

BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisa Masalah

Pada bagian analisis sistem ini akan dibahas mengenai menganalisa kebutuhan pengguna dari aplikasi *game* yang akan dibangun. Dimana pada perancangan menentukan *player* yang akan digunakan, dimana pada *game* ini menceritakan tentang pertandingan satu lawan satu. Tujuan dari pertarungan dari karakter antar pemain adalah membuat musuh kalah (*KO*), untuk menghabiskan nyawa dengan batas nyawa yang diberikan untuk memenangkan pertandingan.

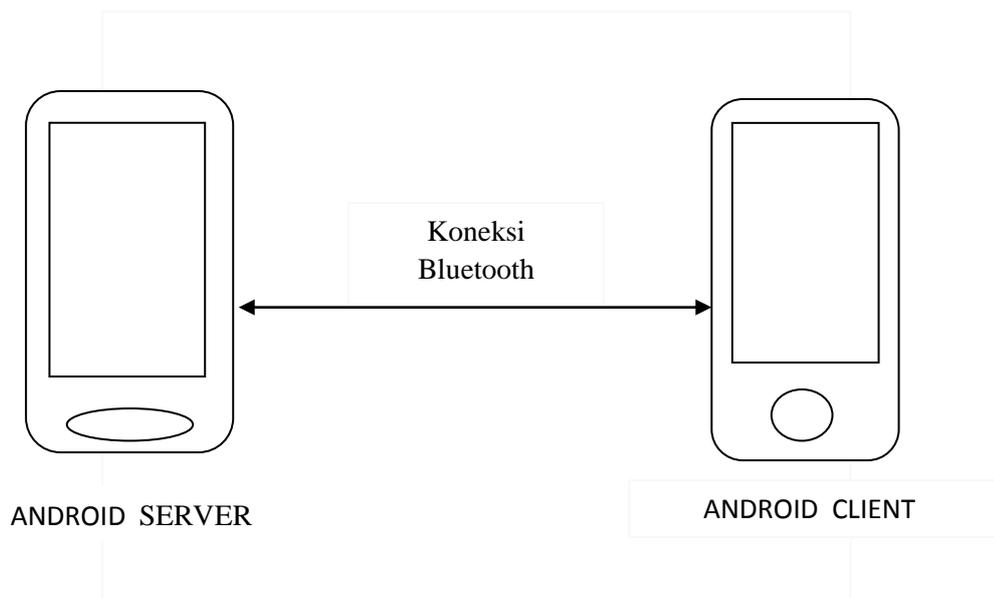
Pertandingan antar pemain memanfaatkan koneksi jaringan *Bluetooth* sehingga para pemain dapat bermain pada perangkat *mobile phone* berbeda. Fokus pertandingan *game* yang dirancang adalah 2 (dua) orang pemain yang bermain pada perangkat *mobile phone* yang berbeda dan mengendalikan karakter untuk memenangkan pertandingan.

Berikut ini akan dilakukan analisa konsep komunikasi data agar permainan dapat berjalan pada *game Crazy pong* yang dirancang.

1. Koneksi jaringan *bluetooth*, berfungsi agar masing-masing perangkat *mobile phone* dapat saling terhubung, sehingga para pemain dapat bermain dalam pertandingan.
2. Berjalan secara *client-server*, dengan menempatkan salah satu pemain sebagai aplikasi *server* dan yang lainnya sebagai *client*.

3. Grafis 2D, setelah terhubung layar permainan dengan grafis 2D akan ditampilkan, untuk pergerakan dan aktivitas masing-masing pengguna di sesuaikan secara *streaming* melalui perintah dari aplikasi.

Proses kerja dari aplikasi yang di bangun ini berbasis *client-sever* dengan salah satu perangkat sebagai *server* dan perangkat lain sebagai *client*, dapat di lihat pada gambar III.1. berikut:



Gambar III.1. Skema Jaringan Pada Permainan

Dari gambar diatas terlihat bahwa salah satu perangkat akan bertugas sebagai perangkat *android server* dan *android* lain *client*, akses data *android server* dan *android client* yang saling terhubung. Dalam perancangan ini Pengguna dapat saling terhubung memakai jaringan *bluetooth*. untuk memainkan permainan *game* yang di rancang dengan menghubungkan nama *bluetooth* yang terdeteksi ke jaringan *server*. Adapun aturan permainan pada perancangan aplikasi ini dapat di jelaskan sebagai berikut:

1. Pemain dapat memilih bermain dengan 1 pemain ataupun 2 pemain dari sebelum memulai permainan.
2. Setelah memilih maka selanjutnya pemain akan masuk ke arena pertandingan dan dapat memulai pertandingan dengan lawan permainan
3. Permainan akan tambah sulit, dengan bertambahnya bola yang disesuaikan dari pengaturan bola.
4. Setiap gerakan yang dilakukan menggunakan *control* dari perangkat *android*.
5. Permainan berakhir ketika salah satu pemain berhasil memasukan bola pada lawan sesuai banyak nyawa dan membuat papan penangkis menjadi kecil sebanyak nyawa yang disesuaikan, maka pemain tersebut menang.

III.1.1. Spesifikasi Perangkat

Ada beberapa perangkat yang penulis gunakan agar aplikasi berjalan sebagaimana mestinya, yaitu sebagai berikut :

1. Perangkat Lunak (*Software*)
 - a. *Operating System*, OS yang digunakan dalam perancangan dan tes untuk program aplikasi yang dirancang adalah *Windows 7, 32 bit*.
 - b. *JDK Java 1.7*, sebagai bahasa program dan *compilerJava*.
 - c. *Eclipse Luna*, sebagai *editor source code Java*.
 - d. Serta jaringan *Bluetooth*.
 - e. *ANDROID SDK 4.0*.
 - f. *ADT Plugin 18*.

2. Perangkat Keras (*Hardware*)

- a. *Laptop* spesifikasi setara *core i3*
- b. *Mouse, keyboard, dan Monitor.*
- c. *Mobile phone Android* dengan spesifikasi *ram* minimal 512Mb.

III.1.2. Konfigurasi Jaringan

Agar dapat terhubung pada masing-masing perangkat *mobile phone android* untuk dapat memainkan aplikasi *game*, pada perancangan ini memanfaatkan jaringan *bluetooth*, instalasi dan konfigurasi *server* dan *client* sangatlah penting diketahui agar penerapan *streaming* data dapat berjalan. Adapun hal-hal yang harus disesuaikan adalah sebagai berikut:

1. *Port* akses untuk komunikasi data antar *client* dan *server*, penentuan *port* berapa yang digunakan pada aplikasi. Dan *setting* izin pada sebuah *port* menghindari penolakan dari *firewall operating system* saat *client* mengakses *android server*. Pada system *android* sebenarnya ada dua jenis *Port* yaitu :
 - a. *Port* fisik, adalah soket atau slot atau colokan yang ada di *android* sebagai penghubung peralatan *input-output*.
 - b. *Port* Logika (*non* fisik), adalah *port* yang di gunakan oleh *Software* sebagai jalur untuk melakukan koneksi dengan *android* lain, tentunya termasuk koneksi *bluetooth*.

Dalam perancangan ini jelas bahwa *port* yang digunakan adalah *port* logika yang bertujuan menghubungkan aplikasi *server* dan *client* agar dapat saling berkomunikasi, pengalamatan *port* berupa nilai angka sampai dengan 65535.

2. Konfigurasi jaringan *bluetooth*, hal yang sering mempengaruhi dari aktivitas komunikasi data adalah sebuah jaringan yang tidak stabil sehingga pengiriman maupun penerimaan data sering terputus yang mengakibatkan *request time out* atau gagal *download* dan *upload* data pada proses *streaming*.

III.1.3. Komunikasi Antar Player

Rancangan komunikasi antar pemain atau objek karakter dijumpai oleh aplikasi *server*, sebuah perintah ditentukan dalam perancangan ini perintah ditentukan berbentuk *string*, dan dikirim ke *android* pemain lain, *android* penerima akan mengeksekusi perintah dan disesuaikan untuk mengenali perintah apa yang akan ditampilkan pada objek layar, seperti gerak ke kiri dan ke kanan. Konsep komunikasi data yang digunakan berjalan setelah masing-masing android saling terhubung pada *server*. Setelah terhubung pemain baru dapat mengendalikan objek karakter pada permainan. Tidak ada eksekusi lokal *android* pada *game* ini. Sehingga pemain baru dapat menjalankan objek karakter setelah terhubung pada aplikasi *server*.

III.2. Teknik Pemecahan Masalah

Dalam proses perancangan aplikasi *game android* yang akan dibangun, tahapan-tahapan analisis sistem yang dilakukan penulis adalah sebagai berikut:

1. Mempelajari dan menganalisis bagaimana penyesuaian *streaming* data dapat berjalan pada sebuah perangkat android dan melihat dari cara penggunaan dan fungsinya pada bahasa pemrograman yang digunakan.

2. Menentukan kebutuhan pengguna dan tingkat kesulitan perancangan dengan menyimpulkan kontrol apa saja yang dapat diakses.
3. Menentukan *hardware* dan *software* yang dibutuhkan dari perangkat sebagai pendukung perancangan.
4. Perancangan aplikasi, setelah memahami kebutuhan dan tujuan dari perancangan maka dilakukan pembuatan dan pembangun sistem, aturan yang ditentukan, dan perintah yang direncanakan.
5. Pengujian dan *fix error*, melakukan pengujian terhadap aplikasi yang telah selesai dengan menganalisis target yang dicapai serta kekurangan aplikasi untuk diperbaharui agar dapat berjalan baik.

III.3. Perancangan Sistem

Pada perancangan aplikasi menjelaskan mengenai rancangan yang akan dikerjakan serta fitur-fitur yang akan dipakai pada aplikasi tersebut seperti algoritma rancangan, rancangan layar, dan lainnya. Dimana objek dari perancangan ini difokuskan pada akses *grafis* 2D. Untuk aplikasi pada masing-masing android, *interface* yang tampil merupakan aplikasi *desktop* yang berektensi *executejava* yakni **.jar. file *.jar* merupakan suatu *class java* yang telah *dipacking* sehingga bisa dibaca oleh mesin penggerak *Java Development Kit* Untuk menggambarkan proses sistem penulis mendeskripsikan dengan menggunakan beberapa diagram, yaitu sebagai berikut.

III.3.1. Use Case Diagram

Use case diagram berfungsi untuk menggambarkan kegiatan aktor atau pengguna aplikasi. Adapun *use case* diagram aplikasi yang dirancang dapat dilihat pada gambar III.2 berikut.



Gambar III.2. Use Case Diagram

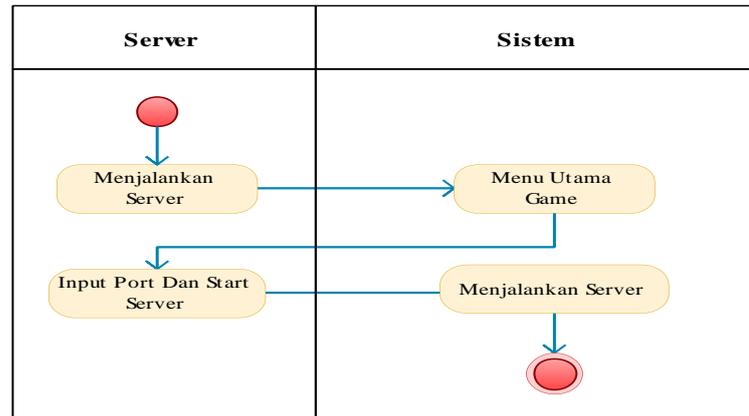
Dari gambar III.2 *use case* diagram diatas dapat dilihat pada masing-masing aktor memiliki peran sendiri dalam jaringan, salah satu menjadi *server* dan lainnya *client*, untuk opsi pilihan dari aplikasi hampir sama seperti pilihan *connect* dan *disconnect*.

III.3.2. ActivityDiagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity* diagram merupakan state diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar *transisi* di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity* diagram tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari *level* atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

1. Activity Diagram Form Server

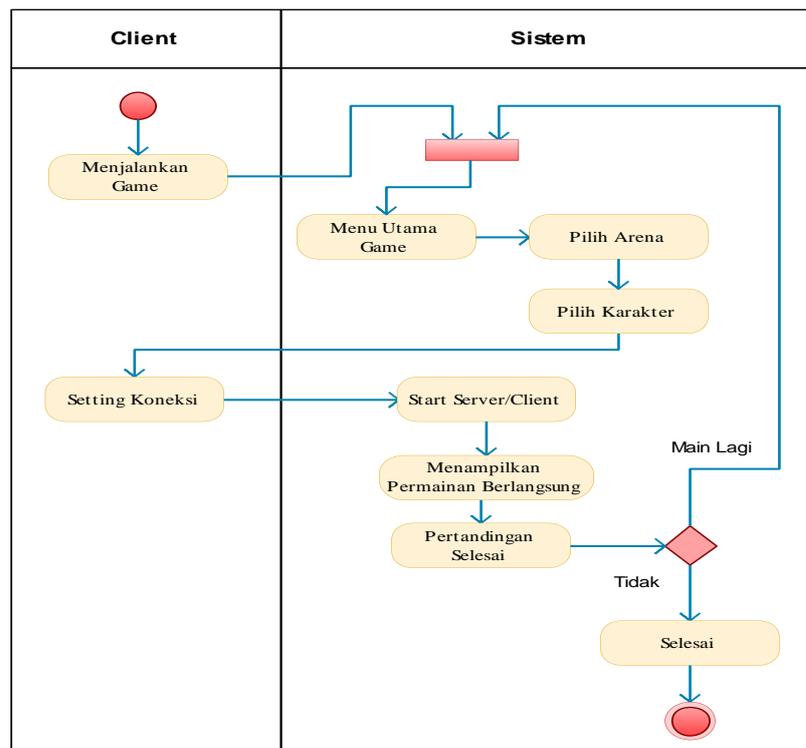
Berikut ini adalah aktivitas pengguna dalam proses menjalankan aplikasi *server*, yang dapat dilihat pada gambar III.3 sebagai berikut.



Gambar III.3. Activity Diagram Form Server

2. Activity Diagram Form Client

Berikut ini adalah proses aktivitas pemain dalam menjalankan aplikasi untuk memulai permainan dengan menyesuaikan *setting* koneksi. Dapat dilihat pada gambar III.4 dibawah ini.

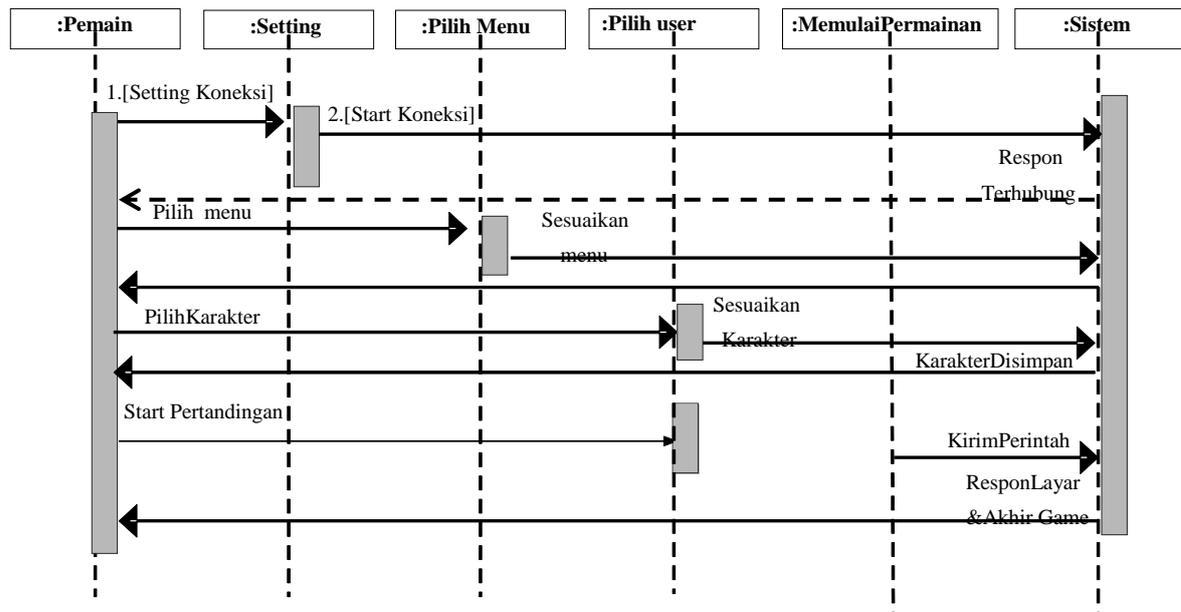


Gambar III.4. Activity Diagram Form Client

Pada gambar diatas dapat dijelaskan bahwa *android client* menjalankan permainan *game* terlebih dahulu, kemudian masuk ke *menu* utama pada *game*, lalu pemain/*player* dapat memilih karakter di dalam *game* tersebut. Kemudian *client* melakukan *setting* koneksi, apabila tidak ada masalah pada *setting* koneksi, maka permainan dapat dimulai pada *server* ataupun *client*, dan menampilkan permainan yang berlangsung pada *game* tersebut. Jika salah satu pemain/*player* kalah, maka permainan akan selesai/berakhir.

III.3.3. Sequence Diagram

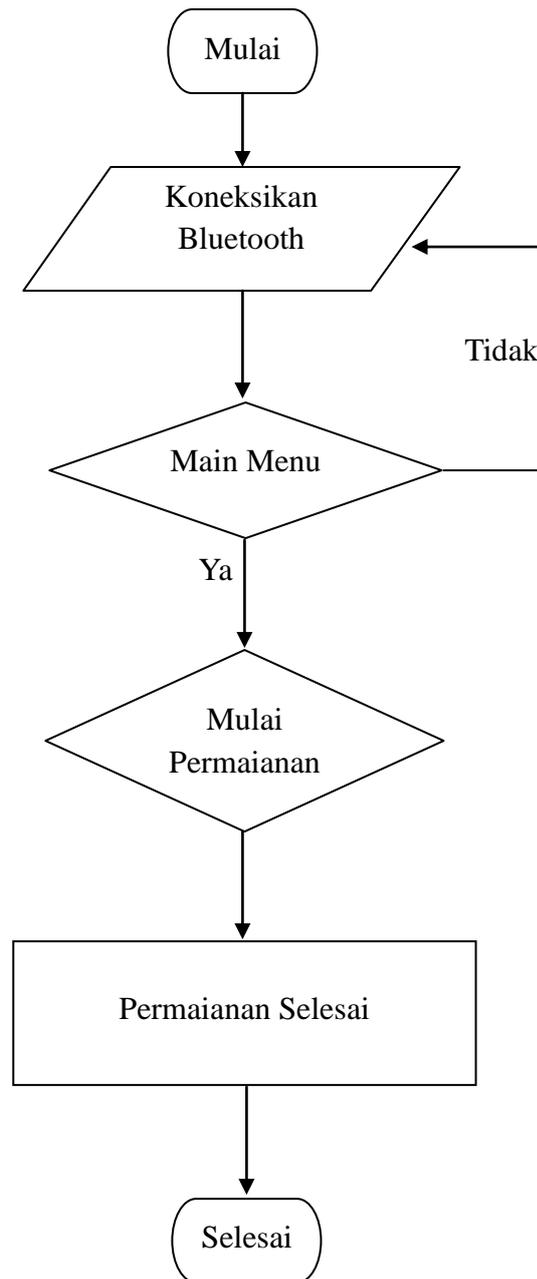
Sequence diagram menggambarkan interaksi antar objek di dalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya). *Sequence* diagram terdiri atas dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait). Pada *Sequence* diagram ini menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan pada sistem sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu dapat dilihat pada gambar III.5 berikut:



Gambar III.5. Sequence Diagram

III.3.4. Flowchart

Flowchart merupakan gambar atau bagian yang memperlihatkan urutan dan hubungan antar proses beserta intruksinya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung. *Flowchart* ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan proses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Jadi urutan dan hubungan antar proses beserta intruksinya pada game *Crazy pong* dapat di lihat pada gambar III.6. berikut:



Gambar III.6. Flowchart Aplikasi Game Crazy Pong

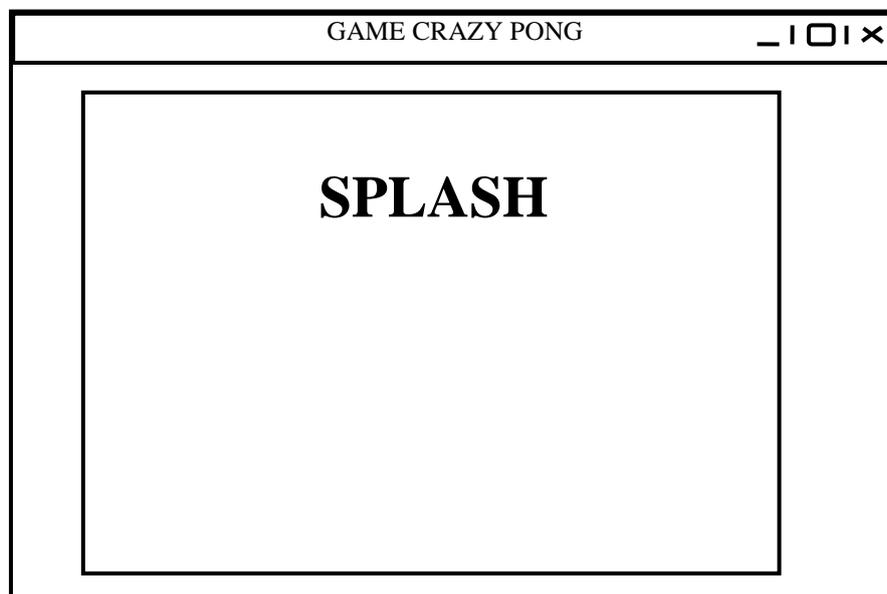
Pada gambar diatas dapat dijelaskan, aplikasi game crazy pong memulai permainan dengan mengkoneksikan *server* dan *client* melalui izin Bluetooth. Memilih menu pilihan , dan memulai permainan, hingga permaina berakhir.

III.4. Perancangan Layar/*Interface*

Aplikasi yang dirancang merupakan aplikasi yang berjalan dengan 2 (dua) aplikasi yang berjalan dengan fungsi sebagai *client* dan *server*. Dalam hal ini bertujuan untuk menerapkan solusi pemecahan masalah yang telah ditentukan pada sistem. Berikut ini adalah perancangan *interface* dari aplikasi yang direncanakan.

1 Rancangan *Form Splash*

Pada *form* ini merupakan tampilan *loading* menuju menu utama, yaitu pada gambar III.7 sebagai berikut:



Gambar III.7. Rancangan *Form Splash*

2 Rancangan *Form Konfigurasi jaringan*

Tampilan layar *form* konfigurasi berfungsi untuk menfaktur jaringan untuk dihubungkan perangkat *server*, dapat dilihat pada gambar III.8. berikut.



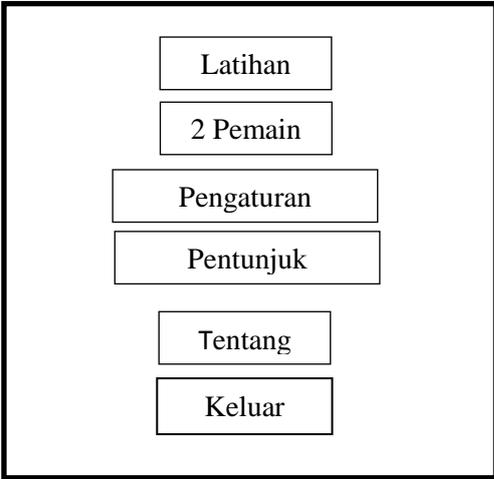
The image shows a rectangular dialog box with a title bar at the top containing the text "Permintaan Izin Bluetooth... ?". Below the title bar is a large empty rectangular area. At the bottom of the dialog box, there are two rounded rectangular buttons. The left button is labeled "Tidak" and the right button is labeled "Ya".

Gambar III.8. Rancangan *Form* Konfigurasi jaringan

Pada gambar diatas terdapat *form* permintaan izin *bluetooth*, yang terdapat button tidak dan *button* Ya. Silakan tekan *button* Ya untuk mengkonfigurasi perangkat *bluetooth*.

3 Rancangan *Form* Menu Utama

Tampilan layar *form* menu utama merupakan tampilan layar permainan untuk pengguna, yang dapat dilihat pada gambar III.9 sebagai berikut.



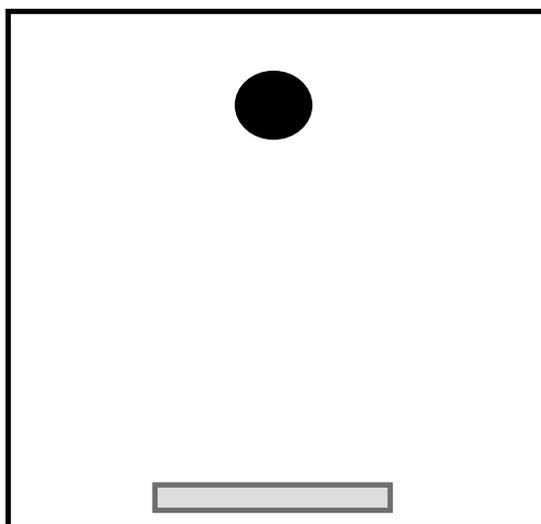
The image shows a rectangular main menu screen. It contains seven rectangular buttons arranged vertically in the center. From top to bottom, the buttons are labeled: "Latihan", "2 Pemain", "Pengaturan", "Pentunjuk", "Tentang", and "Keluar".

Gambar III.9. Rancangan *form* Menu Utama

Pada gambar diatas terdapat rancangan *menu* utama dimana terdapat *button* 1 Pemain , *button* 2 Pemain , *button* Pengaturan, *button* Tentang, dan *button* Keluar.

3. Rancangan Layar Pertandingan

Tampilan layar pertandingan merupakan *form* untuk pengguna pertandingan dalam arena, disini pengguna mengendalikan masing masing karakter yang digunakan untuk mengalahkan pemain lain agar memenangkan permainan. Dapat dilihat pada gambar III.10. sebagai berikut.



Gambar III.10. Rancangan Layar Pertandingan

Pada gambar diatas dapat dijelaskan bahwa layar pertandingan dapat menjadi *player* 1 atau *player* 2 pada *game* pada perangkat *android* tersebut. Jika kedua perangkat sudah terhubung dengan koneksi *bluetooth*.