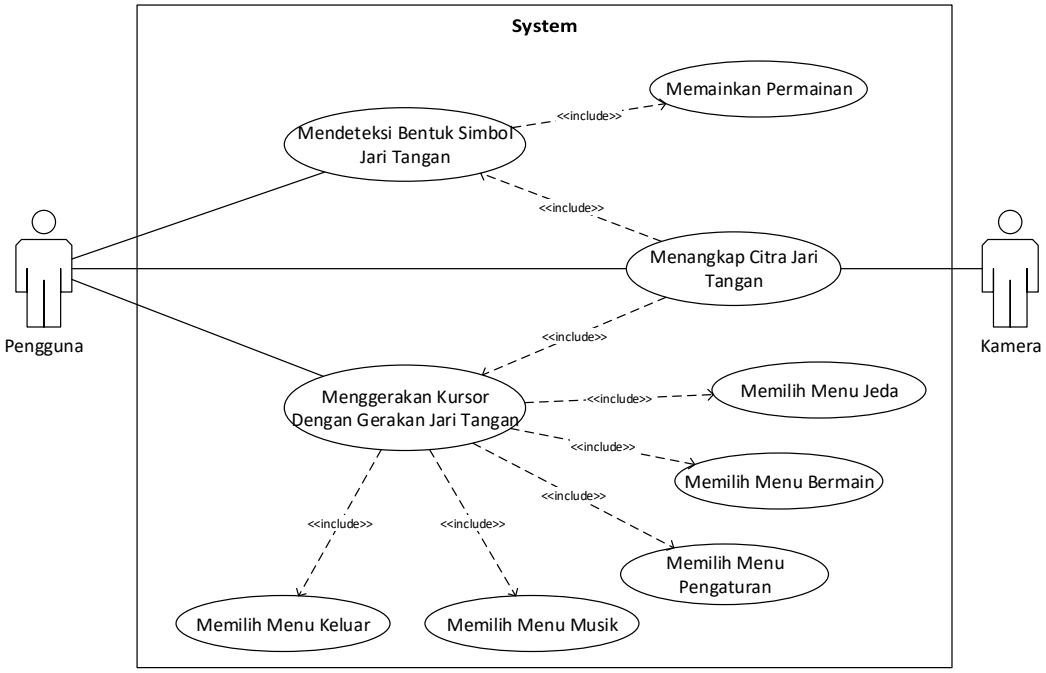
3.2.3 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional menggambarkan perancangan dari beberapa elemen terpisah kedalam satu kesatuan yang utuh dan berfungsi. Alat bantu yang digunakan untuk menggambarkan sistem secara umum yang akan dibangun yaitu dengan model *use case* dan perancangan antarmuka. Untuk menjelaskan bagaimana suatu masukan di proses pada sistem. Pemodelan sistem dimodelkan dengan menggunakan UML (*Unified Modeling Language*). Tahap-tahap pemodelan dalam analisis tersebut antara lain *Use Case*, *Use Case Scenario*, *Activity* Diagram, *Sequence* Diagram dan *Class* Diagram.

3.2.4 *Use Case* Diagram

Use case diagram merupakan pemodelan untuk perilaku (*behaviour*) sistem aplikasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem aplikasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem aplikasi dan apa saja yang dapat dilakukan menggunakan fungsi – fungsi itu.

Dalam *use case* diagram akan membahas tentang identifikasi aktor, identifikasi *use cas*, dan sekanario *use case*. Berikut adalah gambar 3.39 diagram *use case* yang di bangun :



Gambar 3.39 *Use Case* Diagram

Dalam sistem ini terdapat dua aktor dalam *use case* ini, berikut adalah penjelasnnya.

3.2.4.1 Identifikasi Aktor

Actor yaitu sesorang atau sesuatu seperti perangkat atau sistem lain yang mengakses *use case* dengan berperan sebagai pengguna yang akan menggunakan sistem. Actor

Tabel 3.14 Identifikasi Aktor

No.	Aktor	Definisi
1	Pengguna	Merupakan aktor dari aplikasi yang dibangun atau pengguna aplikasi.
2	Kamera	Merupakan aktor yang menjadi media untuk dapat mengambil citra jari tangan pengguna

3.2.4.2 Identifikasi *Use Case*

Berikut tabel Identifikasi *Use Case* yang terdapat pada aplikasi:

Tabel 3.15 Identifikasi *Use Case*

No.	Use Case	Definisi
1	Menangkap citra jari tangan	Fungsional untuk mengambil video input dari kamera yang secara otomatis menggambil video input dari jari tangan yang berjalan di *background*.
2	Mengenali bentuk simbol jari tangan	Fungsional untuk menampilkan menu utama yang telah disiapkan.
3	Menggerakan kursor dengan jari tangan	Fungsional untuk menggerakan kursor dalam *game* dengan menggerakan jari tangan pengguna
4	Memainkan permainan	Fungsional untuk memainkan *game*.
5	Memilih menu bermain	Fungsional untuk memilih menu bermain
6	Memilih menu pengaturan	Fungsional memilih menu yang berisi menu pengaturan dari *game*.
7	Memilih menu jedah	Fungsional untuk menghentikan sementara *game* yang berjalan di dalam *game*.
8	Memilih menu musik	Fungsional memilih menu yang berisi menu musik yang akan di pilih.
9	Memilih menu keluar	Fungsional untuk keluar dari *game*.

3.2.4.3 Skenario *Use Case*

Tabel 3.16 Skenario *Use Case* Menangkap Citra Jari Tangan

Skenario		
Nomor	1	
Nama	Menangkap citra jari tangan	
Tujuan	Menangkap jari tangan	
Deskripsi	Menangkap inputan objek merupakan jari tangan	
Aktor	Pengguna	
Skenario Utama		
Kondisi awal	Pengguna berada pada antarmuka menu integrasi jari tangan tahap pertama	
Aksi Aktor		**Reaksi Sistem**
		1. Menampilkan antarmuka menu integrasi jari tangan.
		2. Mengaktifkan kamera.
3. Menempatkan jari tepat di depan kamera.		
		4. Menangkap citra objek
5. Menyelesaikan menu integrasi jari tangan tahap pertama		

Tabel 3.17 Skenario *Use Case* Mendeteksi Bentuk Jari Tangan

Skenario	
Nomor	2
Nama	Mendeteksi bentuk simbol jari tangan

Tujuan	Mendeteksi bentuk simbol jari tangan pengguna
Deskripsi	Mendeteksi inputan apakah objek merupakan bentuk simbol jari tangan atau bukan
Aktor	Pengguna
Skenario Utama	
Kondisi awal	Pengguna berada pada antarmuka menu integrasi jari tangan tahap kedua
Aksi Aktor	**Reaksi Sistem**
	1. Menampilkan antarmuka menu integrasi jari tangan tahap kedua
	2. Mengaktifkan kemera
	3. Menampilkan bentuk jari tangan
4. Membentuk jari tangan seperti animasi yang tampil pada menu integrasi	
	5. Memeriksa jari tangan pengguna
	6. Memeriksa bentuk jari tangan
	7. Memeriksa simbol bentuk jari tangan yang tersedia
	8. Menampilkan animasi berhasil mengikuti bentuk jari tangan
9. Menyelesaikan menu integrasi jari tangan tahap kedua	

Tabel 3.18 Skenario *Use Case* Menggerakan Kursor dengan Gerakan Jari Tangan

Skenario	
Nomor	3
Nama	Menggerakan kursor dengan geraan jari tangan
Tujuan	Melacak gerakan jari tangan pengguna
Deskripsi	Melacak gerakan yang di lakukan jari tangan pengguna
Aktor	Pengguna
Skenario Utama	
Kondisi awal	Pengguna berada pada antarmuka menu integrasi jari tangan tahap ketiga
Aksi Aktor	**Reaksi Sistem**
	1. Menampilkan antarmuka menu integrasi jari tangan tahap ke tiga
	2. Mengaktifkan kamera
3. Menggerakan jari telunjuk ke berbagai arah.	
	4. Mendekteksi posisi gerakan jari tangan
	5. Memeriksa arah gerakan jari tangan
6. Mengklik tombol selesai dengan membentuk kepalan tangan	
	7. Memeriksa bentuk jari tangan
8. Menyelesaikan menu integrasi jari tangan tahap ketiga.	

Tabel 3.19 Skenario *Use Case* Memainkan Permainan

Skenario		
Nomor	4	
Nama	Memainkan permainan	
Tujuan	Menampilkan antarmuka dari permainan *game* dance touch.	
Deskripsi	Pengguna dapat memainkan *game* dengan menggunakan jari tangan	
Aktor	Pengguna	
Skenario Utama		
Kondisi awal	Pengguna berada pada antarmuka memainkan permainan	
Aksi Aktor		**Reaksi Sistem**
		1. Menampilkan antarmuka memainkan permainan
		2. Menyediakan tombol yang bersimbol bentuk jari tangan
3. Memainkan *game* dengan mengerakan jari mengikuti bentuk jari tangan yang muncul dengan irama musik.		
		4. Memeriksa bentuk jari tangan
		5. Memeriksa simbol bentuk jari tangan yang tersedia
		6. Menghitung skor
		1. Menampilkan antarmuka skor

Tabel 3.20 Skenario *Use Case* Memilih Menu Bermain

Skenario		
Nomor	5	
Nama	Menu bermain	
Tujuan	Menampilkan antarmuka menu bermain dari *game*	
Deskripsi	Menampilkan menu bermain dari *game*.	
Aktor	Pengguna	
Skenario Utama		
Kondisi awal	Pengguna berada pada antarmuka menu utama	
Aksi Aktor		**Reaksi Sistem**
1. Memilih menu bermain		
		2. Menampilkan menu bermain.

Tabel 3.21 Skenario *Use Case* Memilih Menu Musik

Skenario	
Nomor	6
Nama	Memilih menu musik
Tujuan	Menampilkan antarmuka menu musik dari *game*
Deskripsi	Menu musik berisi antarmuka untuk memilih musik yang ingin di mainkan beserta antarmuka tingkat kesulitan yang ingin dipilih

Aktor	Pengguna	
Skenario Utama		
Kondisi awal	Pengguna berada di antarmuka menu bermain	
Aksi Aktor		**Reaksi Sistem**
1. Memilih menu musik		
		2. Menampilkan menu musik..

Tabel 3.22 Skenario *Use Case* Memilih Menu Pengaturan

Skenario		
Nomor	7	
Nama	Memilih menu pengaturan	
Tujuan	Menampilkan antarmuka menu pengaturan dari *game*	
Deskripsi	Menu pengaturan berisi pengaturan yang disediakan seperti mengatur suara dan effek suara dalam *game*	
Aktor	Pengguna	
Skenario Utama		
Kondisi awal	Pengguna berada pada antarmuka menu utama	
Aksi Aktor		**Reaksi Sistem**
1. Memilih menu pengaturan		
		2. Menampilkan menu pengaturan.

Tabel 3.23. Skenario *Use Case* Memilih Menu Jedah

Skenario		
Nomor	8	
Nama	Memilih menu jedah	
Tujuan	Menghentikan sejenak dan menampilkan pop up antarmuka menu jedah di dalam permainan	
Deskripsi	*Game* akan terhenti sejenak dan akan tampil pop up antarmuka menu jedah di dalam permainan	
Aktor	Pengguna	
Skenario Utama		
Kondisi awal	Pengguna berada pada antarmuka memainkan permainan	
Aksi Aktor		**Reaksi Sistem**
1. Memilih menu jedah		
		2. Menampilkan menu jedah.

Tabel 3.24. Skenario *Use Case* Memilih Menu Keluar

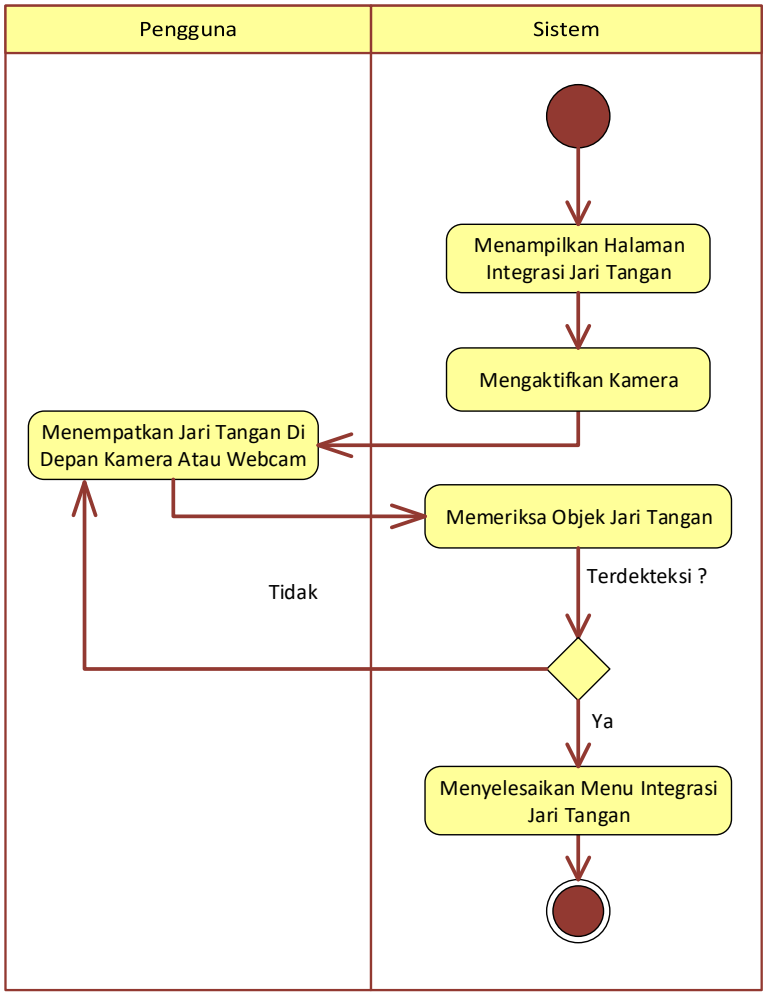
Skenario	
Nomor	9
Nama	Memilih Menu Keluar
Tujuan	Fungsi untuk keluar dari *game*
Deskripsi	Menu keluar adalah akses untuk menutup atau keluar dari *game*.
Aktor	Pengguna

Skenario Utama	
Kondisi awal	Pengguna berada pada antarmuka menu utama
Aksi Aktor	**Reaksi Sistem**
1. Memilih menu keluar.	
	2. Menghentikan permainan dan menutup permainan dari proses sistem operasi.

3.2.5 Aktivity Diagram

Activity diagram merupakan penggambaran dari sistem secara fungsional yang menjelaskan proses-proses logika atau fungsi yang dapat diimplementasikan melalui kode program. *Activity diagram* memodelkan event-event yang terjadi didalam suatu *Use Case* dan digunakan untuk pemodelan sistem. *Activity diagram* untuk masing-masing *use case* dari aplikasi ini yakni sebagai berikut :

1. Berikut ini adalah gambar 3.40 aktivity diagram menangkap citra jari tangan.

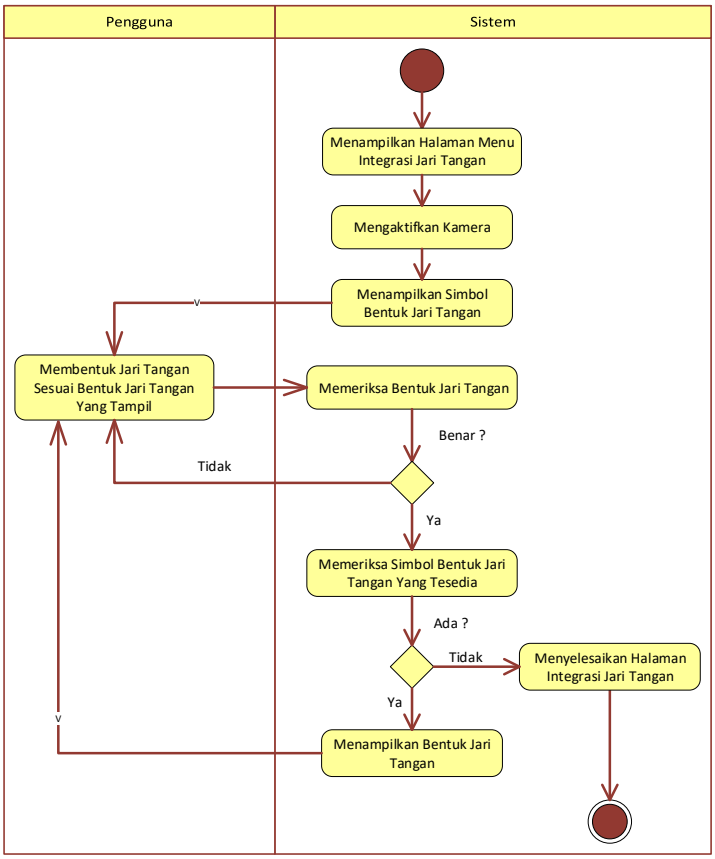


Gambar 3.40 *Activity diagram* Menangkap Citra Jari Tangan

Keterangan :

1. Sistem menampilkan halaman integrasi jari tangan.
2. Sistem mengaktifkan kamera.
3. Pengguna meletakan jari tepat di depan kamera.
4. Sistem akan mendeteksi apakah objek yang di input merupakan jari tangan atau bukan.
5. Sistem akan memeriksa apakah objek merupakan jari tangan atau bukan.
6. Jika tidak terdeteksi maka sistem akan mendeteksi kembali objek jari tangan.
7. Jika terdeteksi menyelesaikan halaman integrasi jari tangan.

2. Berikut ini adalah gambar 3.41 aktivity diagram mendeteksi bentuk jari tanggan pengguna.



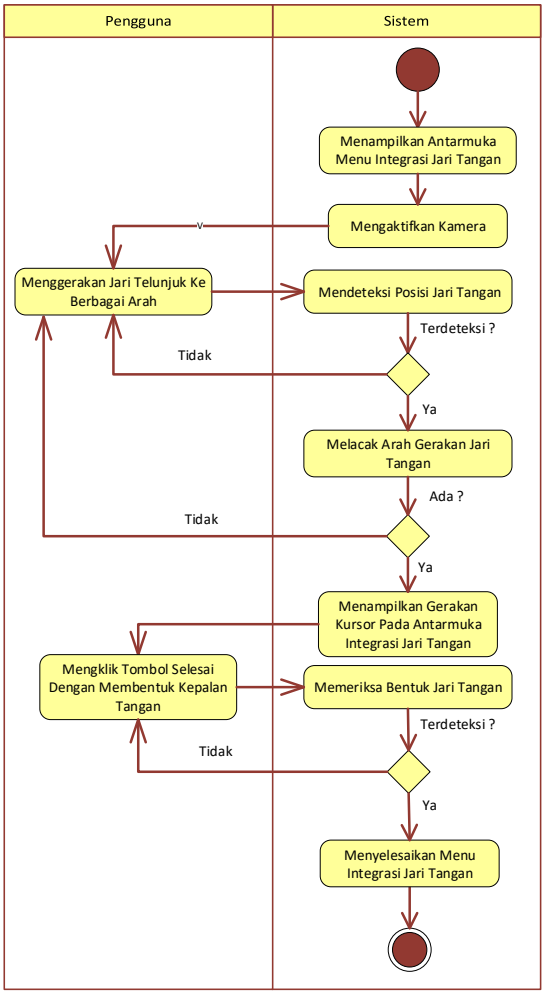
Gambar 3.41 *Activity* diagram Mendeteksi Bentuk Jari Tanggan Pengguna

Keterangan :

1. Sistem menampilkan halaman integrasi jari tangan.

2. Sistem mengaktifkan kamera.
3. Pengguna meletakan jari tepat di depan kamera.
4. Sistem akan memeriksa apakah objek yang di input merupakan jari tangan atau bukan.
5. Jika iya maka akan masuk ke tahap berikutnya yaitu memeriksa simbol bentuk jari tangan yang tersedia
6. Jika tidak maka kembali lgi ke proses memasukan inputan kembali oleh pengguna.
7. Setelah itu masuk ke tahap berikutnya yaitu memeriksa apakah masih ada simbol bentuk.
8. Jari tangan berikutnua jika iya maka akan menampilkan kembali simbol bentuk jari tangan
9. Jika tidak maka akan tampil mnyelesaikan halaman integrasi jari tangan.

3. Berikut ini adalah gambar 3.42 aktivity diagram mendeteksi bentuk jari tanggan pengguna.



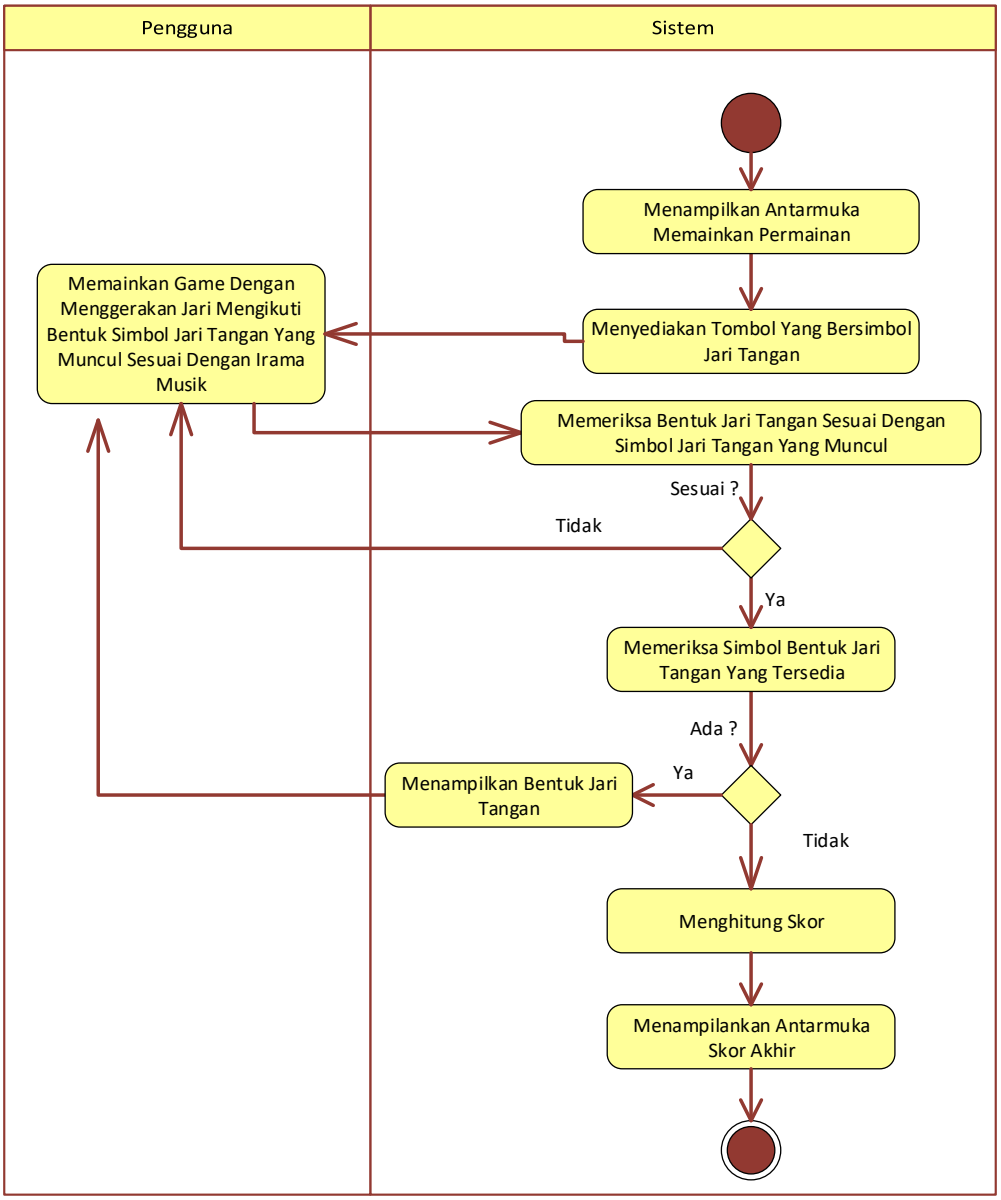
Gambar 3.42 *Activity* diagram Menggerakan Kursor dengan Gerakan Jari Tangan

Keterangan :

1. Sistem menampilkan halaman integrasi jari tangan.
2. Sistemakan mengaktifkan kamera.
3. Pengguna menggerakan jari telunjuk ke berbagai arah.
4. Mendeteksi posisi jari tangan pengguna.
5. Jika tidak terdekteksi maka akan mengulang proses pendekteksi poisisi jari tangan.
6. Jika jari tangan berhasil melacak gerakan jari tangan maka akan melacak gerakan jari tangan pengguna.
7. Sistem akan menampilkan gerakan kusor pada halaman integrasi jari tangan.

8. Pengguna dapat melakukan klik tombol selesai dengan membentuk kepalan tangan
9. Sistem akan mendeteksi bentuk kepalan tangan yang di inputkan.
10. Jika tidak terdeteksi maka proses klik tombol akan diulang kembali oleh pengguna..
11. Jika iya maka proses klik berhasil dan menyelesaikan halaman integrasi jari tangan.

4. Berikut ini adalah gambar 3.43 aktivity diagram memainkan *game*.



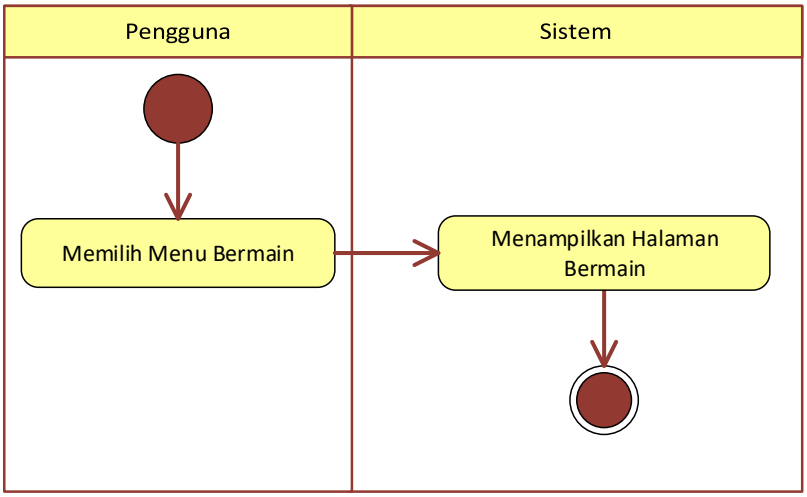
Gambar 3.43 *Activity* diagram Memainkan *Game*

Keterangan :

1. Sistem menampilkan halaman memainkan permainan
2. Sistem menyediakan tombol yang bersimbol bentuk jari tangan
3. Untuk dapat mamainkan *game* pengguna diharuskan menggerakan jari mengikuti simbol bentuk jari tangan yang yang muncul sesuai dengan irama musik yang mengiringi permainan.

4. Sistem memeriksa bentuk jari tangan yang diperagakan oleh pengguna apakah sudah sesuai dengan simbol bentuk jari tangan yang muncul di halaman permainan
5. Setelah melakukan pengecekan maka masuk ke tahap memeriksa apakah masih ada simbol bentuk tangan yang akan muncul berikutnya
6. Jika masih ada maka akan kembali menampilkan simbol bentuk jari tangan untuk di ikuti oleh pengguna
7. Sistem akan menghitung skor akhir dari permainan
8. Sistem akan menampilkan halaman skor.

5. Berikut ini adalah gambar 3.44 aktivity diagram memilih menu bermain.

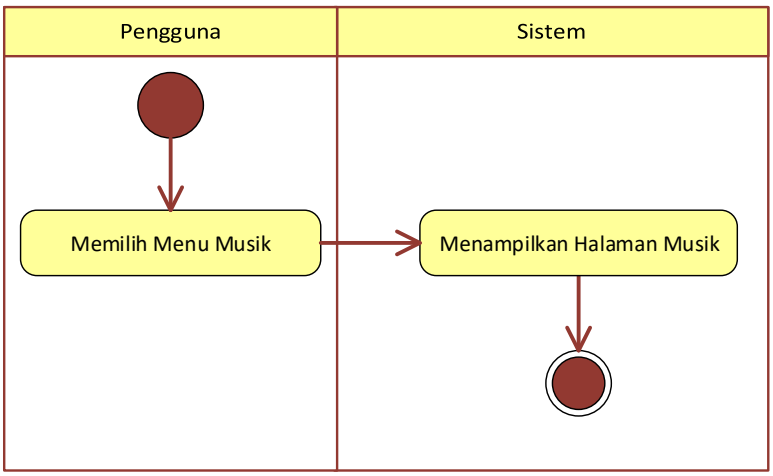


Gambar 3.44 *Activity* diagram Memilih Menu Bermain

Keterangan :

1. Pengguna memilih menu bermain untuk memulai permanian.
2. Sistem menampilkan halaman bermain yang berisi pilihan menu musik.

6 Berikut ini adalah gambar 3.45 aktivity diagram memilih menu musik.

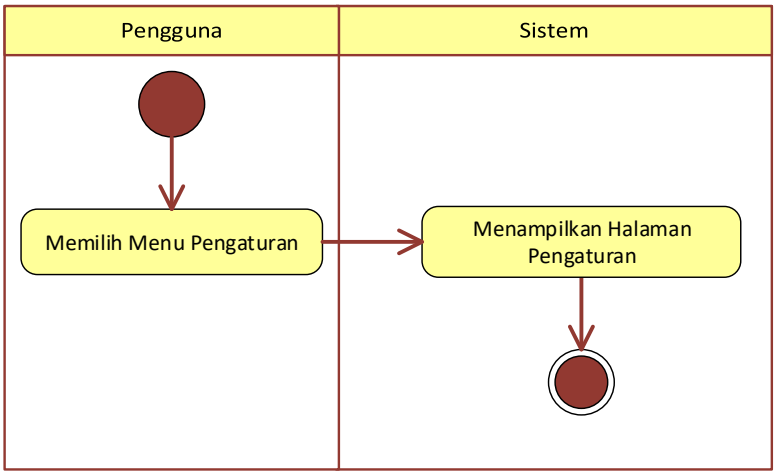


Gambar 3.45 *Activity* diagram Memilih Menu Musik

Keterangan :

1. Pengguna memilih menu musik untuk memilih musik dan level yang diinginkan.
2. Sistem menampilkan halaman musik yang berisi daftar musik dan level yang dapat dipilih oleh pengguna.

7 Berikut ini adalah gambar 3.46 *aktivity* diagram memilih menu pengaturan.

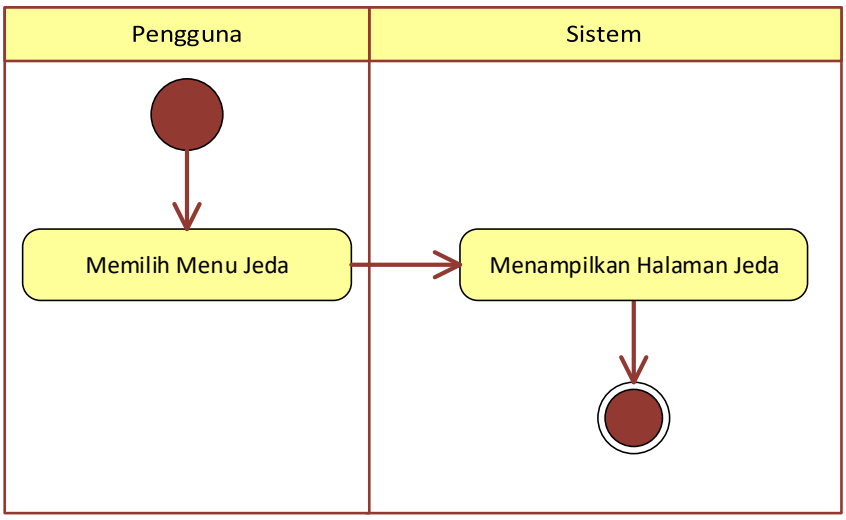


Gambar 3.46 *Activity diagram* Memilih Menu Pengaturan

Keterangan :

1. Pengguna memilih menu pengaturan untuk mengatur suara dan effek suara.
2. Sistem menampilkan halaman pengaturan yang berisi pengaturan suara dan effek suara yang dapat di atur oleh pengguna.

8 Berikut ini adalah gambar 3.47 *aktivity* diagram memilih menu jedah.

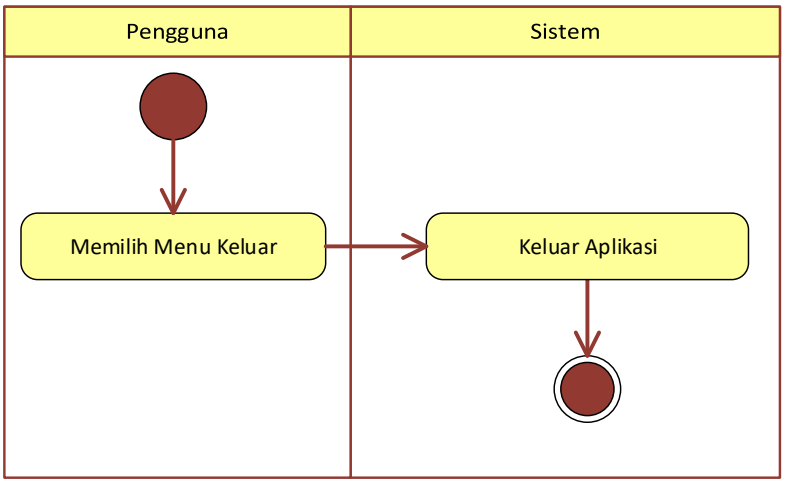


Gambar 3.47 *Activity* diagram Memilih Menu Jedah

Keterangan :

1. Pengguna memilih menu jedah untuk menghentikan sementara musik yang terdapat di dalam memainkan permainan.
2. Sistem menampilkan halaman jedah yang berisi halaman pengaturan.

9 Berikut ini adalah gambar 3.48 aktivity diagram memilih menu keluar.



Gambar 3.48 *Activity* diagram Memilih Menu Keluar

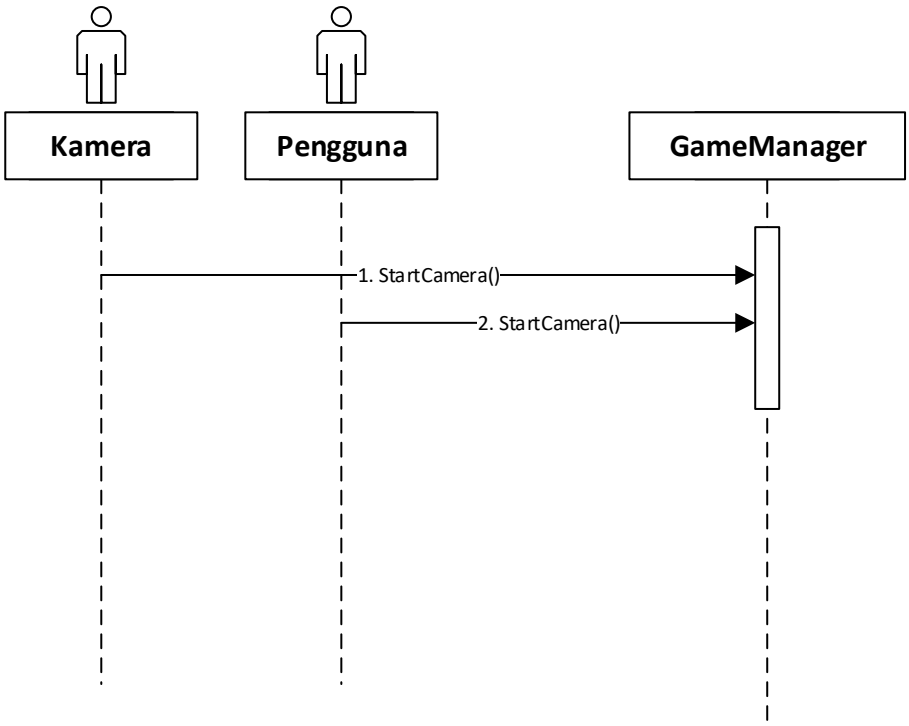
Keterangan :

1. Pengguna memilih menu keluar untuk keluar dari *game* dance touch
2. Sistem menutup *game* dance touch.

3.2.6 *Sequence* Diagram

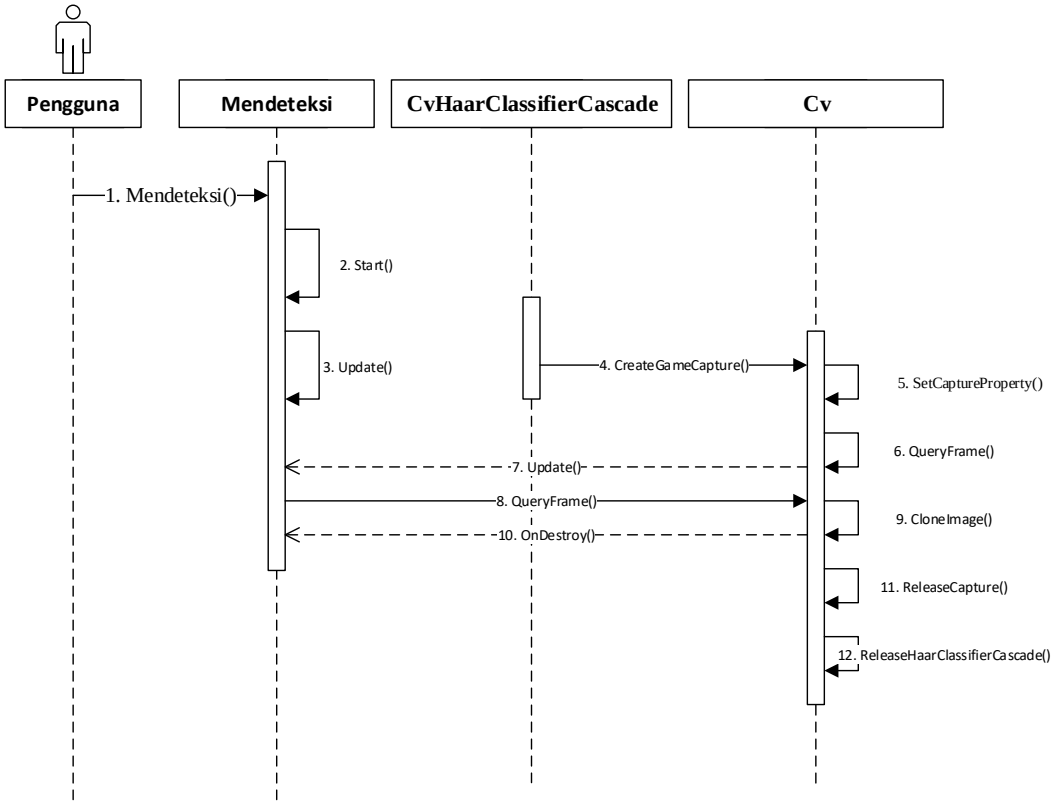
Sequence diagram merupakan diagram yang menggambarkan interaksi antara relasi kelas pada objek *use case*, dengan mendeskripsikan *life cycle* objek yang dikirimkan dan diterima objek. Berikut adalah gambaran *sequence* diagram pada *game* dance touch terbagi menjadi beberapa bagian berikut diantaranya :

1 Berikut ini adalah gambar 3.49 *sequence* diagram menangkap citra jari tangan.



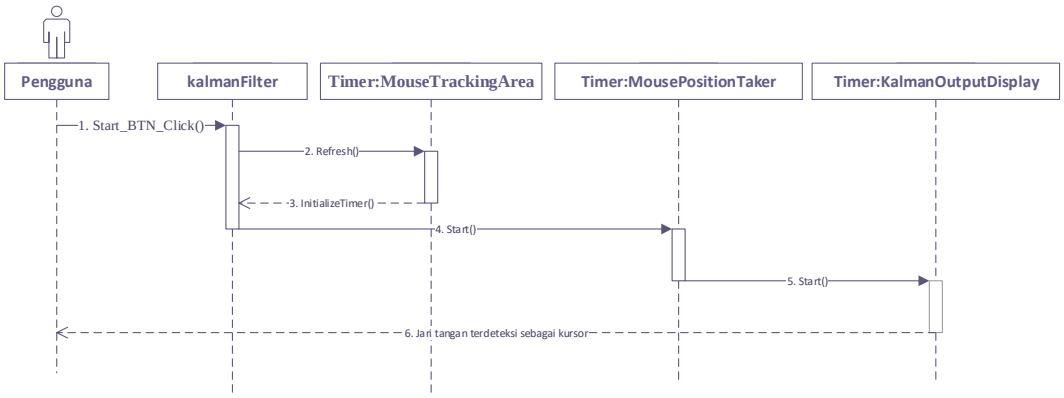
Gambar 3.49 *Sequence* diagram menangkap citra jari tangan

2 Berikut ini adalah gambar 3.50 *sequence* diagram mendeteksi bentuk simbol jari tangan.



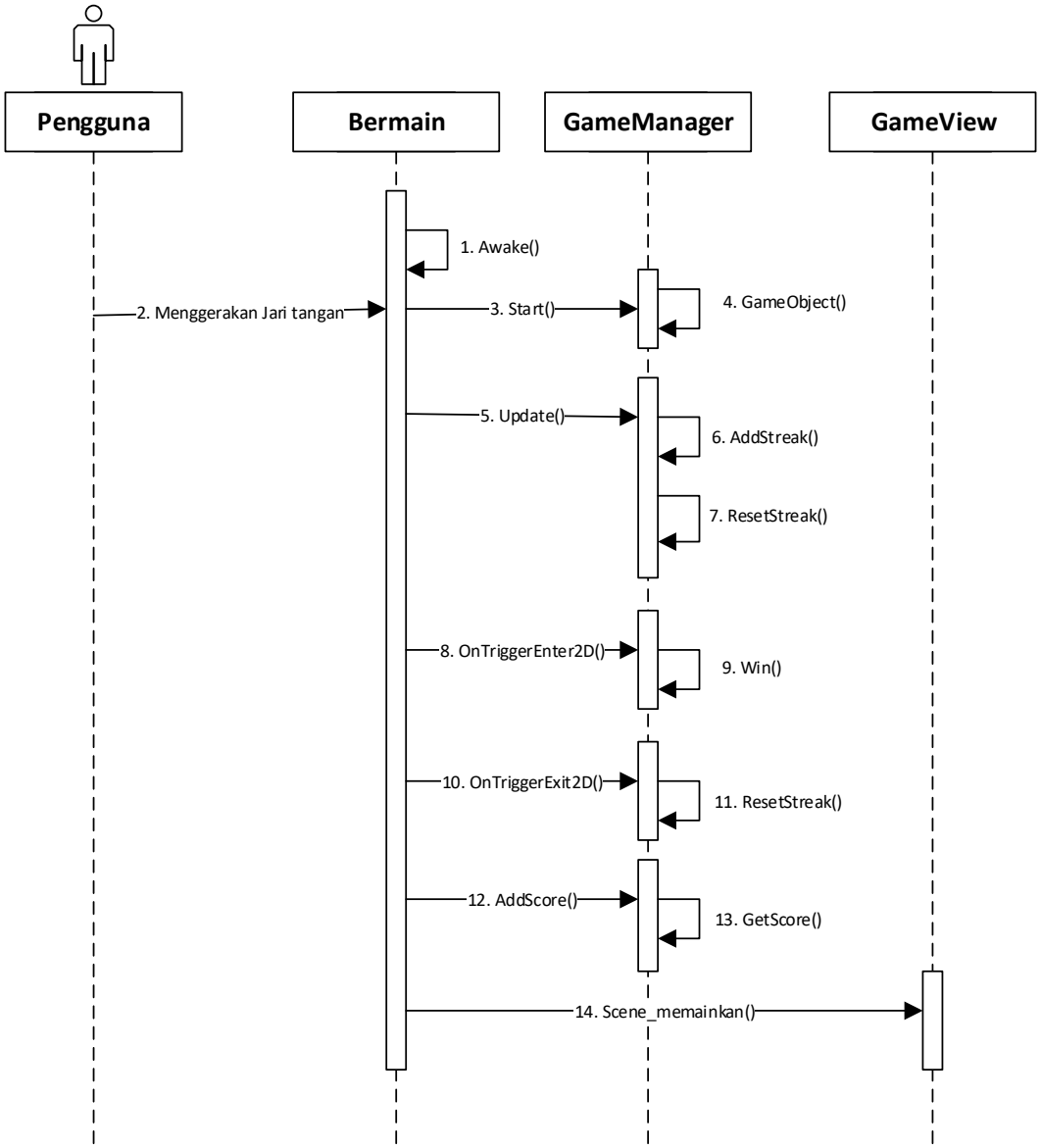
Gambar 3.50 *Sequence* Diagram Mendeteksi Citra Jari Tangan

3 Berikut ini adalah gambar 3.51 *sequence* diagram menggerakan kursor dengan jari tangan.



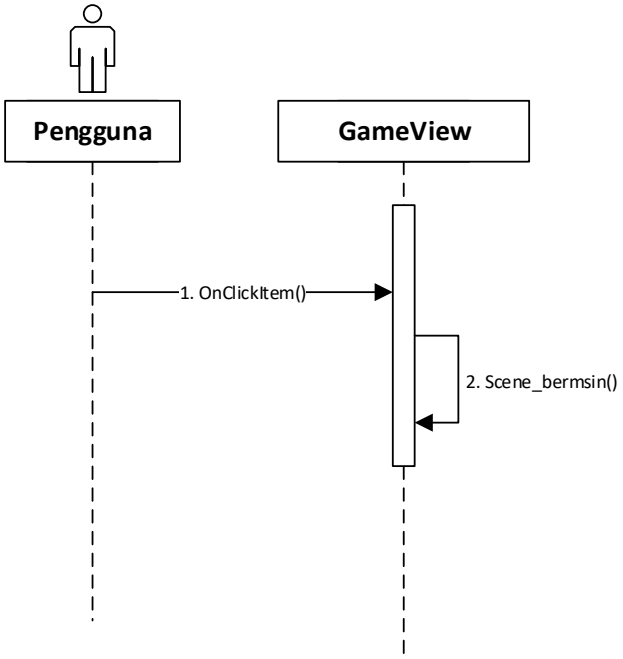
Gambar 3.51 *Sequence* diagram menggerakan kursor dengan jari tangan

4 Berikut ini adalah gambar 3.52 *sequence* diagram memainkan permainan.



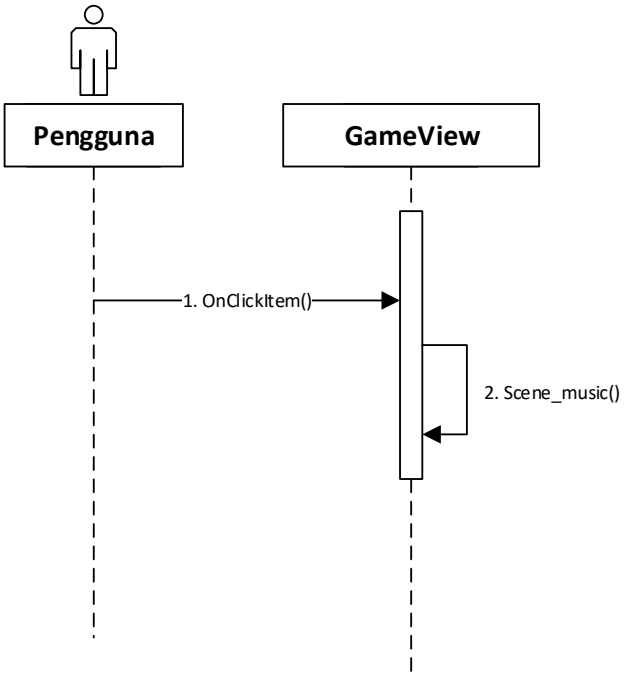
Gambar 3.52 *Sequence* Diagram Memainkan Permainan

5 Berikut ini adalah gambar 3.53 *sequence* diagram memilih menu bermain .



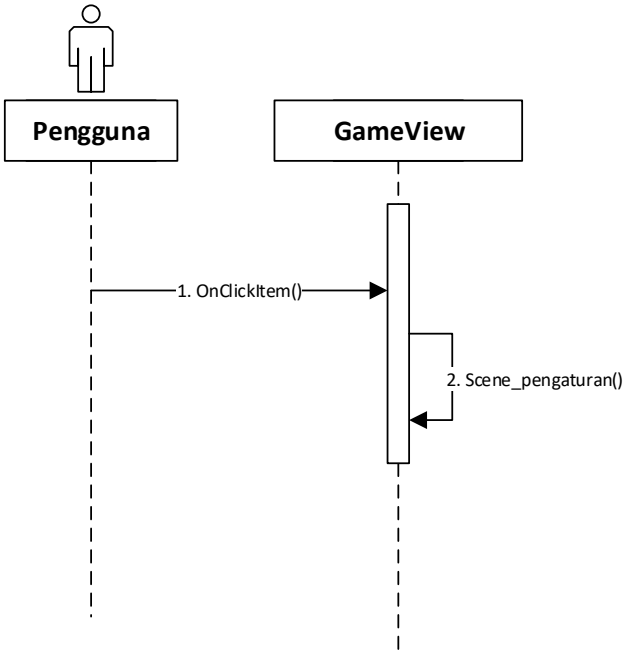
Gambar 3.53 Sequence Diagram Memilih Menu Bermain

6 Berikut ini adalah gambar 3.54 *sequence* diagram memilih menu musik .



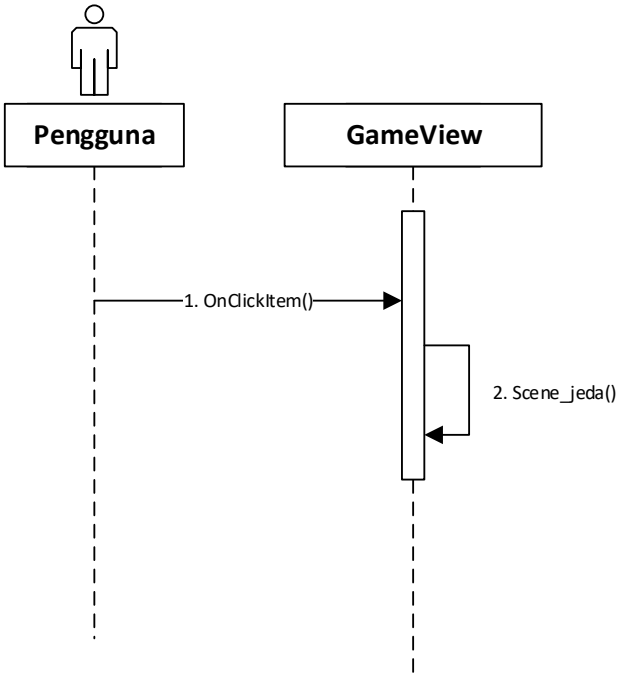
Gambar 3.54 *Sequence* Diagram Memilih Menu Musik

7 Berikut ini adalah gambar 3.55 *sequence* diagram memilih menu Pengaturan .



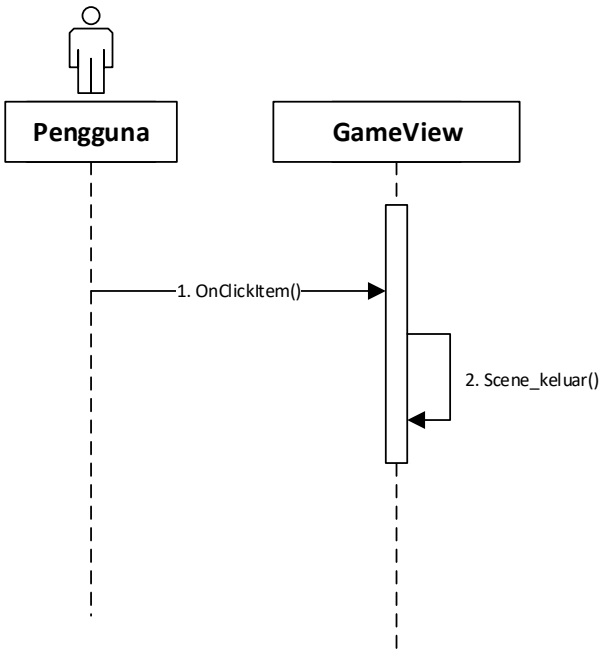
Gambar 3.55 *Sequence* Diagram Memilih Menu Pengaturan

8 Berikut ini adalah gambar 3.56 *sequence* diagram memilih menu jedah .



Gambar 3.56 *Sequence* Diagram Memilih Menu Jedah

9 Berikut ini adalah gambar 3.57 *sequence* diagram memilih menu keluar .

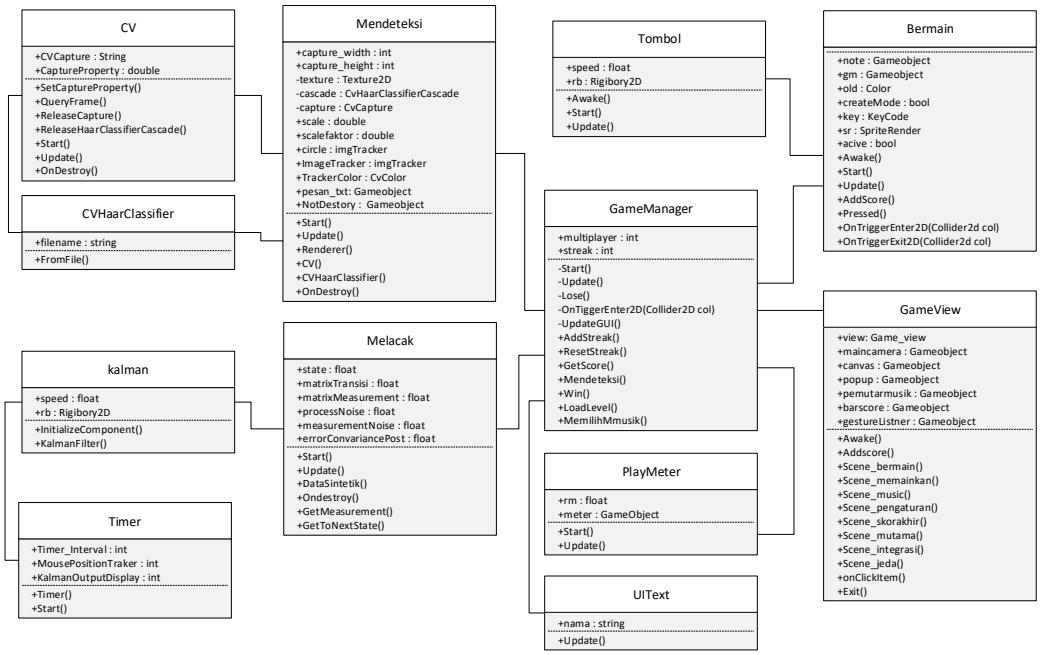


Gambar 3.57 *Sequence* Diagram Memilih Menu Keluar

3.2.7 Class Diagram

Class diagram menggambarkan struktur dan hubungan antar objek-objek yang akan dibuat untuk membangun sistem. Kelas masing-masing memiliki atribut dan metode atau operasi. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas. Metode atau operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Berikut adalah gambar 3.58 *class* diagram dalam *game* dance touch :



Gambar 3.58 *Class* Diagram

3.2.8 Prancangan Sistem

Perancangan sistem untuk menspesifikasi aspek-aspek teknik yang menjadi solusi dalam perencanaan. Pada tahap ini perancangan akan didefinisikan secara detail untuk mengatasi masalah-masalah yang lebih teknis, berkaitan dengan kegiatan implementasi seperti perancangan antarmuka.

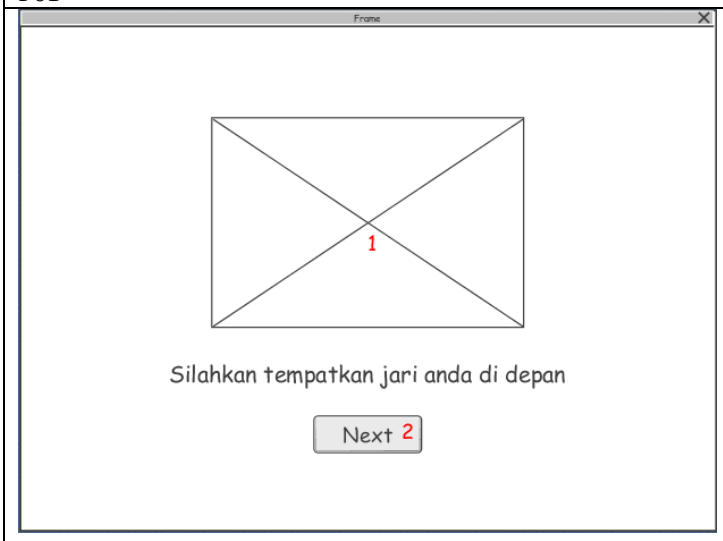
3.2.8.1 Perancangan Antarmuka Sistem

Tahap perancangan antarmuka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga akan mempermudah dalam mengimplementasikan aplikasi serta akan memudahkan pembuatan aplikasi. Berikut ini gambaran perancangan antarmuka yang ada dari aplikasi yang akan dibangun yaitu:

a. Perancangan antarmuka halaman pengenal jari tangan

Berikut adalah perancangan antarmuka halaman pengenal jari tangan.

F01



Keterangan:
1. Tampilan input melalui kamera
2. Tombol Next

Fungsi :
1. Tampilan input akan mangkap objek jari tangan dan mencocokan dengan data positif ataupun negatif.
2. Klik tombol next akan menampilkan F02.

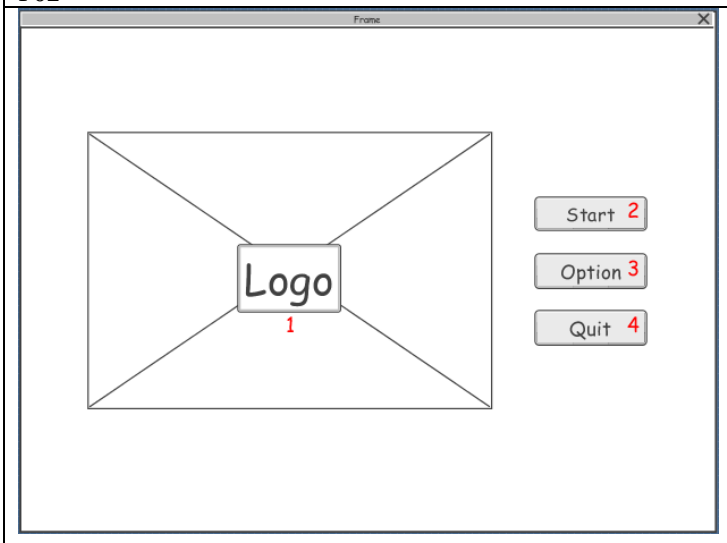
Font : Comic Sans
Orientasi : Landscape

Gambar 3.59 Tampilan halaman pengenal jari tangan

b. Perancangan antarmuka halaman menu utama

Berikut adalah perancangan antarmuka halaman menu utama.

F02



Keterangan:
1. Tampilan Logo Aplikasi
2. Tombol Start
3. Tombol Option
4. Tombol Quit

Fungsi :
1. Klik tombol *start* akan menampilkan F03.
2. Klik tombol *option* akan menampilkan F04.
3. Klik tombol *quit* akan keluar dari aplikasi.

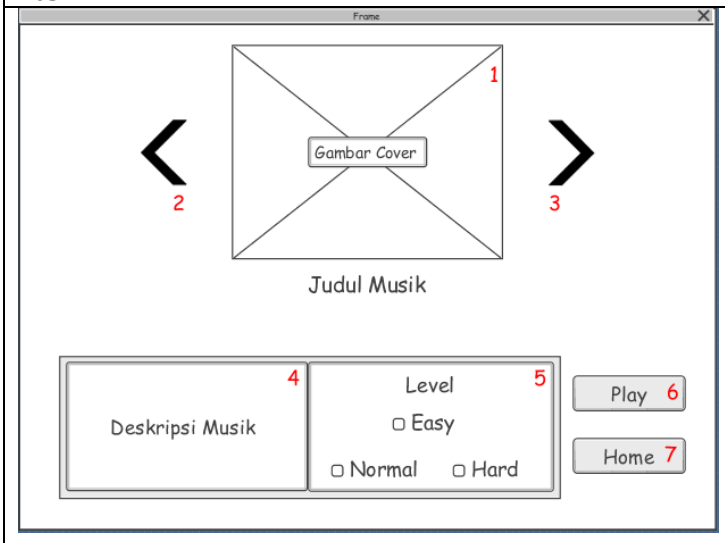
Font : Comic Sans
Orientasi : Landscape

Gambar 3.60 Tampilan Halaman Menu Utama

c. Perancangan Antarmuka Halaman Menu Musik

Berikut adalah perancangan antarmuka halaman menu musik.

F03



Font : Comic Sans
Orientasi : Landscape

Keterangan:
1. Tampilan Gambar Cover musik
2. Tombol Geser ke kiri
3. Tombol Geser ke kanan
4. Tampilan deskripsi musik
5. Tampilan pilih Level
6. Tombol Play
7. Tombol Home

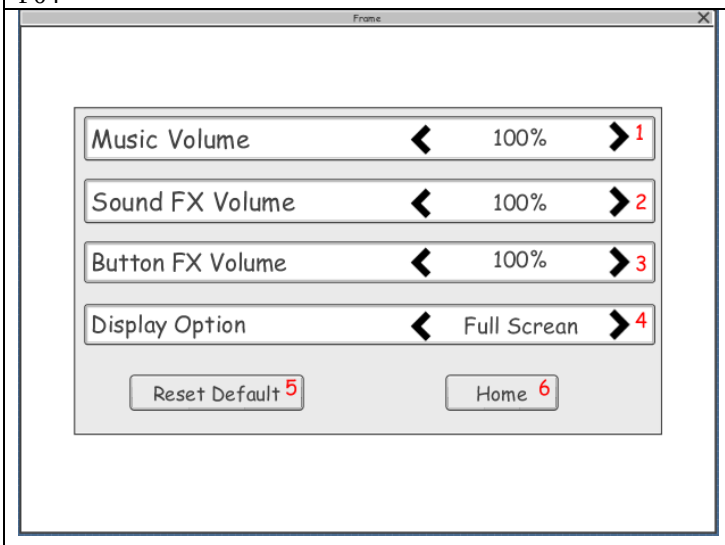
Fungsi :
1. Klik geser kiri dan geser kanan untuk memilih musik.
2. Klik cek box level untuk menentukan tingkat kesulitan bermain *game*.
3. Klik tombol *play* akan menampilkan F05.
4. Klik tombol *home* akan menampilkan F02.

Gambar 3.61 Tampilan Halaman Menu Musik

d. Perancangan Antarmuka Halaman Menu Pengaturan

Berikut adalah perancangan antarmuka halaman menu pengaturan.

F04



Font : Comic Sans
Orientasi : Landscape

Keterangan:
1. Tombol Up-down Music Volume
2. Tombol Up-down Sound FX Volume
3. Tombol Up-down Button FX Volume
4. Tombol Display Option
5. Tombol Reset Default
6. Tombol Home

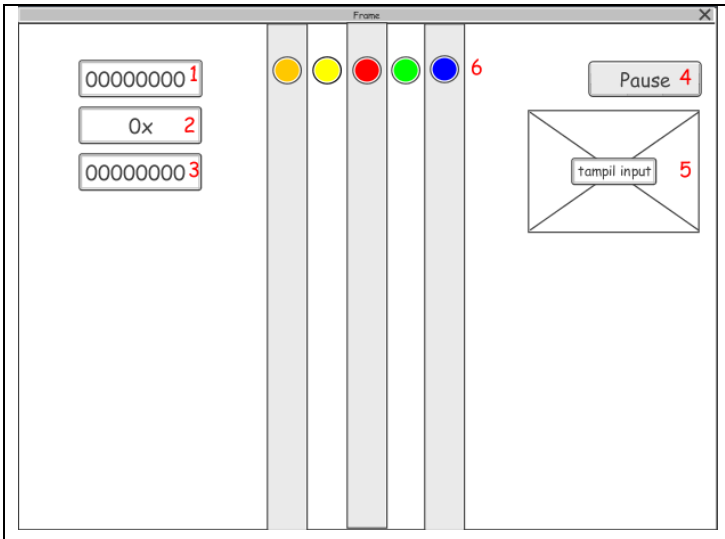
Fungsi :
1. Klik tombol up-down music volume untuk mengatur volume musik
2. Klik tombol up-down sound fx volume untuk mengatur volume sound.
3. Klik tombol up-down button fx volume untuk mengatur volume button.
4. Klik tombol up-down display option untuk menatur resolusi aplikasi.
5. Klik tombol *reset default* untuk mereset pengaturan option ke pengaturan awal.
6. Klik tombol *home* akan menampilkan F02.

Gambar 3.62 Tampilan Halaman Menu Pengaturan

e. Perancangan Antarmuka Halaman Bermain *Game*

Berikut adalah perancangan antarmuka halaman Bermain *game*.

F05



Font : Comic Sans
Orientasi : Landscape

Keterangan:
1. Tampilan Score
2. Tombol Combo
3. Tombol Streak
4. Tombol Pause
5. Tampilan video input data
6. Tombol berwarna untuk memainkan aplikasi

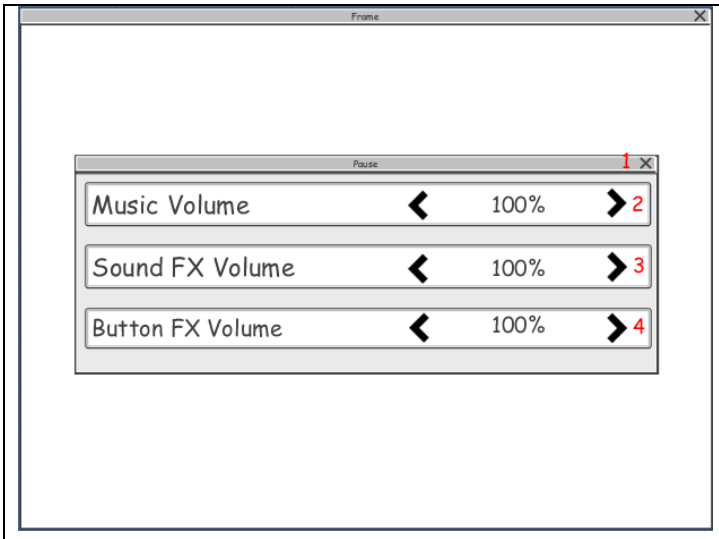
Fungsi :
1. Tombol berwarna akan dikendalikan melalui gerakan jari tangan dan pola jari tangan yang di terima dari video input.
2. Tombol akan ter-klik jika objek jari tangan sesuai dengan gambar pola yang terdapat pada setiap warna. Setiap tombol berwarna memiliki bentuk pola yang berbeda-beda.
3. Klik tombol *pause* akan menampilkan F06.

Gambar 3.63 Tampilan Halaman Bermain *Game*

f. Perancangan Antarmuka Halaman Menu Jedahh

Berikut adalah perancangan antarmuka halaman menu jedahh.

F06



Font : Comic Sans
Orientasi : Landscape

Keterangan:
1. Tombol Keluar
2. Tombol Up-down Music Volume
3. Tombol Up-down Sound FX Volume
4. Tombol Up-down Button FX Volume

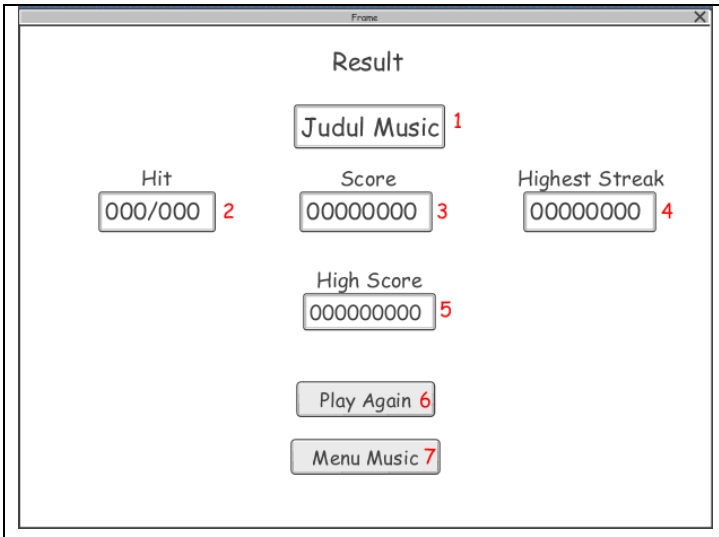
Fungsi :
1. Klik tombol keluar akan kembali ke F05.
2. Klik tombol up-down music volume untuk mengatur volume musik
3. Klik tombol up-down sound fx volume untuk mengatur volume sound.
4. Klik tombol up-down button fx volume untuk mengatur volume button.

Gambar 3.64 Tampilan Halaman Menu Jedahh

g. Perancangan Antarmuka Halaman Menu Skor Akhir

Berikut adalah perancangan antarmuka halaman menu skor akhir.

F07



Font : Comic Sans
Orientasi : Landscape

Keterangan:
1. Tampilan Judul Musik
2. Tampilan Hit Score
3. Tampilan Score
4. Tampilan Highest Streak
5. Tampilan High Score
6. Tombol Play Again
7. Tombol Menu Music

Fungsi :
1. Klik tombol up-down music volume untuk mengatur volume musik
2. Klik tombol up-down sound fx volume untuk mengatur volume sound.
3. Klik tombol up-down button fx volume untuk mengatur volume button.
4. Klik tombol *play again* akan kembali ke F05.
5. Klik tombol *menu music* akan kembali ke F03.

Gambar 3.65 Tampilan Halaman Menu Skor Akhir