

BAB III

METODOLOGI

3.1 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem adalah sebuah upaya yang bertujuan untuk memperoleh gambaran rancangan sistem yang akan dibangun mulai dari apa saja proses yang akan dilakukan dalam sistem yang kemudian dianalisis dan disaring untuk mencapai pemahaman dalam pengembangan aplikasi tersebut. Perancangan sistem permainan bentengan ini terdiri sebuah skema papan permainan yang digunakan sebagai area pergerakan *Player* dan *enemy*.

Analisis kebutuhan sistem terdiri dari 6 bagian yaitu analisis kebutuhan fungsional, analisis kebutuhan *input*, analisis kebutuhan *output*, analisis kebutuhan proses, analisis kebutuhan perangkat keras, dan analisis kebutuhan perangkat lunak.

3.1.1 Analisis Kebutuhan Fungsional

Pemain diminta untuk menggerakkan player untuk menyentuh benteng lawan. Bergerak dari titik awal (benteng sendiri) ke benteng lawan tanpa tersentuh *enemy* penjaga benteng. Pemain dapat menggerakkan lebih dari satu player sekaligus. Pemain bebas menggerakkan player ke arah manapun dalam papan permainan. Jika player sebelum menyentuh benteng lawan tersentuh oleh *enemy* yang waktu dari menyentuh bentengnya lebih kecil dari player, maka player akan dijadikan tawanan di penjara. Namun jika salah satu player dapat menyentuh benteng lawan tanpa tersentuh *enemy* atau dapat “menawan” semua *enemy* maka pemain memenangkan permainan.

Terdapat 3 mode permainan, yaitu mode *easy*, *normal*, *hard*. Pada setiap mode mempengaruhi kecepatan gerak dari *enemy* pada arena dari yang pelan hingga cepat.

3.1.2 Analisis Kebutuhan Input

Kebutuhan input atau masukan yang diperlukan untuk memenuhi fungsi – fungsi nyadalam pengembangan permainan bentengan, meliputi :

1. Perintah user untuk menggerakkan pemain *player*.

3.1.3 Analisis Kebutuhan Output

Analisis kebutuhan output dalam pengembangan aplikasi permainan bentengan ini adalah :

1. Pemain Player dan *Enemy*.
Pemain *player* dan *enemy* yang masing – masing terdiri dari 5 pemain. Posisi pemain *player* terdapat pada benteng kanan atas dan posisi pemain *enemy* berada di kiri bawah.
2. Penjara
Terdapat 2 penjara yang masing-masing untuk *player* dan *enemy*. Posisi penjara *player* di atas dan penjara *enemy* berada di layar bawah.
3. *Player* di Penjara
Saat permainan berlangsung dan *player* bertabrakan dengan *enemy* dengan prioritas lebih tinggi, *player* berada penjara milik *enemy* yang posisi berada di bagian bawah layar.
4. *Enemy* di Penjara
Saat permainan berlangsung dan *player* bertabrakan dengan *enemy* dengan prioritas lebih rendah, *enemy* berada di penjara milik *player* yang posisi berada di bagian atas layar.
5. Status Permainan Menang atau Kalah.
Status permainan berupa tulisan “*you win*” untuk kemenangan *player* dan “*you lose*” untuk kemenangan *enemy* muncul menandakan permainan berakhir

3.1.4 Analisis Kebutuhan Proses

Setelah menganalisa kebutuhan *input* pada sistem ini, maka diperoleh proses-proses yang terdapat pada sistem ini diantaranya :

1. Proses pemilihan tingkat kesulitan

Pada proses ini user memilih tingkat kesulitan yang digunakan untuk menentukan kecepatan gerak dari *enemy*.

2. Proses *Play*

Pada proses ini dilakukan saat user memilih tombol *play* untuk memulai permainan dan tampilan dipindahkan ke dalam tampilan arena permainan.

3. Proses Keluar Permainan.

Proses dalam sistem yang digunakan untuk keluar dari aplikasi.

3.1.5 Analisis Kebutuhan Perangkat Keras

selama pengembangan aplikasi bentengan ini menggunakan computer PC dengan spesifikasi sebagai berikut :

1. Processor intel core I3-2100 (4 CPUs) @ 3.10Ghz.
2. Hardisk 1.5 TB.
3. Memory RAM 4 GB
4. Graphic Card AMD RADEON 1 GB

3.1.6 Analisis Kebutuhan Perangkat Lunak

Dalam membangun aplikasi permainan bentengan ini dibutuhkan perangkat lunak sebagai berikut :

1. Sistem Operasi Windows 7 Ultimate 32 Bit
2. Corel Draw X5
3. App Inventor

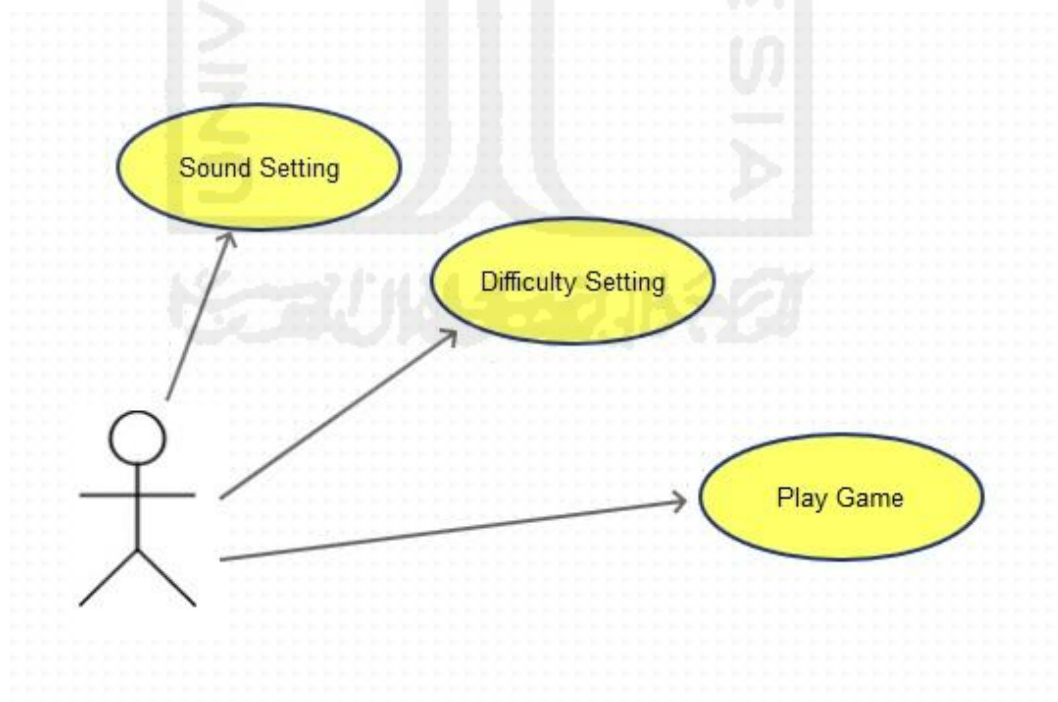
3.2 Perancangan Sistem

Proses selanjutnya setelah analisis kebutuhan sistem adalah proses perancangan sistem. Pada tahap ini sistem dibuat dengan menggunakan hasil dari analisis kebutuhan. Untuk memudah dalam implementasinya maka digunakan perancangan UML (*Unified Modeling Language*). Pada perancangan UML digunakan untuk mendeskripsikan, memodelkan dan mendokumentasikan proses *software development*.

Dalam perancangan sistem ini yang digunakan untuk permodelan sistem dan perancangan tampilan menggunakan *Use Case Diagram*, *Sequence Diagram* dan *Activity Diagram*.

3.2.1 Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamat luar. *Use Case* digunakan untuk menggambarkan fungsionalitas dari sistem yang dikembangkan. Dan juga mempresentasikan interaksi antara pengguna dengan sistem



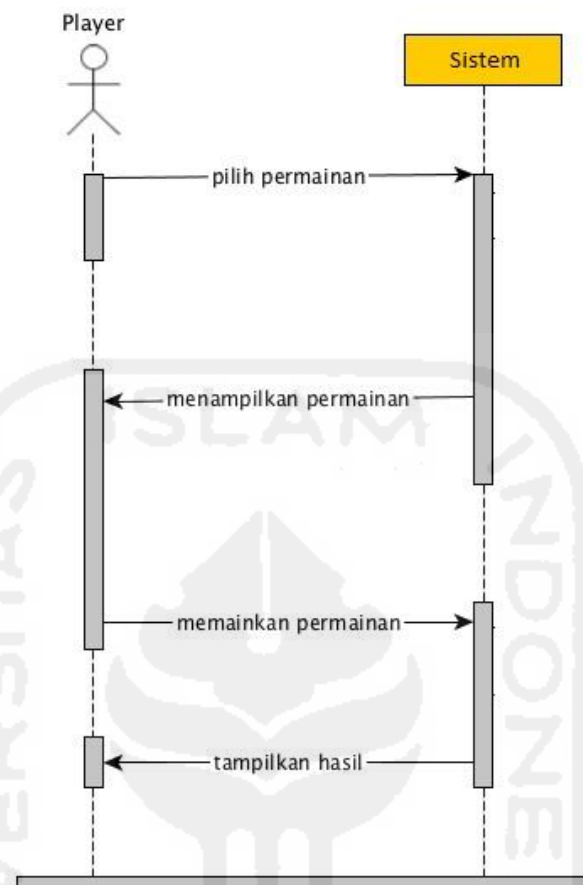
Gambar 3.1 *Use Case Diagram*

Pada gambar 3.1 tentang gambar *use case diagram* menjelaskan tentang *use case* yang digunakan dalam aplikasi game ini. *Sound Setting* merupakan *use case* tentang pengaturan suara untuk aktif atau tidak aktif saat selama aplikasi dijalankan. *Difficulty Setting* merupakan *use case* untuk mengatur tingkat kesulitan yang dimainkan oleh user. *Use case* tersebut terdapat beberapa pilihan tingkatan yaitu *easy*, *medium* dan *hard*. *Use case Play Game* merupakan sebuah *use case* untuk perintah memulai permainan dan menampilkan arena permainan dan para pemain.

3.2.2 Sequence Diagram

Sequence diagram atau diagram sekuensial merupakan salah satu diagram interaksi yang menjelaskan bagaimana suatu operasi itu dilakukan. Pesan (*message*) apa yang dikirim dan kapan pelaksanaannya. Diagram ini diatur berdasarkan waktu. Obyek – obyek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut

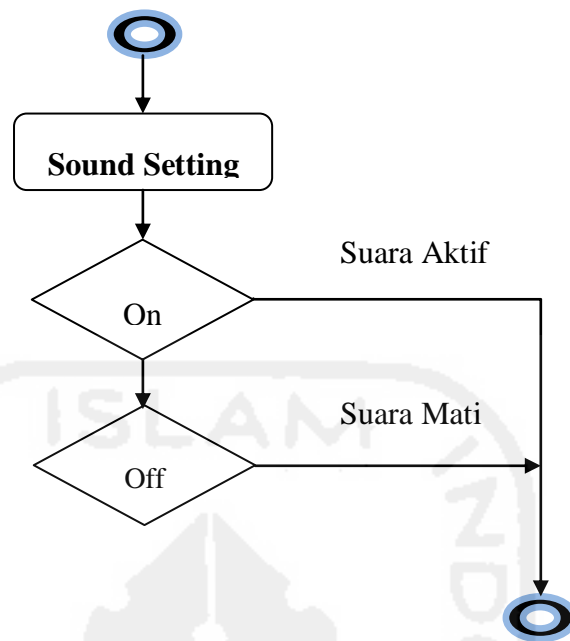
Dengan adanya *sequence diagram* ini maka dapat diketahui urutan proses pada game Bentengan.



Gambar 3.2 *Sequence Diagram*

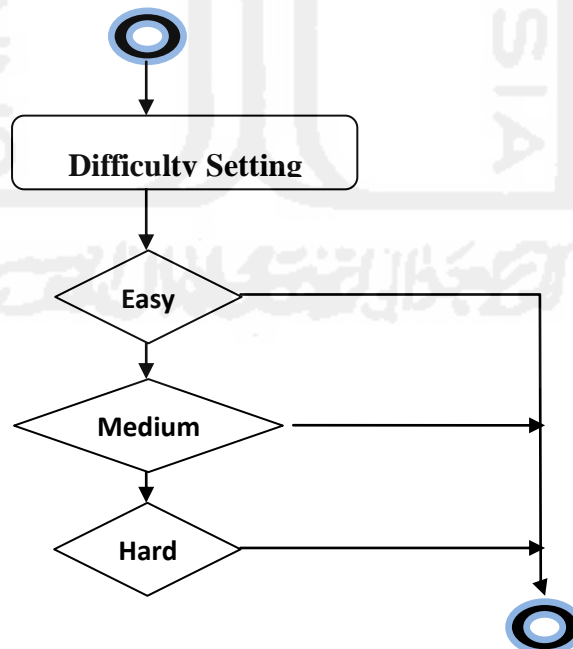
3.2.3 *Activity Diagram*

Activity diagram atau diagram aktifitas adalah sebuah diagram yang menggambarkan aktifitas-aktifitas dalam sistem yang sedang dibuat. Beberapa aktifitas itu adalah seperti pengaturan suara, pilihan tingkat kesulitan, dan saat memainkan permainan. Pada diagram aktifitas juga dapat menggambarkan aktifitas yang lebih dari satu proses (*parallel*). Untuk aktifitas diagram aplikasi ini dapat dilihat pada Gambar 3.3, Gambar 3.4 dan Gambar 3.5



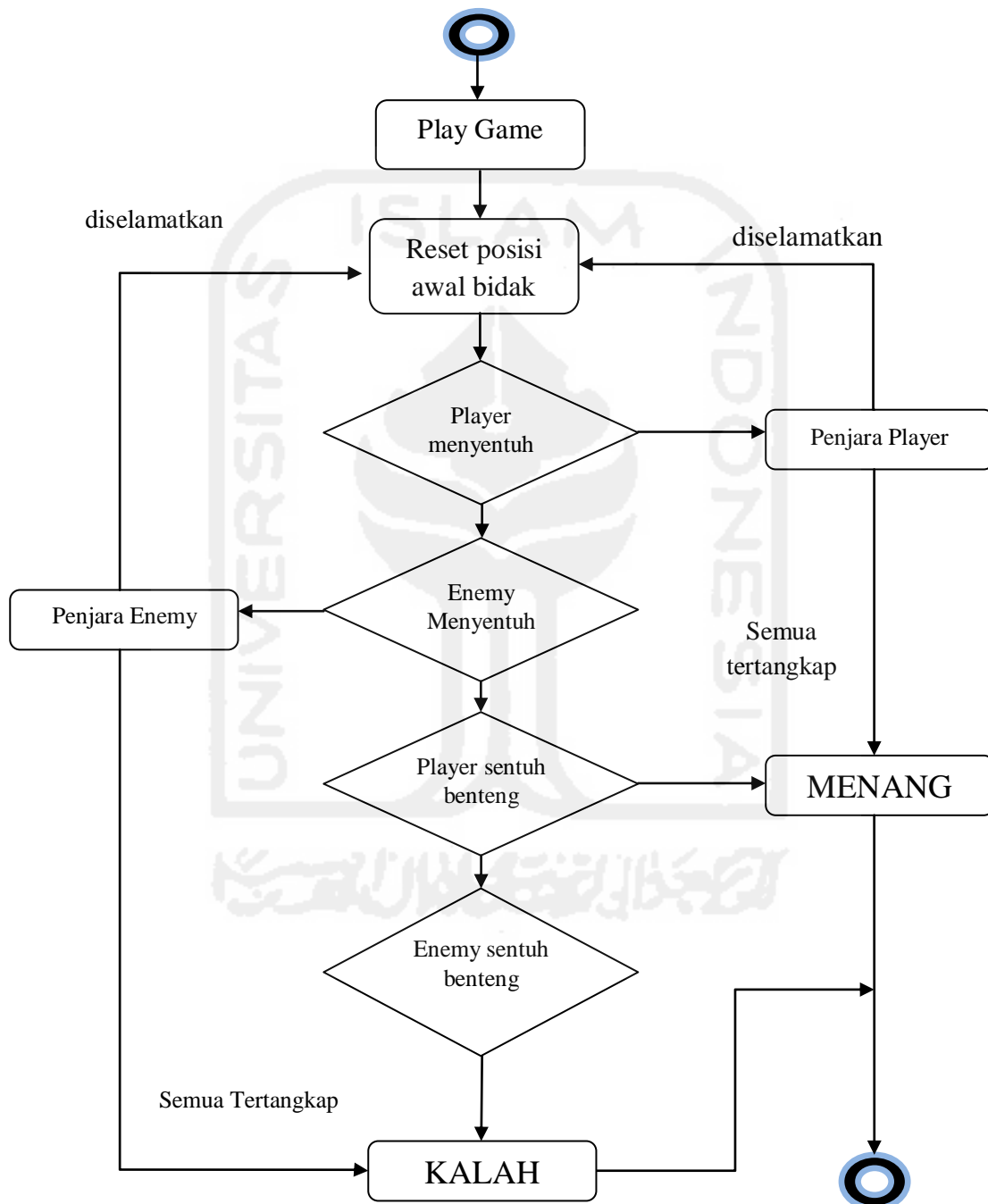
Gambar 3.3 Diagram Aktifitas *Sound Setting*

Diagram tersebut tentang aktifitas pengaturan suara dalam posisi menyala atau mati.



Gambar 3.4 Diagram Aktifitas *Difficulty Setting*

Diagram tersebut tentang aktifitas pengaturan tingkat kesulitan dari *easy*, *medium* dan *Hard*. Pengaturan ini mempengaruhi kecepatan gerak enemy saat permainan berlangsung.



Gambar 3.5 Diagram Aktifitas *Play Game*

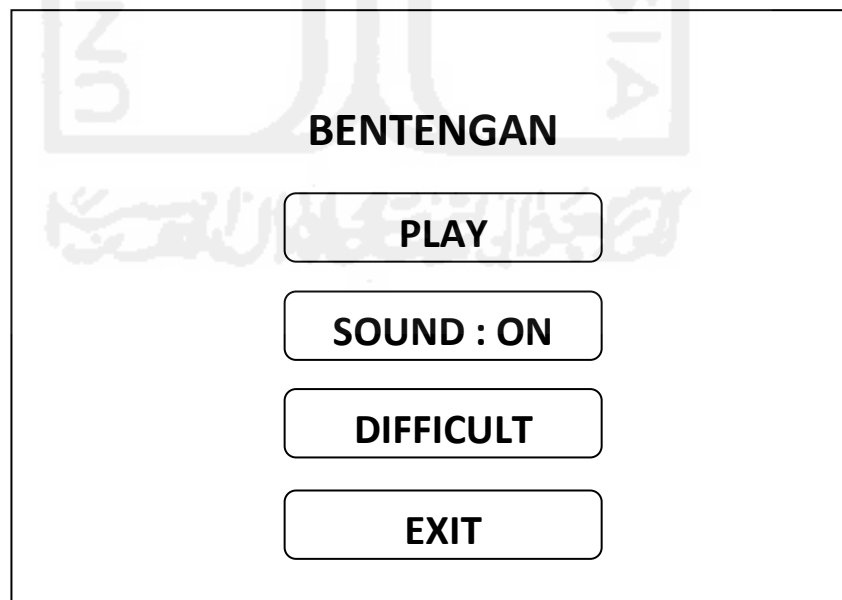
Aktifitas yang terjadi saat play game dan kondisi saat user menang dan juga kalah.

3.2.4 Perancangan Antar Muka

Antar muka (*User nterface*) bertujuan untuk menampilkan *display* atau tampilan yang berguna untuk menunjukkan memberikan informasi mengenai bentuk atau tampilan dalam permainan tersebut. Antarmuka yang yang baik adalah antarmuka yang mudah dipahami oleh user sehingga user tidak bingung dalam menjalankan sistem tersebut. Kemudahan user dalam menggunakan sistem tersebut merupakan salah satu keberhasilan suatu *user interface*.

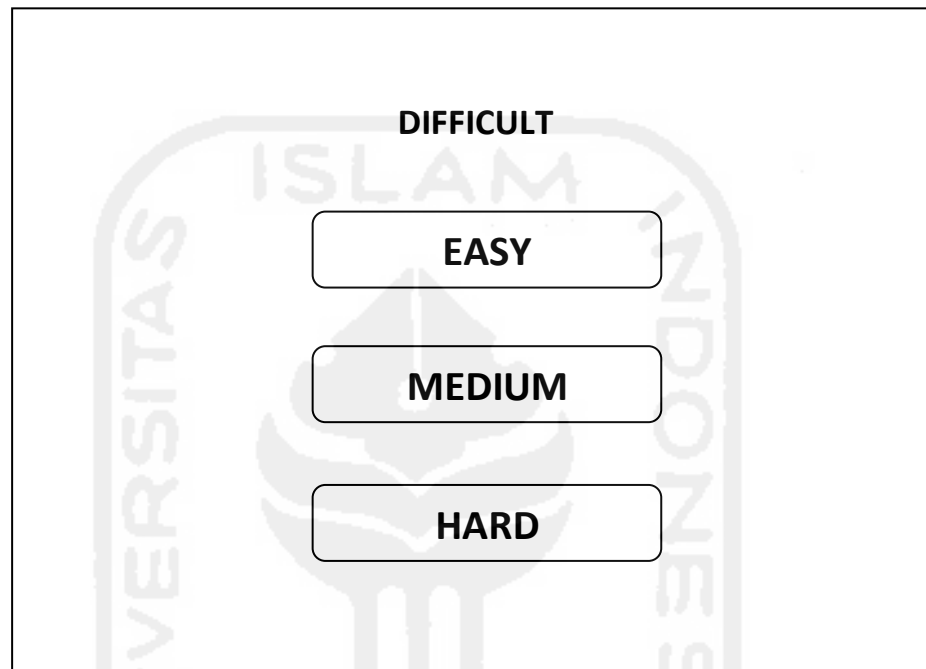
Berikut ini rancangan *user interface* (UI) yang akan di gunakan dalam game bentengan ini :

Pada rancangan antar muka menu utama terdiri dari 4 buah tombol yaitu tombol *Play*, tombol *Difficult*, tombol *Level*, dan tombol *Exit*. Tombol *Play* di gunakan untuk memulai permainan, tombol *Difficult* di gunakan untuk meilih tingkat kesulitan, sedangkan tombol *Exit* untuk keluar dari aplikasi. Berikut tampilan antarmuka dari menu utama dapat dilihat dari gambar 3.4.



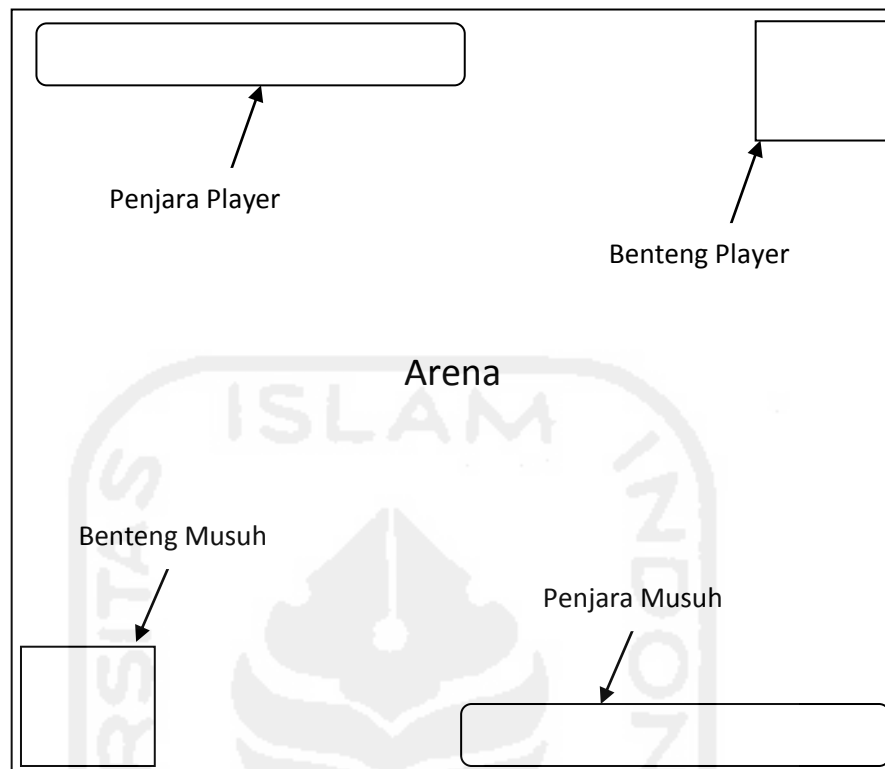
Gambar 3.6 Tampilan Home

Pada rancangan antarmuka menu *DIFFICULT* terdiri dari 3 tombol pilihan tingkat kesulitan yaitu tombol Mudah, tombol sedang, tombol Sulit. Pada tingkat kesulitan yang dipilih berpengaruh terhadap kecepatan gerak dari *enemy*. Berikut ini tampilan antarmuka dari menu *Level* dapat dilihat pada Gambar 3.7.



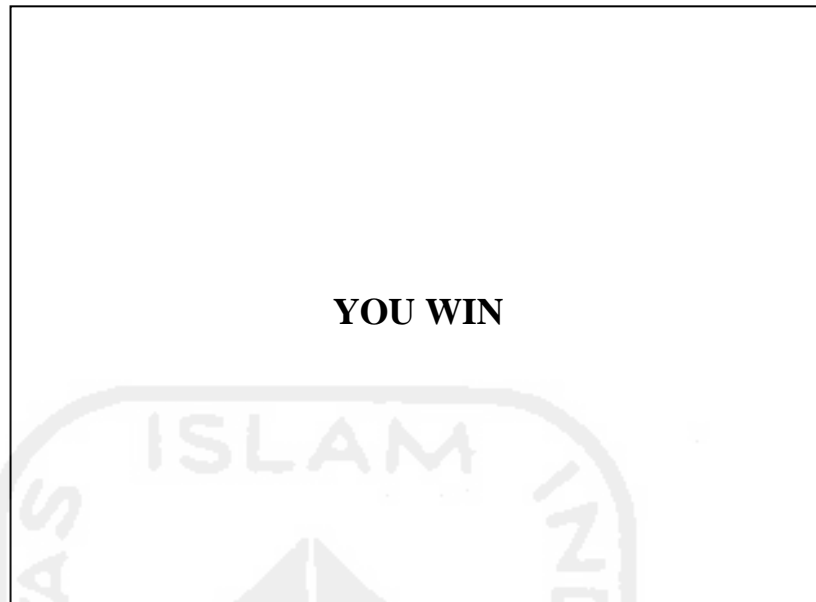
Gambar 3.7 Tampilan Pilihan kesulitan

Didalam tampilan antarmuka permainan, terdapat sebuah bidang yang merupakan area permainan dimana disetiap sisi bawah kiri dan kanan atas merupakan benteng dari player dan *enemy*. Bentuk kotak pada kiri bawah merupakan benteng *enemy* sedangkan kota pada kanan atas merupakan benteng dari player. Pada posisi tengah atas dan tengah bawah merupakan penjara. Pada lingkaran panjang di tengah bawah merupakan penjara milik *enemy*, sedangkan lingkaran panjang pada tengah atas merupakan penjara player. Berikut ini tampilan antarmuka dari tampilan permainan dapat dilihat pada Gambar 3.8.



Gambar 3.8 Tampilan Arena Bentengan

Terdapat dua jenis tampilan antarmuka permainan saat permainan berakhir yang akan ditampilkan salah satu saja sesuai dengan hasil yang telah diselesaikan oleh pengguna. Yaitu Status permainan menang dan status permainan kalah. Ditampilkannya status menang ketika berhasil menangkap semua *enemy* atau berhasil menyentuh benteng *enemy* tanpa tersentuh oleh *enemy*. Sedangkan status kalah akan muncul saat semua player dipenjara atau *enemy* berhasil menyentuh benteng player. Rancangan tampilan antarmuka keterangan status kalah dan menang dapat dilihat pada Gambar 3.9 dan Gambar 3.10.



Gambar 3.9 Gambar Status Permainan Menang



Gambar 3.10 Gambar Status Permainan Kalah