

APLIKASI PERMAINAN TRADISIONAL INDONESIA “BENTENGAN” BERBASIS ANDROID

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat

Untuk Memperoleh Gelar Sarjana Teknik Informatika



Disusun Oleh :

Nama : Agusta Wahyu Saputra

No. Mahasiswa : 09523119

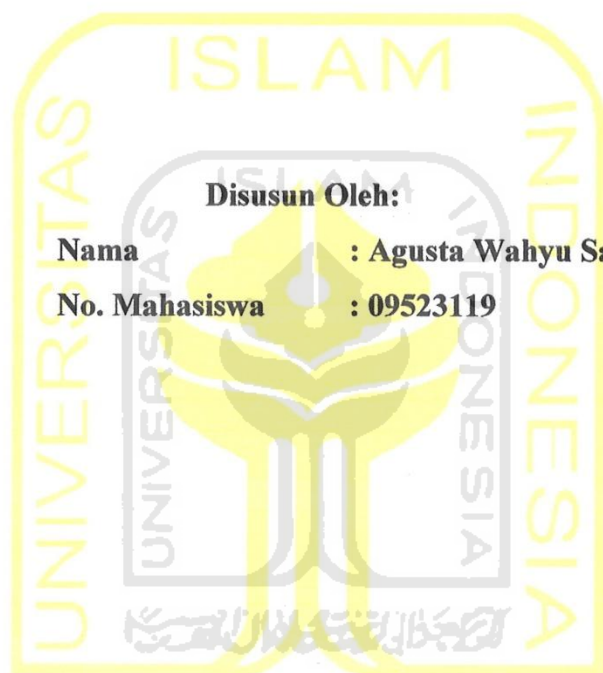
**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2016

LEMBAR PENGESAHAN PEMBIMBING

**APLIKASI PERMAINAN TRADISIONAL INDONESIA “BENTENGAN”
BERBASIS ANDROID**

TUGAS AKHIR



Disusun Oleh:

Nama : Agusta Wahyu Saputra

No. Mahasiswa : 09523119

Yogyakarta, 22 Agustus 2016

Pembimbing Tunggal

A handwritten signature in black ink, appearing to read 'Taufiq Hidayat', is written over a horizontal line.

Taufiq Hidayat S.T.,M.Cs.

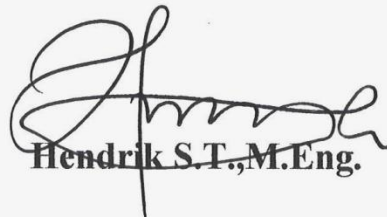
LEMBAR PENGESAHAN PENGUJI**APLIKASI PERMAINAN TRADISIONAL INDONESIA “BENTENGAN”
BERBASIS ANDROID****TUGAS AKHIR**

Disusun Oleh:

Nama : Agusta Wahyu Saputra

No. Mahasiswa : 09523119

Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat Untuk
Memperoleh Gelar Sarjana Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia

TIM PENGUJI**Taufiq Hidayat S.T.,M.Cs.****Ketua****Izzati Muhimmah S.T.,M.Sc.,Ph.D.****Anggota****Elyza Gustri Wahyuni S.T.,M.Cs.****Anggota****Mengetahui,****Ketua Program Studi Teknik Informatika****Fakultas Teknologi Industri****Universitas Islam Indonesia**
Hendrik S.T.,M.Eng.

LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Saya yang bertanda tangan di bawah ini :

Nama : Agusta Wahyu Saputra
No. Mahasiswa : 09 523 119
Tugas Akhir dengan Judul :

APLIKASI PERMAINAN TRADISIONAL INDONESIA “BENTENGAN” BERBASIS ANDROID

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti bahwa ada beberapa bagian dari karya ini bukanlah hasil karya saya sendiri, maka saya akan siap menanggung resiko dan konsekuensi apapun.

Demikianlah pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 22 Agustus 2016



(Agusta Wahyu Saputra)

HALAMAN PERSEMBAHAN

Rasa syukur saya persembahkan kepada sang pemberi nikmat Allah SWT dan nabi besarnya Muhammad SAW.

Tugas akhir yang telah saya buat ini saya persembahkan kepada ibu, ayah, dan keluarga saya yang selalu memberikan do'a dan semangat di setiap nafasnya sehingga akhirnya tugas akhir ini dapat selesai.

Terima kasih juga kepada saudara dan teman-teman seperjuangan saya yang telah membantu dan member semangat serta doa untuk kelancaran dalam pengerjaan tugas akhir ini.



MOTTO

“Intelligence is not the determinant of success, but hard work is the real determinant of your success.”

"Seseorang yang berhenti belajar adalah orang lanjut usia, meskipun umurnya masih remaja. Seseorang yang tidak pernah berhenti belajar akan selamanya menjadi pemuda"
(Henry Ford)

"Pengetahuan yang benar tidak diukur dari seberapa banyak Anda menghafal dan seberapa banyak yang mampu Anda jelaskan, melainkan, pengetahuan yang benar adalah ekspresi kesalehan (melindungi diri dari apa yang Allah larang dan bertindak atas apa yang Allah amanatkan)"
(Diriwayatkan oleh Abu Na'im)

"Engkau berpikir tentang dirimu sebagai seonggok materi semata, padahal di dalam dirimu tersimpan kekuatan tak terbatas"
(Ali Bin Abi Thalib RA.)

“Allah tidak akan membebani seorang manusia melainkan sesuai dengan kemampuan manusia itu sendiri”
(Al Baqarah : 286)

SELALU AWALI DENGAN UCAPAN BASMALAH

KATA PENGANTAR



Assalamu'alaikum Wr. Wb

Puji syukur penulis panjatkan kehadirat Allah SWT atas segala limpahan rahmat dan hidayah-Nya, sehingga penulis dapat menyusun laporan Skripsi yang berjudul **“Aplikasi Permainan Tradisional Indonesia “Bentengan” Berbasis Android”** dengan baik. Laporan ini dimaksudkan untuk memenuhi salah satu syarat akademis untuk menyelesaikan pendidikan jenjang strata-1 (S-1) gelar sarjana pada Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Penulis menyadari bahwa dalam penyusunan Laporan ini tidak dapat diselesaikan tanpa bantuan dan bimbingan dari berbagai pihak. Pada kesempatan ini penyusun ingin menyampaikan rasa terima kasih kepada :

1. Allah SWT yang telah melimpahkan rahmat dan kemudahan kepada penulis, sehingga laporan kerja praktek ini dapat selesai dengan lancar.
2. Kepada kedua orang tua saya yang selalu senantiasa memberi dukungan dan doa sehingga terciptanya semangat untuk menyelesaikan laporan kerja praktek ini.
3. Bapak Dr. Ir. Harsoyo, M.sc. selaku rector Universitas Islam Indonesia
4. Bapak Dr. Imam Djati Widodo, M.Eng.M.Sc selaku dekan Fakultas Teknologi Industri, Universitas Islam Indonesia.
5. Kepada bapak Hendrik, ST.,M.Eng selaku ketua jurusan Teknik Informatika dan para staf pengajar Universitas Islam Indonesia yang berperan membantu memperlancar penyelesaian tugas laporan kerja praktek ini..
6. Kepada dosen pembimbing saya bapak Taufiq Hidayat S.T.,M.Cs. yang selalu sabar membimbing saya dan memberi banyak arahan kepada saya.
7. Teman-teman yang telah memberikan masukan dalam penelitian ini yang tidak dapat disebutkan satu per satu.

Penulis menyadari bahwa dalam penyusunan laporan skripsi ini masih belum sempurna. Penulis mengharapkan kritik dan saran dari semua pihak yang bersifat membangun guna kesempurnaan dalam penulisan berikutnya menjadi lebih baik.

Akhir kata, penulis berharap semoga skripsi ini dapat berguna bagi semua pihak yang membutuhkan.

Yogyakarta, 22 Agustus 2016

Penulis



SARI

Pada saat ini teknologi perangkat telekomunikasi berkembang dengan sangat cepat khususnya telpon pintar atau smartphome. Sistem operasi yang paling populer digunakan pada smartphome saat ini adalah sistem android. Dengan munculnya smartphome android banyak anak-anak jaman sekarang yang menghabiskan waktunya untuk bermain gadget baik itu internet, sosmed dan juga game. Sehingga saat ini permainan tradisional mulai dilupakan dan digantikan dengan gadget. Salah satu permainan tradisional yang mulai dilupakan adalah permainan bentengan.

Pada penelitian ini penulis mencoba mengenalkan kembali permainan bentengan dengan dikemas dan menyajikannya dalam bentuk game android. Game ini dibuat berdasarkan permainan bentengan asli dan aturan yang diterapkan juga sesuai dengan permainan yang sebenarnya. Penulis menggunakan software App Inventor dalam pengembangan game bentengan tersebut. diharapkan dengan adanya game bentengan di android dapat mengenalkan kembali salah satu permainan tradisional di Indonesia kepada masyarakat. Game ini pada setiap sisi terdiri dari 5 pemain baik enemy maupun player. Terdapat beberapa pilihan kesulitan dalam permainan dari mudah hingga sulit dan hanya bisa dimainkan oleh satu player atau hanya single player.

Kata kunci : smartphome. Android, game android, gadget

TAKARIR

<i>Hero</i>	Pahlawan
<i>Enemy</i>	Musuh
<i>Player</i>	Pemain
<i>Interface</i>	Antarmuka
<i>Gadget</i>	Alat
<i>Game</i>	Permainan
<i>Software</i>	Perangkat Lunak
<i>Method</i>	Metode



DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PENGESAHAN PENGUJI	iii
LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTTO	vi
KATA PENGANTAR	vii
SARI	ix
TAKARIR	x
DATAR ISI	xi
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	2
1.5. Manfaat Penelitian	3
1.6. Metodologi Penelitian	3
1.6.1 Metode Pengumpulan Data	3
1.6.2 Metode Pembangunan Aplikasi	3
1.7. Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Android	5
2.2 Game	8
2.2.1 Klasifikasi Game	8
2.2.2 Tipe Game	9
2.3 Bentengan	12

2.4 App Inventor	13
BAB III METODOLOGI	14
3.1 Analisis Kebutuhan Sistem.....	14
3.1.1 Peraturan Permainan Bentengan	14
3.1.2 Analisis Kebutuhan Input	15
3.1.3 Analisis Kebutuhan Output	15
3.1.4 Analisis Kebutuhan Proses	16
3.1.5 Analisis Kebutuhan Perangkat Keras	16
3.1.6 Analisis Kebutuhan Perangkat Lunak	16
3.2 Perancangan Sistem	17
3.2.1 <i>Use Case Diagram</i>	17
3.2.2 <i>Sequence Diagram</i>	18
3.2.3 <i>Activity Diagram</i>	19
3.2.4 Perancangan Antar Muka	22
BAB IV HASIL DAN PEMBAHASAN	26
4.1 Implementasi Sistem	26
4.1.1 <i>Interface Permainan</i>	26
4.1.2 Implementasi Gerakan Player	28
4.1.3 Implementasi Gerakan Enemy	29
4.1.4 Implementasi Prioritas Pemain	31
4.1.5 Implementasi Tabrakan Hero.....	32
4.1.6 Implementasi <i>Oncollide</i>	33
4.1.7 Implementasi Memasukkan Penjara.....	34
4.1.7.1 <i>Prison Hero</i>	34
4.1.7.2 <i>Prison Enemy</i>	36
4.1.8 Implementasi Keluar Penjara	37
4.1.8.1 Pada Hero	38
4.1.8.2 Pada Enemy	39
4.1.9 Kondisi Menang dan Kalah	40

4.2 Pengujian Aplikasi	41
BAB V	43
5.1 Kesimpulan.....	43
5.2 Saran	43
DAFTAR PUSTAKA	xv
LAMPIRAN	



DAFTAR GAMBAR

Gambar 3.1 <i>Use Case Diagram</i>	17
Gambar 3.2 <i>Sequence Diagram</i>	19
Gambar 3.3 Diagram Aktifitas <i>Sound Setting</i>	20
Gambar 3.4 Diagram Aktifitas <i>Difficult Setting</i>	20
Gambar 3.5 Diagram Aktifitas <i>Play Game</i>	21
Gambar 3.6 Tampilan Home	22
Gambar 3.7 Tampilan Pilihan Kesulitan	23
Gambar 3.8 Tampilan Arena Bentengan	24
Gambar 3.9 Gambar Status Permainan Menang	25
Gambar 3.10 Gambar Status Permainan Kalah	25
Gambar 4.1 Tampilan Home	27
Gambar 4.2 Tampilan Arena	27
Gambar 4.3 Tampilan Pilihan Kesulitan	28
Gambar 4.4 Kode blok <i>when.ImageSprite1.Dragged</i>	29
Gambar 4.5 Kode blok <i>to.UpdateSteering.enemy</i>	31
Gambar 4.6 Kode blok <i>when.castle1.NoLongerCollidingWith</i>	32
Gambar 4.7 Kode blok <i>when.ImageSprite1.Collidedwith</i>	33
Gambar 4.8 Kode blok <i>to.OnCollide.heroid.enemyid</i>	34
Gambar 4.9 Kode blok <i>to.doPrisonHero.heros</i>	35
Gambar 4.10 Kode blok <i>to.doPrisonEnemy.enema</i>	37
Gambar 4.11 Kode blok <i>to.OnRescueHero.hero1.hero2</i>	38
Gambar 4.12 Kode blok <i>to.OnRescueEnemy.enemy1.enemy2</i>	39
Gambar 4.13 Status menang dengan blok <i>when.castle2.collidedwith</i>	40
Gambar 4.14 Status kalah dengan blok <i>when.castle1.collidedwith</i>	41

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan berjalannya waktu teknologi berkembang dengan pesat khususnya perangkat telekomunikasi. Saat ini perangkat telekomunikasi telah membuat pekerjaan manusia menjadi lebih mudah. Perangkat telekomunikasi yang berupa *handphone* atau HP sekarang ini tidak hanya sekedar untuk menelpon maupun sms saja tapi telah menjadi sebuah komputer portable yang biasa disebut dengan *smartphone*.

Beberapa tahun terakhir ini, perkembangan teknologi informasi terutama dalam hal perangkat lunak berlangsung sangat cepat. Hal ini ditandai dengan munculnya berbagai perangkat yang sudah mempunyai sistem komputasi. Salah satu perangkat komunikasi tersebut adalah *smartphone*, pengguna *smartphone* di Indonesia diprediksi mencapai 82 juta pada tahun 2014. (Kemkominfo, 2014). Perkembangan teknologi yang semakin canggih ini disertai dengan munculnya sistem operasi yang terbaru untuk mengimbangnya. Salah satu sistem operasi yang paling populer yaitu android.

Game merupakan salah satu daya tarik dan tidak bisa lepas dari perangkat android. Baik tua maupun muda banyak masyarakat khususnya di Indonesia yang gemar bermain game. Orang-orang tertarik dengan grafik dari game maupun alur permainan dari game tersebut.

Anak-anak jaman sekarang sebagian besar waktunya hanya berada di rumah dan digunakan untuk bermain *gadget* baik itu internet, sosmed, dan juga game. Sehingga permainan-permainan yang biasanya dimainkan anak-anak jaman dulu seperti petak umpet, kelereng, grobak sodor, lompat tali, kejar kejaran, lompat tali dll perlahan mulai hilang.

Salah satu permainan yang sudah jarang dimainkan saat ini adalah bentengan. Bentengan merupakan salah satu permainan tradisional yang biasa di mainkan anak-anak di daerah Jawa Tengah.

Penelitian yang dilakukan dalam tugas akhir ini membuat sebuah aplikasi berupa game tradisional tentang bentengan. Untuk mengenalkan kembali permainan tradisional kepada masyarakat khususnya anak-anak, maka di game ini dikemas dalam *platform* android sehingga mempunyai daya tarik dan terkesan tidak membosankan.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut maka dapat dirumuskan bagaimana mengembangkan sebuah game yang dapat menjadi hiburan sekaligus sebagai pengenalan terhadap permainan lokal Bentengan yang sudah mulai dilupakan.

1.3 Batasan Masalah

Batasan-batasan masalah yang digunakan antara lain :

1. Game ini dibuat hanya berdasarkan game bentengan saja dan bukan game lain.
2. Game ini mempunyai mempunyai 3 pilihan kesulitan yaitu : *Easy*, *Medium* dan *Hard* dan jumlah pemain setiap kelompok terdiri dari 5 pemain.

1.4 Tujuan Penelitian

Adapun tujuan yang hendak dicapai dari penyusunan tugas akhir ini adalah mengembangkan aplikasi game yang dapat mengenalkan salah satu permainan lokal Bentengan kepada masyarakat khususnya pengguna *smartphone* android. Kepedannya diharapkan permainan Bentengan bisa dikenal lagi oleh masyarakat.

1.5 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat memberikan manfaat antara lain masyarakat khususnya pengguna android dapat mengenal permainan Bentengan dan dapat melestarikannya sehingga tidak hilang di telan jaman.

1.6 Metodologi Penelitian

Metode yang digunakan dalam penelitian ini meliputi pengumpulan data dan pembangunan sistem.

1.6.1 Metode Pengumpulan Data

Metode pengumpulan data ini dilakukan dengan cara mengumpulkan beberapa referensi *game* Android yang sejenis dengan yang bertemakan permainan tradisional. Pada pengumpulan data ini mempelajari komponen-komponen apa saja yang terdapat pada *Game* tersebut dan apa yang perlu digunakan dalam penelitian ini.

1.6.2 Metode Pembangunan Aplikasi

1. Analisis Sistem

Sistem dianalisis dengan mengidentifikasi masalah, model keputusan yang digunakan, analisis kebutuhan sistem berupa *input*, *output*, proses, dan *interface* sebagai batasan dalam pembuatan sistem.

2. Perancangan

Perancangan dalam hal ini terdiri dari beberapa macam perancangan seperti *Data Flow Diagram* (DFD), perancangan *database*, dan perancangan *interface* (antar muka)

3. Implementasi

Rancangan atas *design* sistem diimplementasikan ke dalam bahasa pemrograman komputer yaitu dengan menggunakan aplikasi *app Inventory*

4. Pengujian Sistem

Sistem yang dibuat diujikan untuk memeriksa kesalahan yang terjadi dalam sistem untuk kemudian dilakukan perbaikan sistem.

1.7 Sistematika Penulisan

Dalam penyusunan tugas akhir ini, sistematika penulisan dibagi menjadi 5 bab yang disusun sebagai berikut :

BAB I PENDAHULUAN

Berisi pembahasan masalah umum yang meliputi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Berisikan tentang pembahasan mengenai pengertian tentang android, game, dan juga sistem yang digunakan dalam mengembangkan game bentengan.

BAB III METODOLOGI

Berisikan tentang analisis kebutuhan sistem yang terdiri dari kebutuhan fungsional, *input*, *output*, proses, perangkat keras, dan perangkat lunak. Perancangan sistem yang terdiri dari *use case diagram*, *sequence diagram*, diagram aktifitas dan antarmuka.

BAB IV HASIL DAN PEMBAHASAN

Memuat uraian hasil implementasi dari setiap aktifitas dan bagian-bagian yang dilakukan dalam pembuatan sistem. Baik berupa implementasi sistem dan pengujian sistem

BAB V SIMPULAN DAN SARAN

Memuat hasil kesimpulan yang merupakan rangkuman dari hasil pembuatan *Game Bentengan* ,serta saran-saran untuk pengembangan sistem selanjutnya.

BAB II

LANDASAN TEORI

2.1 Android

Android adalah sebuah sistem operasi dari google yang banyak digunakan di perangkat bergerak layar sentuh khususnya handphone atau telpon pintar (*smartphone*). Sistem operasi android memanfaatkan kernel linux. Android awalnya dikembangkan oleh android.inc, dengan dukungan financial dari Google, yang kemudian di beli oleh Google pada tahun 2005 dan dirilis secara resmi pada tahun 2007.

Android adalah sistem operasi dengan sistem terbuka (*open source*) dan Google merilis kodenya dibawah lisensi Apache. Dengan sistem pengkodean yang terbuka maka memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh pembuat perangkat lunak, operator nirkabel, dan pengembang aplikasi. Selain itu android memiliki komunitas pengembang aplikasi yang terus mengembangkan sistem android. Pada bulan Oktober 2013, ada lebih dari satu juta aplikasi yang tersedia untuk android dan sekitar 50 miliar aplikasi telah di download dari *Google Play* yang merupakan toko aplikasi utama android (wikipedia1, 2016).

Dalam perkembangan hingga saat ini Android memiliki beberapa versi Android, berikut versi-versi Android dari pertama rilis hingga sekarang (wikipedia2, 2016) :

1. Tanpa nama (versi 1.0 dan 1.1)

Sistem android versi 1.0 ini pertama kali hadir di tahun 2008, tepatnya pada oktober 2008. Ponsel pertama yang menggunakan system android adalah HTC

2. Android Cupcake (versi 1.2 – 1.5)

Pada 27 April 2009, Android 1.5 dirilis Versi ini adalah rilis pertama yang secara resmi menggunakan nama kode berdasarkan nama-nama

makanan pencuci mulut ("Cupcake"). Dengan penamaan Cupcake ini maka penamaan dari versi Android dimulai.

3. Android Donut (versi 1.6)

Versi selanjutnya adalah Donut, versi ini dirilis pada 15 september 2009, tentu donut sudah tak asing lagi bagi telinga orang Indonesia.

4. Android Éclair (versi 2.0 – 2.1)

Android versi ini dirilis Pada 26 Oktober 2009 dengan dinamai Éclair dengan rilisnya android versi ini google menambahkan fitur Bluetooth 2.1, flash dan kamera dengan digital zoom, multi-touch, live wallpaper dan lainnya.

5. Android Froyo (versi 2.2 – 2.2.3)

Versi ini dirilis pada bulan Mei 2010 dengan update memperbaiki segi kecepatan dan pengadopsian Javascript dari browser Google Chrome dengan fitur-fiturnya.

6. Android Gingerbread (versi 2.3 -2.4)

versi ini dirilis pada bulan desember 2010. Smartphone pertama yang memakai versi android ini adalah Nexus S yang dikeluarkan oleh produsen Samsung. Android 2.3 Gingerbread ini merupakan OS Android yang paling lama berkuasa bahkan sampai sekarangpun beberapa vendor masih mengeluarkan ponsel dengan versi ini.

7. Android Honeycomb (versi 3.0 – 3.2)

versi ini merupakan versi yang ditujukan untuk gadget Tablet. Android Honeycomb rilis pada Februari 2011, kemudian upgrade ke versi 3.1 dan 3.2.

8. Android Ice Cream Sandwich (versi 4.0)

Android ini rilis pada 19 October 2011. Yang baru dalam android ini adalah perubahan interface dari android sebelumnya, antara lain pengoptimalan multitasking, variasi layar beranda yang bisa disesuaikan dan interaktivitas mendalam serta cara baru yang ampuh untuk berkomunikasi dan berbagi konten.

9. Android Jelly Bean (versi 4.1 – 4.3)

Android versi ini secara resmi dikenalkan ke publik sekitar Oktober 2012. Versi ini adalah yang tercepat dan terhalus dari semua versi Android. Fitur baru yang terapat di versi ini adalah meningkatkan kemudahan dan keindahan tampilan dari Ice Cream Sandwich dan memperkenalkan pengalaman pencarian Google yang baru di Android.

10. Android Kitkat (versi 4.4)

Android versi ini dirilis pada bulan oktober 2013. Android versi ini memiliki banyak fitur & semakin memanjakan para pengguna android. Diantaranya : Immersive mode, Akses kontak langsung dari aplikasi telepon, google now launcher, dan pastinya memiliki interface UI yang baru.

11. Android Lolipop (versi 5.0)

Android versi ini dirilis pada tanggal 12 November 2014. Pada versi ini meiliki beberapa perubahan tampilan dan sistem.

12. Android Marshmallow (versi 6.0)

Android versi ini dirilis pada tanggal 5 oktober 2015. Versi ini merupakan versi terbaru yang dirilis oleh android.

2.2 Game

Game atau permainan saat ini mungkin telah menjadi salah satu kebutuhan bagi masyarakat. Game adalah aktifitas / kegiatan yang melibatkan satu atau lebih pemain. Game dapat pula diartikan sebagai sebuah tujuan yang ingin dicapai pemain atau sekumpulan aturan yang menentukan apa yang dapat dilakukan pemain dan yang tidak dapat dilakukan. Game dimainkan terutama untuk hiburan dan kesenangan, tetapi dapat juga berfungsi sebagai sarana latihan , pendidikan dan simulasi

Secara umum game sendiri adalah permainan yang menggunakan media elektronik, dan sebuah media hiburan yang berbentuk multimedia yang dibuat semenarik mungkin agar pemain game tersebut dapat merasa terhibur dengan memperoleh sesuatu atau sebuah pencapaian tertentu (Krisan, 2016).

2.2.1 Klasifikasi Game

Dalam *game* terdapat beberapa jenis yang kemudian di bagi menjadi beberapa klasifikasi. Pembagian *game* menjadi beberapa klasifikasi berdasarkan dari tujuan penggunaan *game* tersebut menurut penggunaannya. Berikut beberapa klasifikasi *game* berdasarkan tujuan penggunaannya (Putra, 2016) :

1. *Game as Game*

Tujuan dari jenis ini adalah *game* sebagai sumber kesenangan dan *fun*.

2. *Game as Media*

Jenis *game* ini bertujuan untuk menyampaikan sebuah pesan atau informasi tertentu dengan media *game*.

3. *Game beyond Game*

Game beyond game biasa disebut dengan *gamefication* yang merupakan penerapan konsep atau cara berfikir *game design* ke dalam lingkup non-*game*.

2.2.2 Tipe Game

Tipe game ini di bagi menjadi beberapa kategori-kategori tertentu yang dibedakan berdasarkan *genre* atau alur permainan nya yang kadang di sebut sebagai *gameplay*. Terdapat beberapa kategori yang sering ditemui antara lain sebagai berikut (Imansaiki,2012):

1. *Role Play Game*

Role Play Game atau sering disebut RPG adalah game yang para pemainnya memainkan peran tokoh-tokoh khayalan dan berkolaborasi untuk merajut sebuah cerita bersama. Ada pula jenis lain game RPG ini, diantaranya :

- a. LARP (*Live Action Role Playing*) adalah game RPG dimana para pemain bisa melakukan gerakan fisik tokohnya. Biasanya pemain menggunakan kostum dan menggunakan alat-alat yang sesuai dengan tokoh, dunia dan cerita yang dia mainkan.
- b. MMORPG (*Massively Multiplayer Online Role-Playing Game*) adalah game RPG yang melibatkan ribuan pemain untuk bermain game bersama dalam dunia maya.

Contoh game MMORPG : Final Fantasy, Ragnarok, Avalon, RF, World of Warcraft, DotA, Perfect World

2. *Strategy*

Game *Strategy* merupakan game yang memiliki *gameplay* untuk mengatur *player* / pasukan milik kita untuk menyerang atau bertahan dari musuh dalam rangka memenangkan permainan

3. *Simulation*

Simulation atau game simulasi merupakan sebuah game yang mementingkan *realisme* atau semirip mungkin dengan situasi di kehidupan nyata

4. *Arcade*

Arcade adalah game yang tidak memiliki alur cerita dan tujuannya hanya untuk kesenangan dan biasanya dituntut untuk meraih poin setinggi tingginya.

5. *Adventure*

Adventure adalah genre game yang mempunyai alur cerita dan bersifat petualangan. Kita memainkan sebuah karakter yang akan terus dimainkan sampai alur cerita tamat.

6. *Shooter games*

Permainan tembak-menembak atau *Shooter games*, adalah sub-genre dari permainan aksi, meskipun banyak pemain yang menganggap bahwa ragam ini merupakan ragam yang berdiri sendiri. Ragam ini menguji kecepatan dan reaksi dari pemainnya. Tembak-menembak memiliki sub-genre tersendiri, yang semuanya terfokus pada "mengontrol seseorang yang menggunakan sebuah senjata. Beberapa *sub-genre* dari *Shooter games* :

- a. *First-Person Shooter* (FPS) menekankan ke menembak dengan perspektif orang-pertama yang dikontrol oleh pemain. Perspektif ini bertujuan untuk memberikan pemain rasa seakan benar-benar berada disana, dan dapat membuat pemain fokus membidik sasaran.
- b. *Third-Person Shooter* (TPS/3PS) menekankan ke menembak dari perspektif orang-ketiga atau karakter pemainnya dapat terlihat sepenuhnya. Perspektif ini memberikan pemain pandangan yang lebih luas akan lingkungan sekitar dari FPS.
- c. *Massively Multiplayer Online First-Person Shooter* (MMOFPS), merupakan permainan online yang menggabungkan pemain FPS pada sebuah dunia virtual dimana para pemain dapat berinteraksi satu dengan lainnya.

7. *Racing Game*

Permainan balap atau *Racing Game* juga termasuk dalam kategori ini, dimana ragam permainan video yang menuntut keterampilan pemain untuk mengemudi dalam sebuah kompetisi balap-membalap dan berusaha meraih posisi pertama melewati pesaing-pesaingnya.

Contoh permainan video dalam ragam ini adalah serial *Need for Speed* dan *Moto GP*.

8. *Fighting Game*

Fighting Game atau permainan pertarungan adalah jenis permainan dimana terjadi pertarungan satu lawan satu antara dua karakter, salah satunya dapat merupakan komputer. Ragam permainan ini biasanya dimainkan dengan menghubungkan kombinasi-kombinasi tombol untuk bertarung. Rata-rata gerakan yang dihasilkan oleh karakter dramatis dan tidak mungkin dilakukan di kehidupan nyata.

Contoh game jenis ini : *Street Fighter* dan Tekken

9. *Board Game*

Board Game merupakan sebuah game yang membutuhkan papan untuk memainkannya. Jenis game ini biasanya berupa game yang mengandalkan otak.

Contoh : Catur

10. *Sport*

Sport merupakan sebuah jenis game bertema olahraga. Sistem permainan akan berbeda tergantung dari jenis olah raga yang dimainkan

Contoh : Sepakbola, Basket, Tenis

2.3 Bentengan

Bentengan atau disebut juga dengan benteng-bentengan, merupakan sebuah permainan tradisional yang dimainkan oleh 2 kelompok atau grup dan setiap kelompok masing – masing terdiri dari 3 sampai 8 orang. Masing – masing kelompok kemudian memilih suatu tempat yang digunakan sebagai benteng, biasanya berupa tiang atau pilar.

Tujuan dari permainan bentengan adalah menyerang dan mengambil alih benteng lawan dengan menyentuh tiang atau pilar yang di pilih lawan sebagai benteng dan meneriakkan kata “benteng”. Kemenangan juga diraih dengan menangkap seluruh anggota lawan dengan menyentuh tubuh mereka. Untuk menentukan siapa yang berhak menjadi “penawan” dan “tertawan” ditentukan dari waktu terakhir si “penawan” atau “tertawan” menyentuh “benteng” mereka masing – masing. (wikipedia3. 2016)

Lebih jelas nya untuk aturan permainan Bentengan sebagai berikut :

1. Permainan bentengan terdiri dari 2 kelompok, masing-masing kelompok terdiri dari 4-5 pemain atau menyesuaikan jumlah orang yang ada (dalam aplikasi ini terdiri 5 pemain tiap kelompoknya).
2. Permainan bentengan dilakukan dengan tujuan melindungi benteng yang biasanya berupa tonggak kayu ataupun tiang dan juga pohon. Benteng di sini bertujuan sebagai markas atau *basecamp* masing-masing kelompok.
3. Pemain yang keluar dari benteng (melepas pegangan dari tiang / pohon) dianggap menyerbu terlebih dahulu. Jika pemain ini dikejar oleh lawan dan tersentuh oleh musuh maka dianggap tertangkap
4. Pemain yang tertangkap dapat diselamatkan dan dapat bermain kembali setelah di selamatkan temannya dengan cara menyentuh tangan atau bagian tubuhnya.

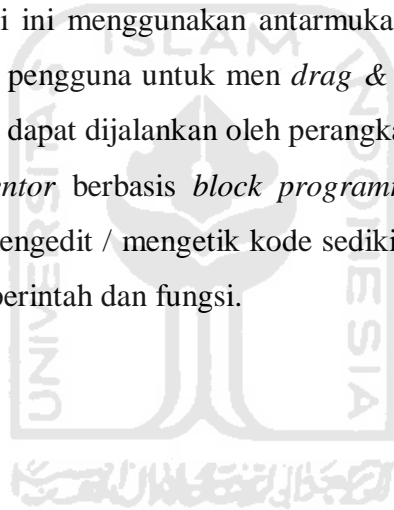
5. Permainan berakhir jika pemain dari salah satu tim dapat menyentuh benteng lawan atau tim tersebut dapat menangkap semua pemain lawan.

2.4 *APP Inventor*

App Inventor merupakan aplikasi yang awalnya dikembangkan oleh Google, namun kemudian saat ini dikelola oleh *Massachusetts Institute of Technology (MIT)*.

App Inventor memungkinkan para pengguna baru agar dapat memprogram dan dapat menciptakan aplikasi untuk perangkat lunak khususnya sistem operasi berbasis *Android*. Aplikasi ini menggunakan antarmuka grafis berbentuk blok – blok yang memungkinkan pengguna untuk men *drag & drop* obyek visual untuk menciptakan aplikasi yang dapat dijalankan oleh perangkat *Android*.

Aplikasi *App Inventor* berbasis *block programming* sehingga kita bisa membuat aplikasi tanpa mengedit / mengetik kode sedikitpun, hanya tinggal *drag & drop* symbol – symbol perintah dan fungsi.



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

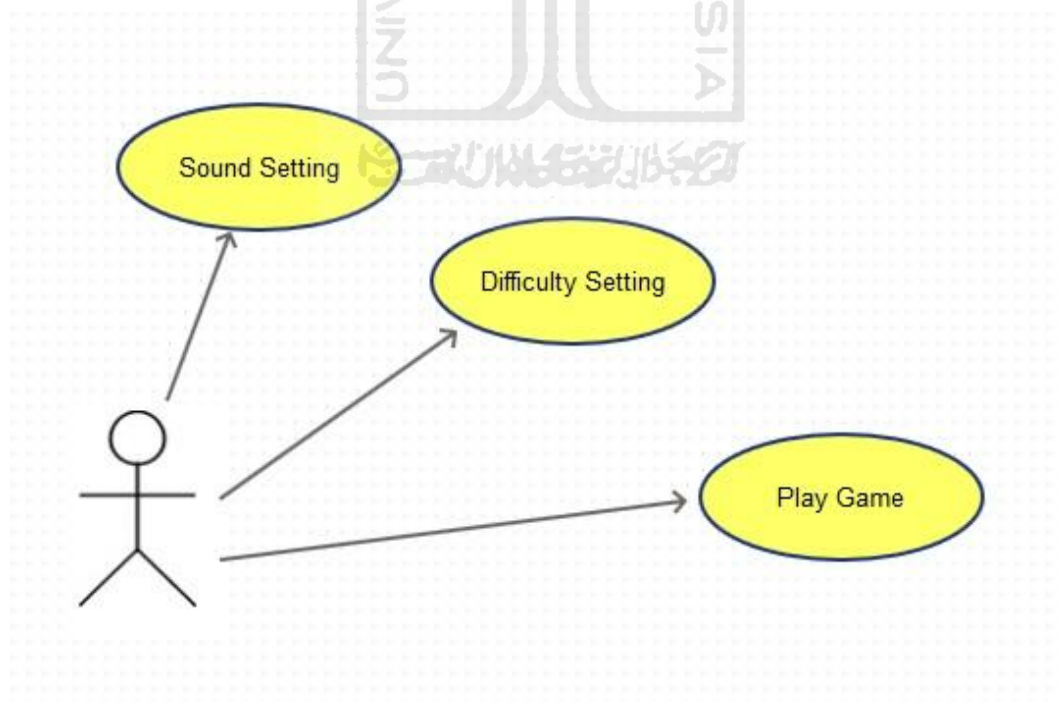
3.2 Perancangan Sistem

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



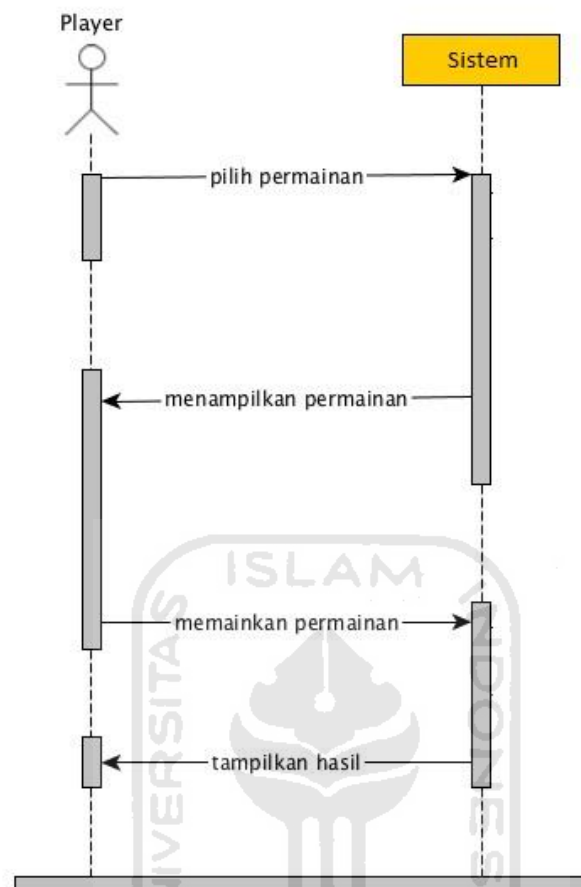
Gambar 3.1 *Use Case Diagram*

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

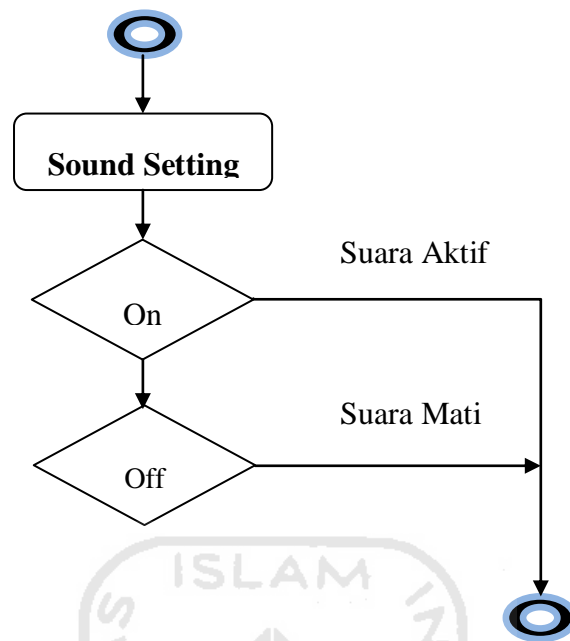
Dengan adanya *sequence diagram* ini maka dapat diketahui urutan proses pada game Bentengan.



Gambar 3.2 *Sequence Diagram*

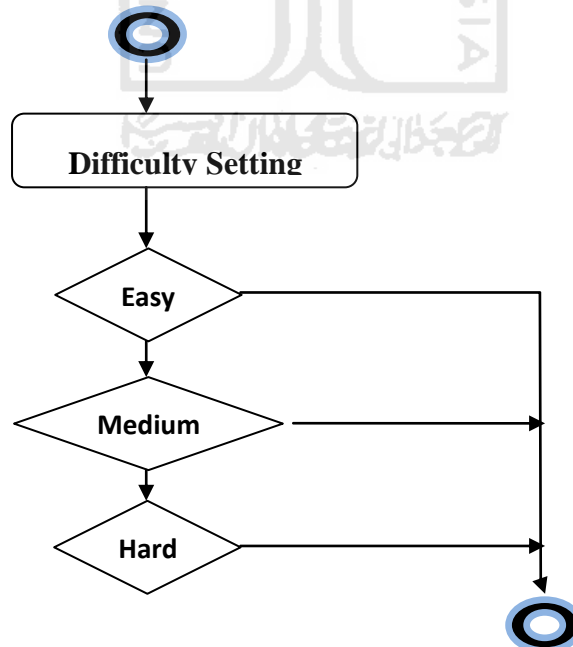
3.2.3 *Actifity Diagram*

Actifity 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



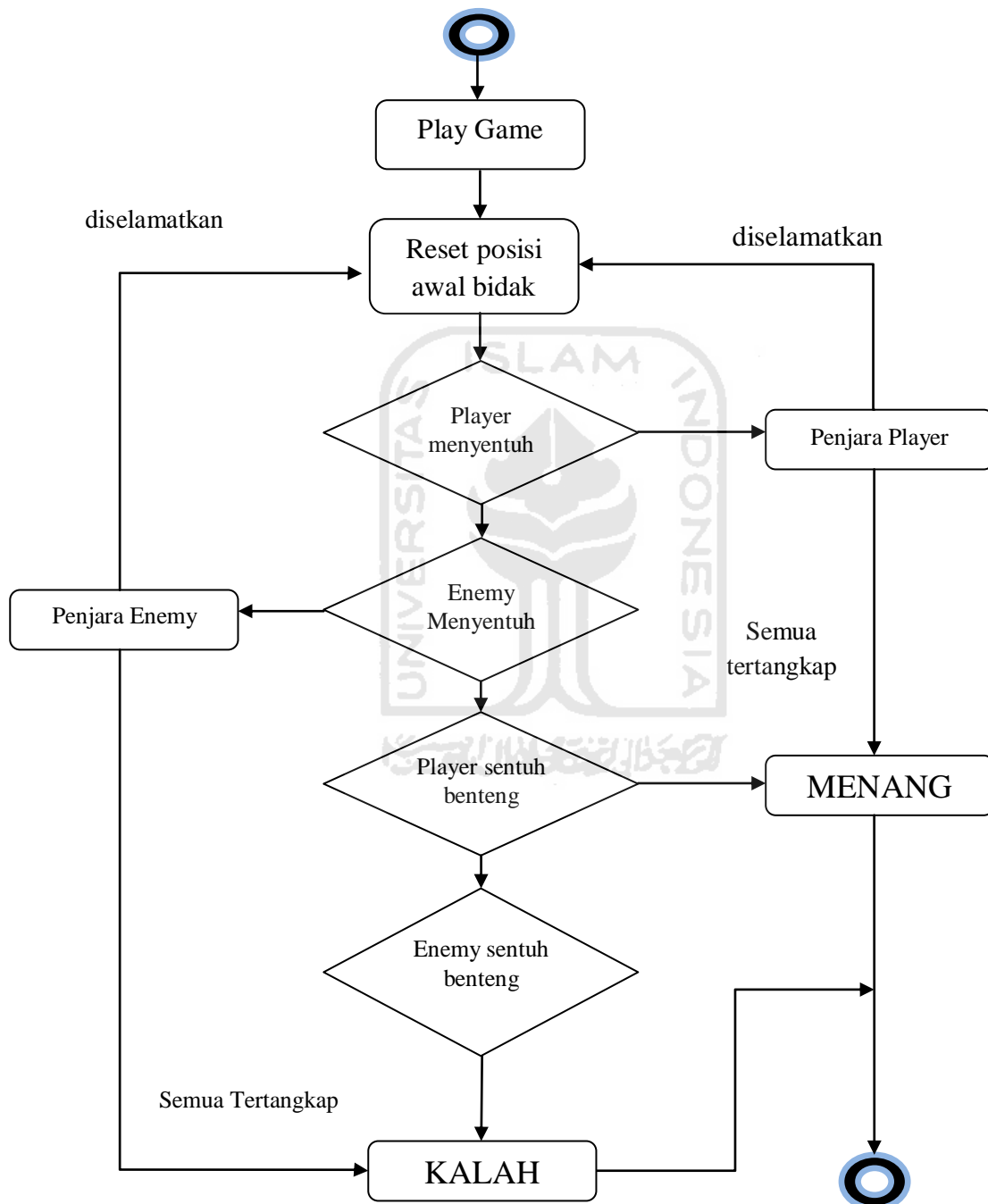
Gambar 3.3 Diagram Aktifitas *Sound Setting*

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*.

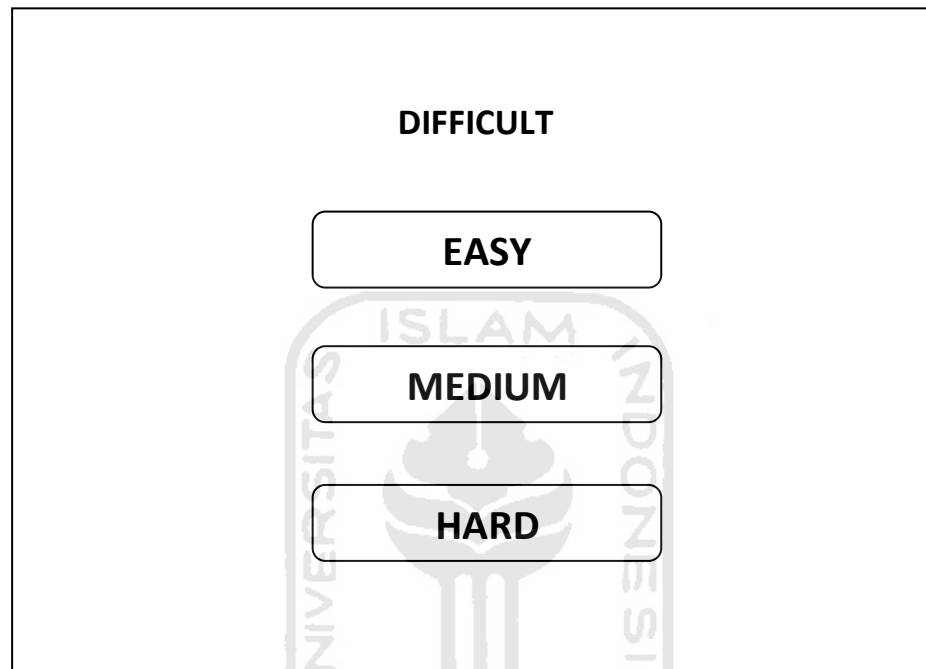
Berikut ini rancangan *user interface* (UI) yang akan di gunakan dalam game bentengan ini :

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.



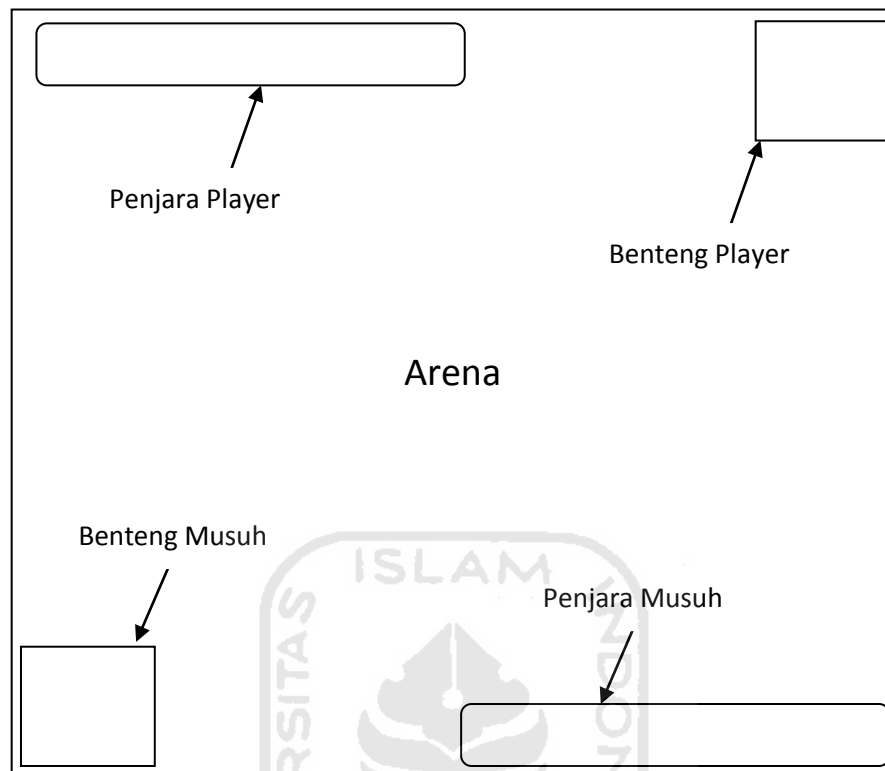
Gambar 3.6 Tampilan Home

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.



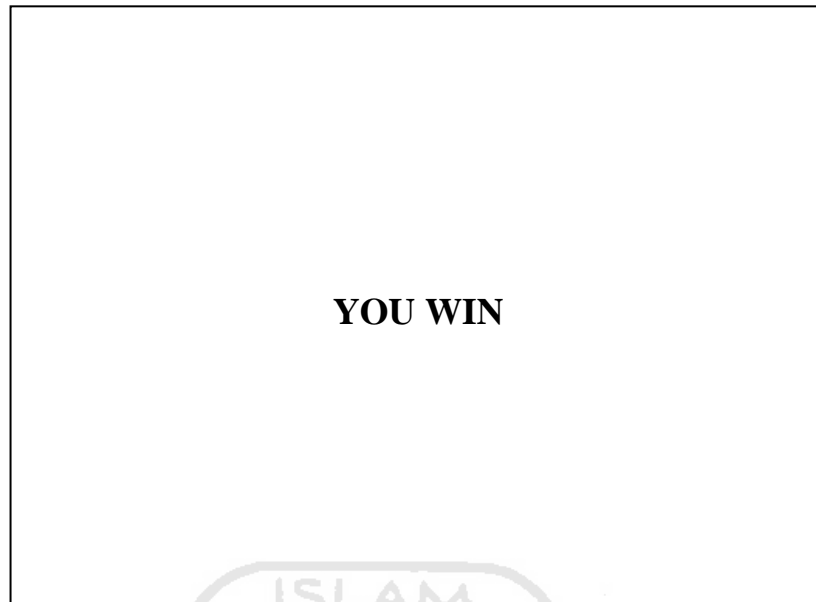
Gambar 3.7 Tampilan Pilihan kesulitan

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.

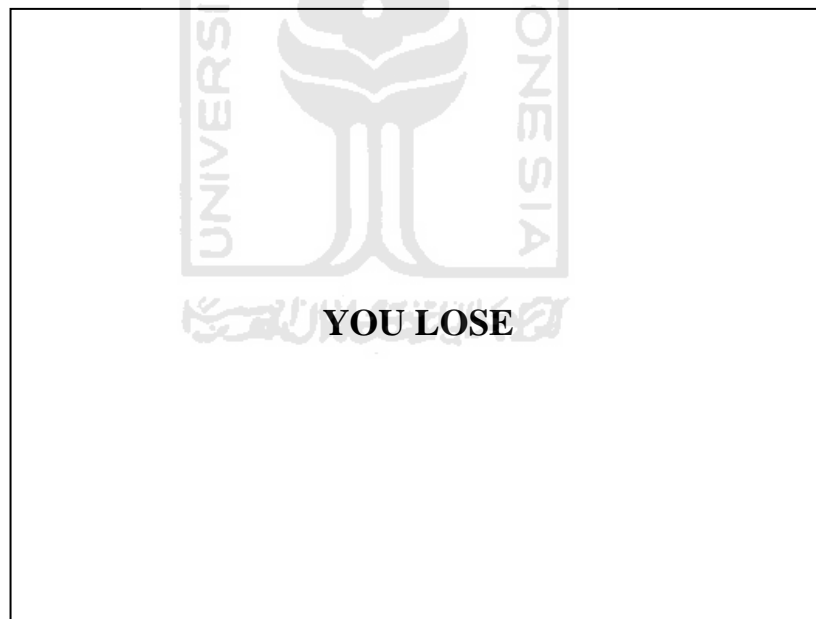


Gambar 3.8 Tampilan Arena Bentengan

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

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Pada bab ini akan dijelaskan bagaimana sistem yang telah dirancang sebelumnya kemudian di implementasikan menjadi sebuah aplikasi utuh. Aplikasi yang dikembangkan ini digunakan untuk mengenalkan permainan tradisional Bentengan.

Dalam implementasi sistem ini terdiri dari beberapa tahap. Yaitu :

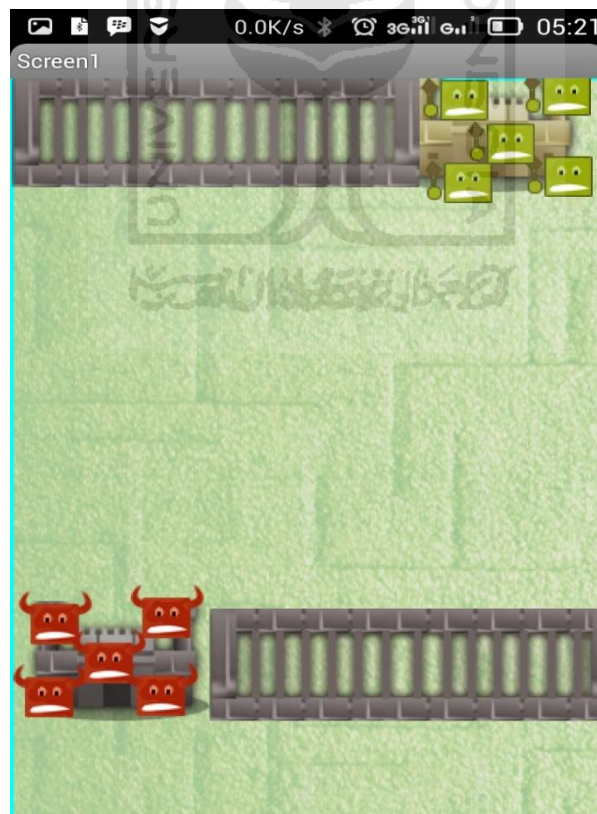
1. Implementasi pada aplikasi app inventor sebagai aplikasi utama dalam mengembangkan game tersebut.
2. Menguji program pada perangkat *smartphone* yang menggunakan sistem operasi android.
3. Menganalisis jalannya program.
4. Memperbaiki program jika terjadi kesalahan dalam berjalannya program.

4.1.1 Interface Permainan

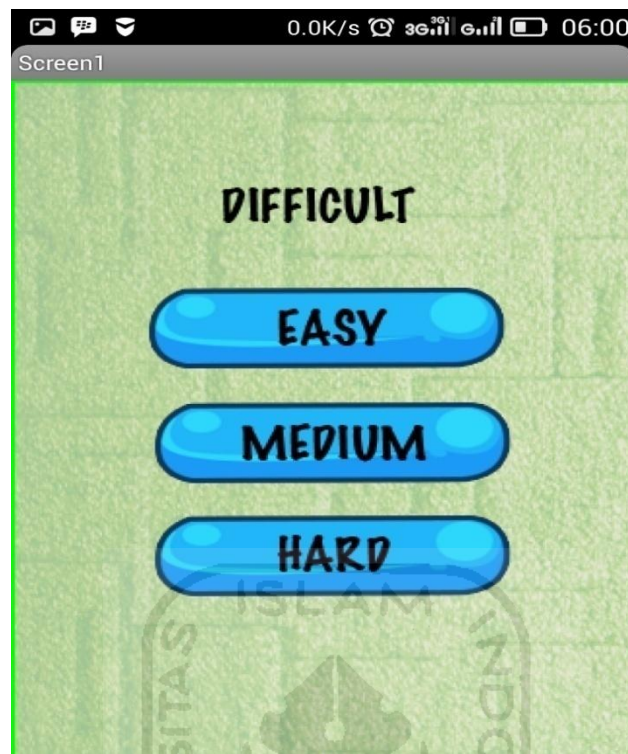
Pada *interface* permainan menunjukkan hasil dari tampilan aplikasi game bentengan yang sudah jadi. *Interface* dari hasil akhir game tersebut berdasarkan dari rancangan tampilan permainan. Untuk *interface* aplikasi game bentengan dapat dilihat pada gambar 4.1, gambar 4.2, gambar 4.3 dan gambar 4.4



Gambar 4.1 Tampilan Home



Gambar 4.2 Tampilan Arena



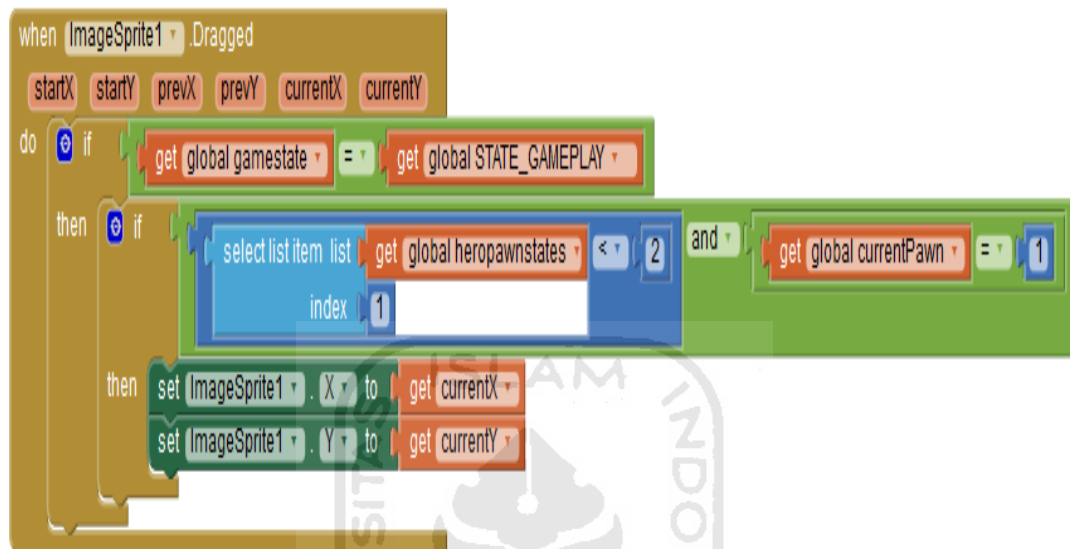
Gambar 4.3 Tampilan pilihan kesulitan

Tampilan pilihan kesulitan atau *Difficult* digunakan untuk mengatur kecepatan gerak dari *enemy*. Penentuan kecepatan gerak dari *enemy* ini saat berada di arena berdasarkan banyak pixel pada layar yang dilewati dalam satuan waktu. Untuk pilihan kesulitan *easy* kecepatan geraknya diseting 13 pixel. Untuk pilihan *medium* kecepatan geraknya diseting 20 pixel. Untuk pilihan kesulitan *hard* kecepatan geraknya diseting 30 pixel.

4.1.2 Implementasi Gerakan Player

Pada aplikasi ini untuk menggerakkan player menggunakan blok dari perintah *when.ImageSprite1.Dragged*. Pada blok ini karakter *ImageSprite1* dapat digerakkan sesuai dengan koordinat X dan Y pada arena. *ImageSprite1* di sini adalah karakter *hero* no 1 dari 5 karakter *hero* yang dimainkan oleh player dan setiap karakter memiliki pengaturan blok sendiri-sendiri. Pada setiap pengaturan gerak pada karakter *hero* 2,3,4 dan 5 tinggal dirubah kondisi dari

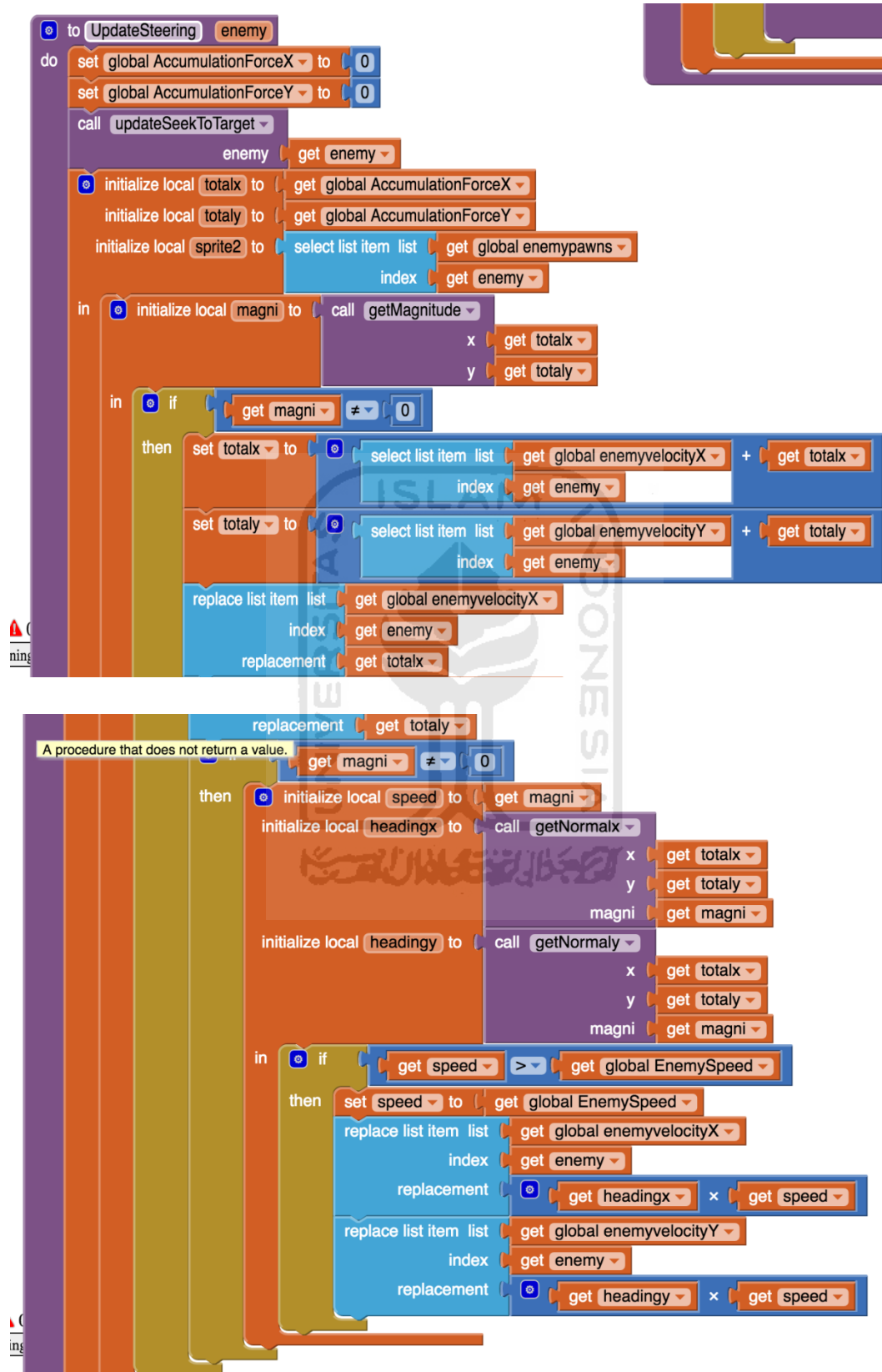
when.ImageSprite1.Dragged menjadi *when.ImageSprite2.Dragged* pada hero 2 , *when.ImageSprite3.Dragged* pada hero 3, *when.ImageSprite4.Dragged* pada hero 4, dan *when.ImageSprite5.Dragged* pada hero 5.



Gambar 4.4 Kode blok *when.ImageSprite1.Dragged*

4.1.3 Implementasi Gerakan Enemy

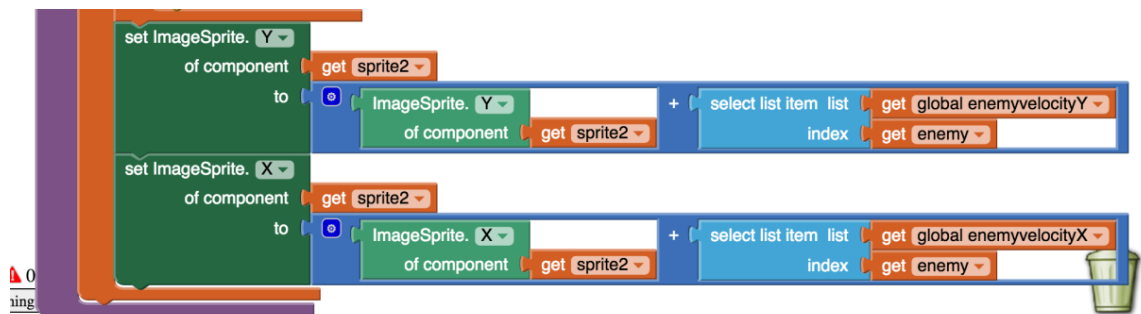
Pada aplikasi ini untuk menggerakkan *enemy* menggunakan *method* dari *to.UpdateSteering.enemy*. Pada *method to.UpdateSteering.enemy* berguna untuk menentukan arah dari gerakan *enemy* berdasarkan koordinat titik x dan y. Untuk menentukan update arah gerakan terdapat pada *method UpdateSeekToTarget*.



The image displays two segments of Scratch code for an enemy's steering logic. The top segment is a procedure named "to UpdateSteering enemy" which initializes global variables for accumulation forces, calls "updateSeekToTarget", and initializes local variables for total force components and a list of enemy pawns. It then enters a loop where it calculates the magnitude of the total force. If the magnitude is non-zero, it iterates through the list of enemy pawns, accumulating their velocity components into the total force. The bottom segment continues the logic by normalizing the total force to determine heading and speed. If the current speed is greater than the global enemy speed, it updates the speed and replaces the corresponding velocity components in the list with the new heading and speed values.

```

to UpdateSteering enemy
do
  set global AccumulationForceX to 0
  set global AccumulationForceY to 0
  call updateSeekToTarget
  enemy
  initialize local totalx to get global AccumulationForceX
  initialize local totaly to get global AccumulationForceY
  initialize local sprite2 to select list item list
  index
  in initialize local magni to call getMagnitude
  x
  y
  in if
  then
    set totalx to select list item list
    index
    set totaly to select list item list
    index
    replace list item list
    index
    replacement
  replacement
  A procedure that does not return a value.
  then
    initialize local speed to get magni
    initialize local headingx to call getNormalx
    x
    y
    magni
    initialize local headingy to call getNormaly
    x
    y
    magni
    in if
    then
      set speed to get global EnemySpeed
      replace list item list
      index
      replacement
      replace list item list
      index
      replacement
  
```

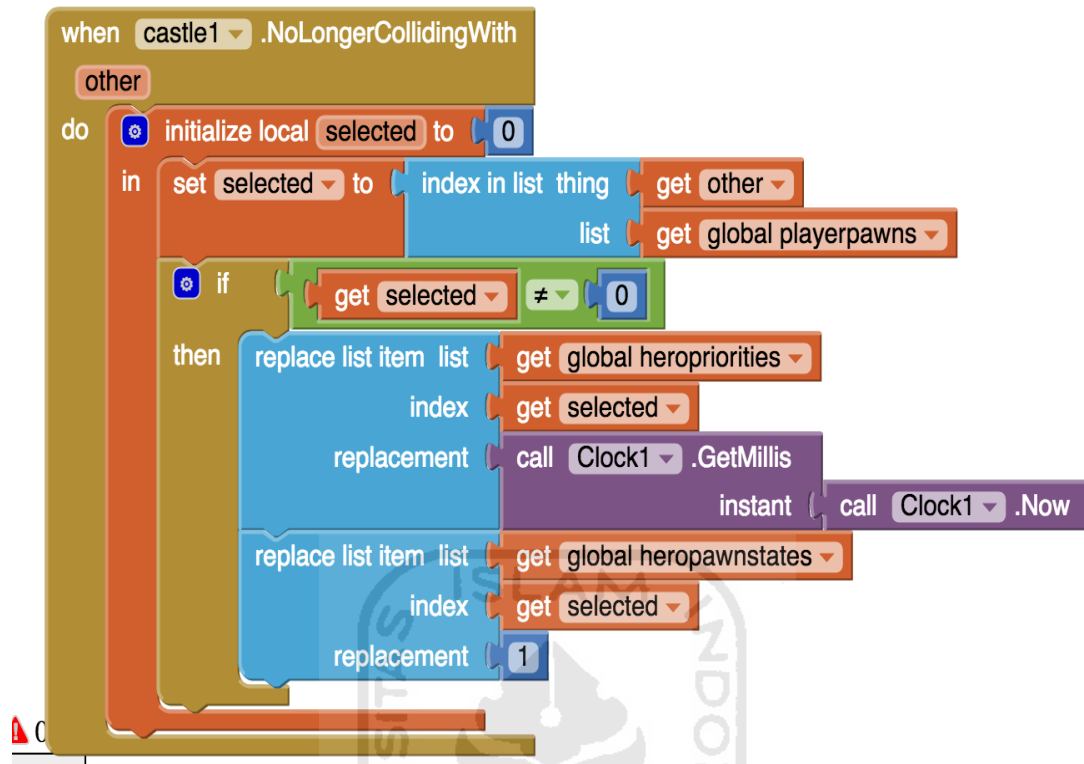


Gambar 4.5 Kode *block to.UpdateSteering.enemy*

4.1.4 Implementasi Prioritas Pemain

Pada arena bentengan ini di bagi menjadi 3 *state* yaitu *state 0*, 1 dan 2. Pada *state 0* merupakan kondisi dimana para karakter *hero* atau juga *enemy* berada pada posisi di dalam benteng atau masih menyentuh benteng. *State 1* adalah kondisi di mana posisi karakter *hero* atau *enemy* keluar dari benteng nya dan masuk ke arena. Pada posisi ini di sebut dengan posisi tempur. sedangkan *State 2* adalah kondisi dimana *hero* atau *enemy* berada di dalam penjara. Untuk penentuan prioritas menurut *state* maka *state 0* prioritas nya lebih tinggi daripada *state 1* dan *state 1* prioritas nya lebih tinggi dari *state 2*. *Method* yang digunakan untuk penentuan *state* tersebut terdapat pada blok *when.castle1.NoLongerCollidingWith*. *method when.castle1.NoLongerCollidingWith* digunakan untuk pemain *hero* sedangkan untuk *enemy* menggunakan blok *when.castle2.NoLongerCollidingWith*

Pada saat *hero* atau *enemy* pada *state 1* atau mode bertempur, pengaturan prioritasnya didasarkan akan waktu. Semakin kecil lama waktu pada kondisi *state 1* maka prioritas nya semakin besar.

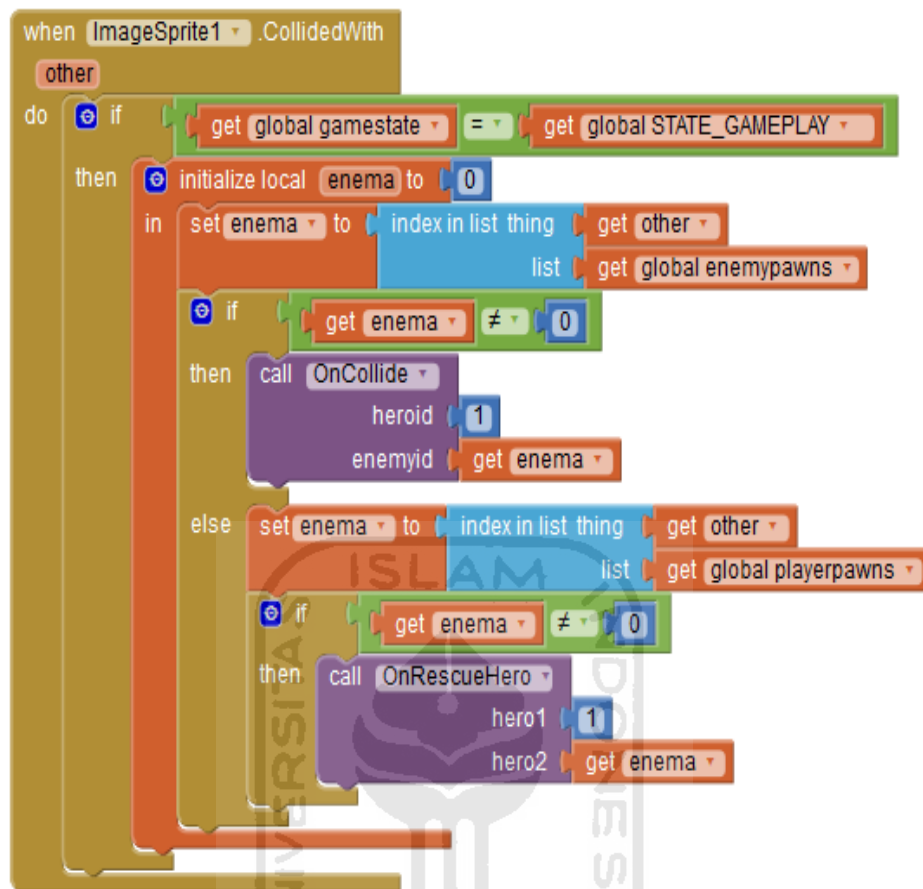


Gambar 4.6 Kode block *when.castle1.NoLongerCollidingWith*

4.1.5 Implementasi Tabarakan *Hero*

Pada permainan bentengan ada kondisi dimana saat *hero* bersentuhan dengan *hero* lain. Ketentuan deteksi tabrakan ini dibagi menjadi 2 kondisi yaitu tabrakan dengan kawan atau tabrakan dengan *enemy*. Jika kondisi nya tabrakan dengan kawan maka tidak terjadi apa-apa namun jika tabrakan dengan *enemy* maka akan di cek status prioritas nya siapa yang lebih tinggi seperti pada implementasi prioritas *hero* untuk menentukan siapa yang masuk ke penjara (*Prison*).

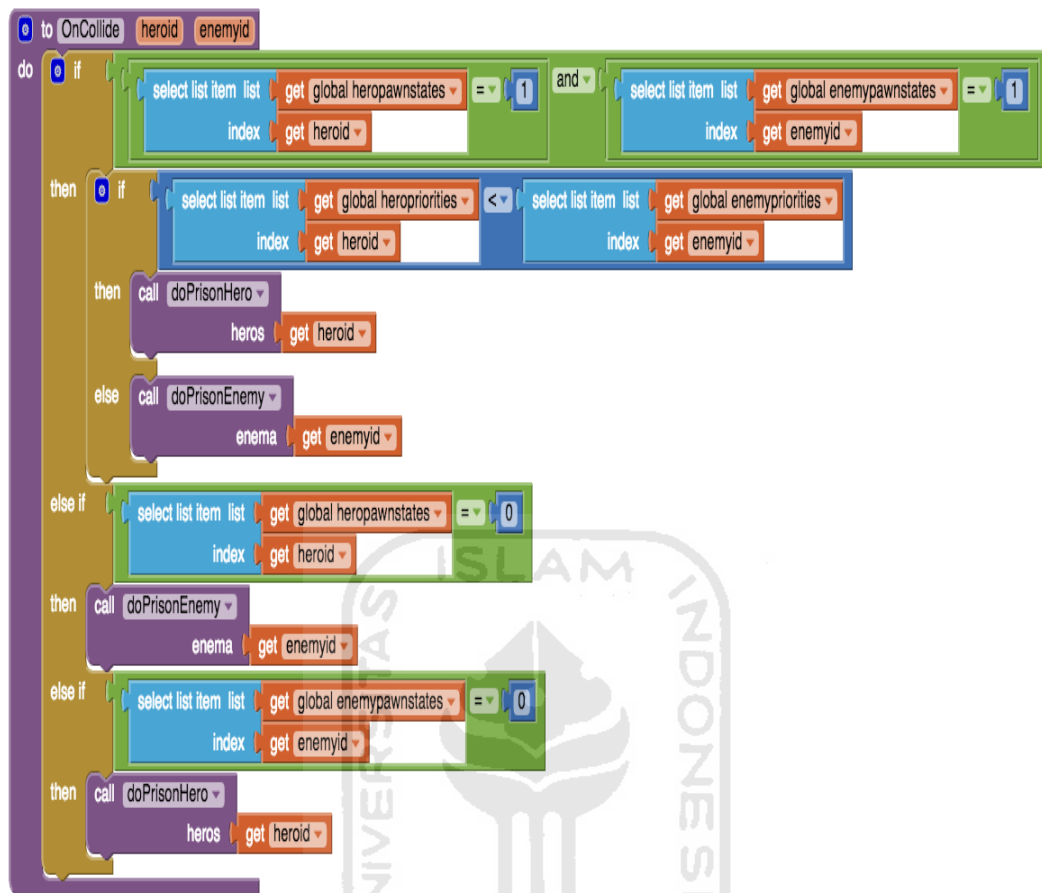
Pada aplikasi ini pola deteksi di atur pada blok *Collidedwith*. Sebagai contoh seperti pada gambar 4.7 untuk karakter *hero 1 (ImageSprite1)* menjadi *when.ImageSprite1.Collidedwith*. Karena di tim player ada 5 pemain, jadi untuk setiap karakter *hero* yang lain mempunyai pengaturan blok sendiri tinggal merubah karakter *ImageSprite1* menjadi *ImageSprit2* dan seterusnya.



Gambar 4.7 Kode blok *when.ImageSprite1.Collidedwith*

4.1.6 Implementasi *Oncollide*

Merupakan sebuah method yang digunakan untuk mengecek prioritas pemain mana yang lebih tinggi antara *hero* dan *enemy* saat mereka bertabrakan. Penentuan prioritas ini di gunakan untuk menentukan pihak mana yang di penjara *enemy* atau *hero*. *Method* yang digunakan dalam penentuan prioritas saat tabrakan terdapat pada gambar 4.8.



Gambar 4.8 Kode blok *to.OnCollide.heroid.enemyid*

4.1.7 Implementasi Memasukkan Penjara

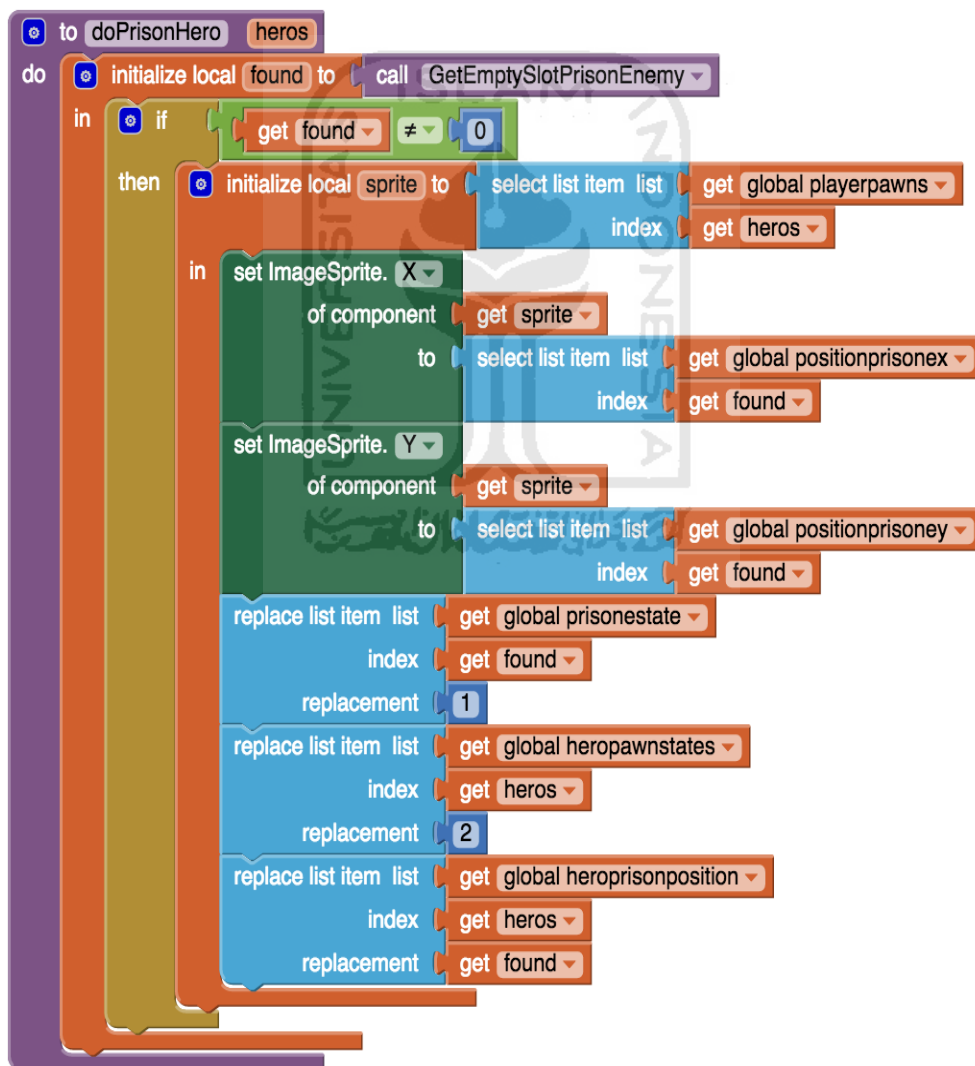
Pada permainan bentengan ada kondisi di mana *hero* atau *enemy* harus masuk ke penjara. Ada 2 kondisi yang menentukan untuk masuk ke dalam penjara yaitu, jika saat *hero* dan *enemy* bertabrakan dan prioritas *hero* lebih tinggi dari pada *enemy* maka *enemy* masuk ke dalam *PrisonEnemy*. Kemudian jika prioritas *enemy* lebih tinggi daripada *hero* maka *hero* masuk ke dalam *PrisonHero*.

4.1.7.1 Prison Hero

Pada aplikasi *Prison Hero* yaitu suatu kondisi dimana *hero* bertabrakan (*Collide*) dengan *enemy* yang memiliki prioritas lebih tinggi dari *hero* maka *hero* tersebut dimasukkan ke dalam penjara. Pada blok di tulis berupa kondisi

$global_heropriorities < global_enemypriorities$ maka *call doPrisonHero*. *doPrisonHero* merupakan sebuah blok perintah untuk menentukan posisi atau koordinat *Hero* di penjara dan merubah *state hero* dari angka 1 yang merupakan kondisi di mana *hero* keluar dari markas (posisi bertempur) menjadi angka 2 yang merupakan kondisi di saat dalam penjara.

Untuk penentuan posisi *hero* saat dipenjara menggunakan *method to.doPrisonHero.heros*. Pada *method* tersebut digunakan untuk menentukan koordinat *hero* di penjara berdasarkan koordinat x dan y.

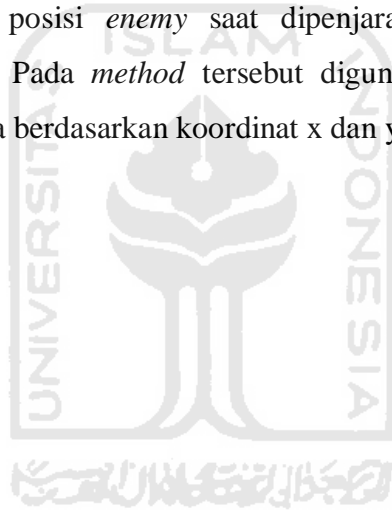


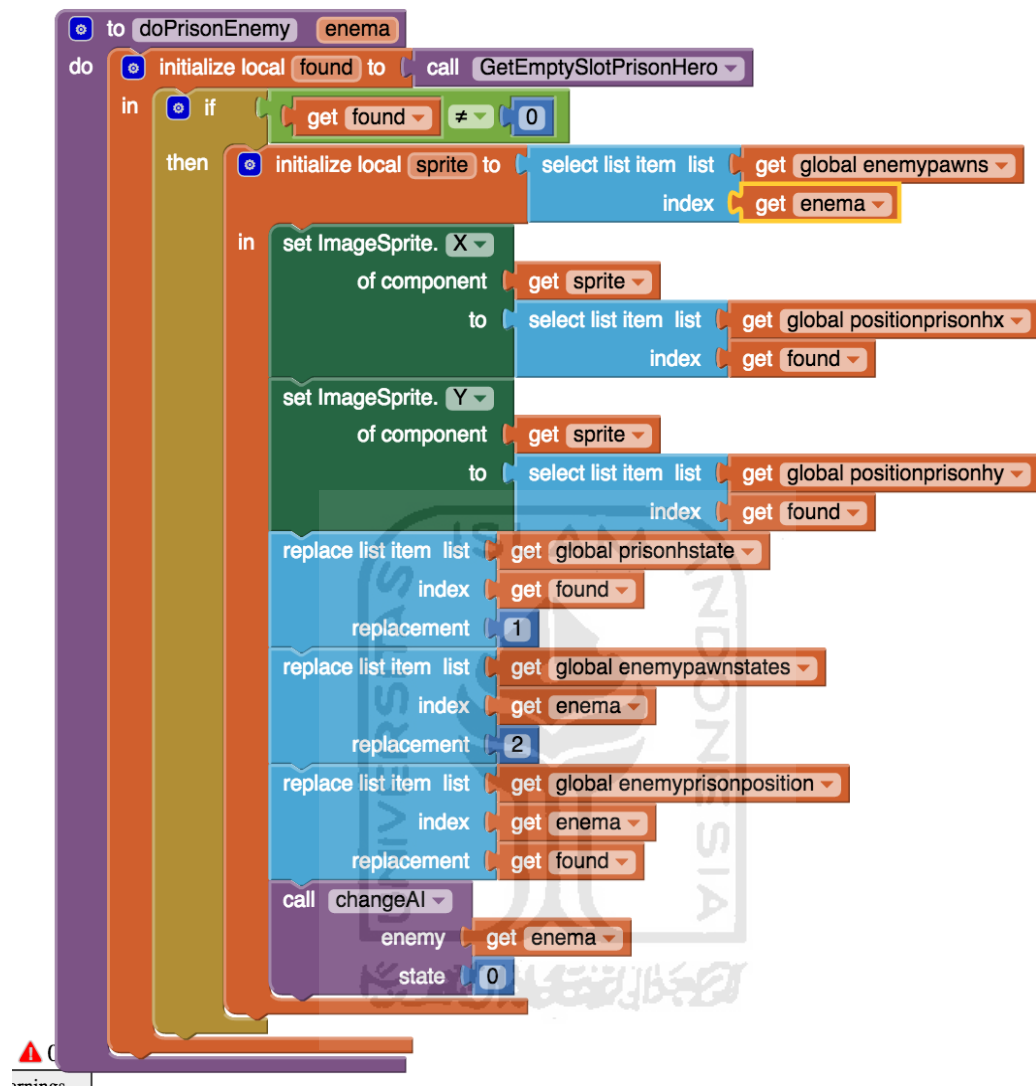
Gambar 4.9 Kode blok *to.doPrisonHero.heros*

4.1.7.2 Prison Enemy

Pada aplikasi *Prison Enemy* yaitu suatu kondisi dimana *hero* bertabrakan (*Collide*) dengan *enemy* dan prioritas *hero* lebih tinggi daripada *enemy* maka *enemy* tersebut dimasukkan ke dalam penjara. Pada blok di tulis berupa kondisi $global_heropriorities > global_enemypriorities$ maka *call doPrisonEnemy*. *doPrisonEnemy* merupakan sebuah blok perintah untuk menentukan posisi atau koordinat *enemy* di penjara dan merubah *state enemy* dari angka 1 yang merupakan kondisi di mana *enemy* keluar dari markas (posisi bertempur) menjadi angka 2 yang merupakan kondisi di saat dalam penjara.

Untuk penentuan posisi *enemy* saat dipenjara menggunakan *method to.doPrisonEnemy.enema*. Pada *method* tersebut digunakan untuk menentukan koordinat *enemy* di penjara berdasarkan koordinat x dan y.





Gambar 4.10 Kode blok *to.doPrisonEnemy.enema*

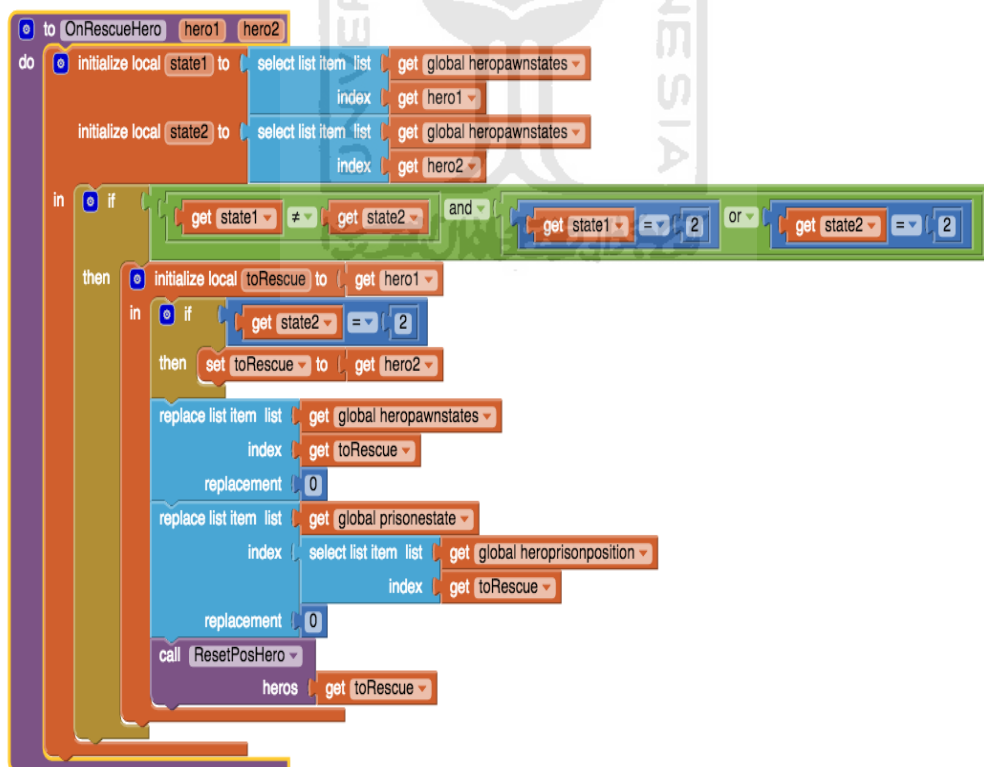
4.1.8 Implementasi Keluar Penjara

Keluar dari penjara merupakan sebuah kondisi dimana *hero* membebaskan *hero* lain yang tertangkap atau di penjara oleh *enemy* dengan bersentuhan dengan *hero* yang dipenjara tanpa tertangkap oleh *enemy*. Begitu juga sebaliknya dimana *enemy* membebaskan *enemy* lain yang telah tertangkap oleh pemain.

4.1.8.1 Pada Hero

Pada aplikasi ini *hero* berusaha membebaskan *hero* lain yang telah tertangkap oleh *enemy* dan di penjara di karena *hero* telah bertabrakan atau *collide* dengan *enemy* yang prioritasnya lebih tinggi dari pemain. Cara membebaskan *hero* yang dipenjara adalah dengan cara menyentuh *hero* tersebut dengan *hero* yang masih bebas dengan syarat *hero* tersebut dalam kondisi *state* 1 dan belum tersentuh oleh *enemy* dengan prioritas yang lebih tinggi.

Method yang digunakan untuk membebaskan *hero* dari penjara ada pada blok *to.onRescueHero.hero1.hero2*. Setelah di bebaskan, maka *hero* yang di penjara tersebut posisi nya di reset kembali ke posisi awal di markas. *State* awal *hero* waktu di tangkap yang memiliki *state* 2 kemudian di rubah menjadi 0, karena kembali ke markas. Untuk lebih detil tentang *method* *to.onRescueHero.hero1.hero2* dapat dilihat pada gambar 4.11.

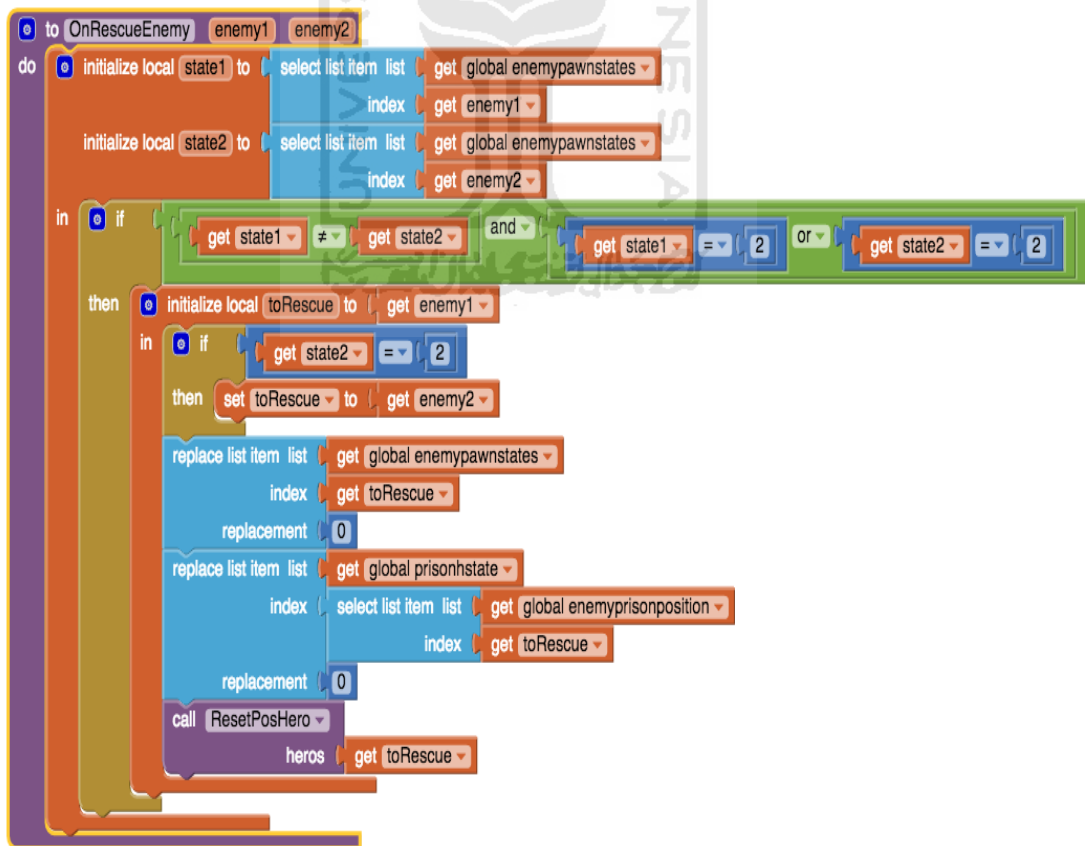


Gambar 4.11 Kode blok *to.OnRescueHero.hero1.hero2*

4.1.8.2 Pada Enemy

Pada aplikasi ini tim *enemy* membebaskan *enemy* yang tertangkap oleh *hero* milik kita dan di penjara di *PrisonEnemy*. Cara membebaskan *enemy* yang dipenjara adalah dengan cara menyentuh *enemy* tersebut dengan *enemy* yang masih bebas dengan syarat karakter *enemy* tersebut dalam kondisi *state* 1 dan belum tersentuh oleh *hero* dengan prioritas yang lebih tinggi.

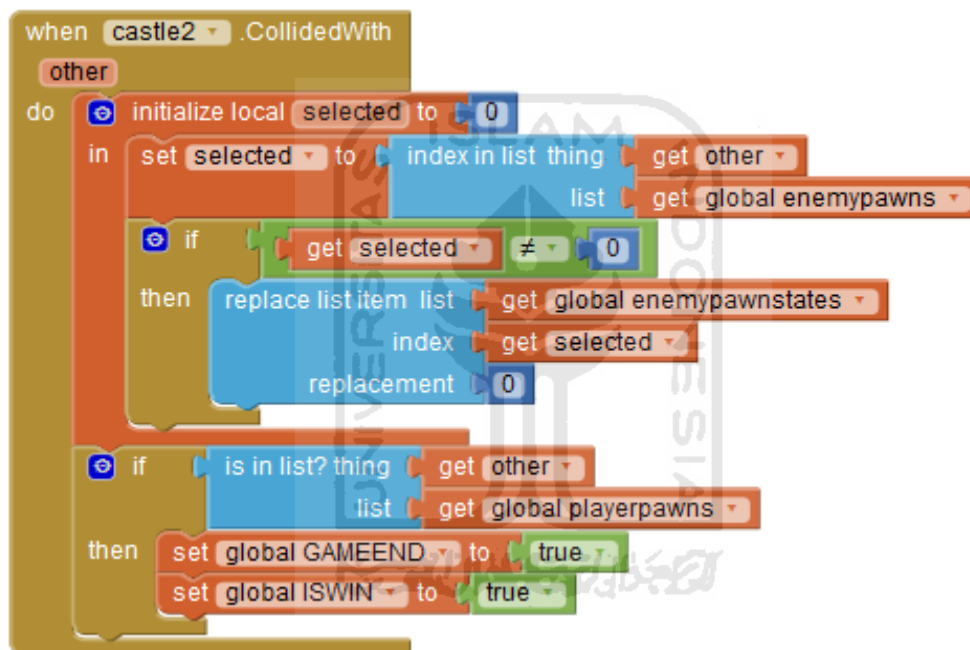
Method yang digunakan untuk membebaskan *enemy* dari penjara ada pada blok *to.onRescueEnemy.enemy1.enemy2*. Setelah di bebaskan, maka *enemy* yang di penjara tersebut posisi nya di reset dan kembali ke posisi awal di markas. *State* awal *enemy* waktu di tangkap yang memiliki *state* 2 kemudian di rubah menjadi 0, karena kembali ke markas. Untuk lebih detail tentang *method* *to.onRescueEnemy.enemy1.enemy2* dapat dilihat pada gambar 4.13.



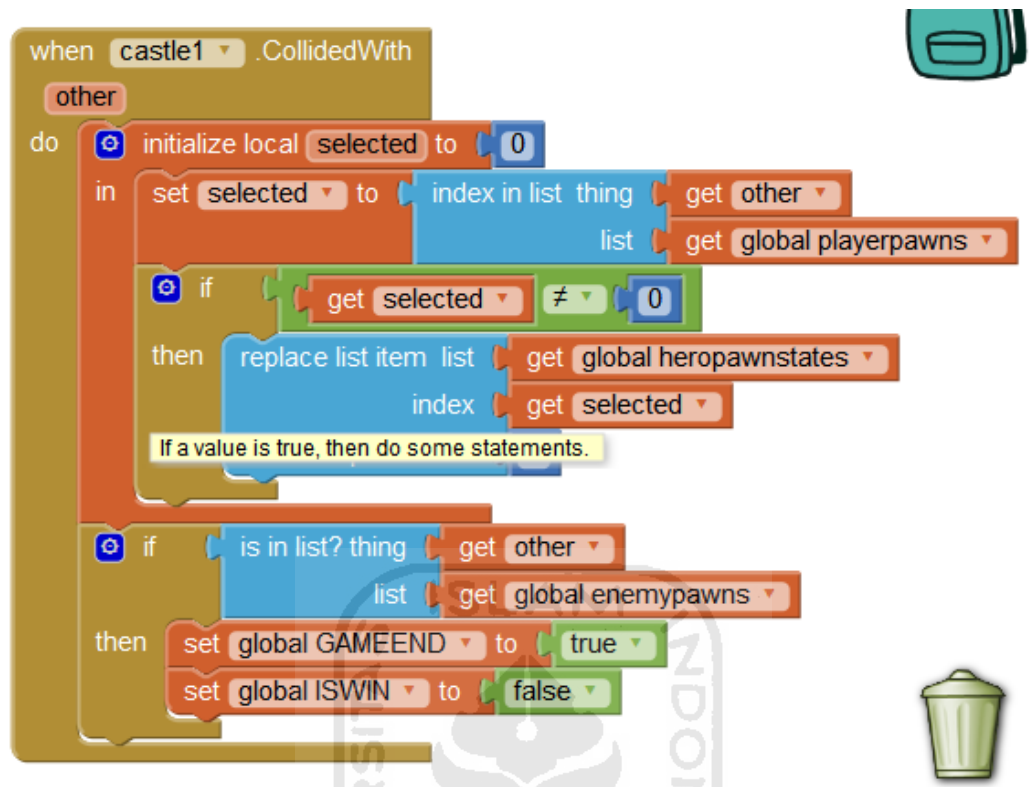
Gambar 4.12 Kode blok *to.OnRescueEnemy.enemy1.enemy2*

4.1.9 Kondisi Menang dan Kalah

Pada permainan benteng terdapat kondisi menang dan juga kalah. *Method* yang digunakan ada 2 yaitu kondisi saat *Hero* menyentuh benteng *enemy* maka status nya menang. Sedangkan saat *enemy* menyentuh benteng *hero* maka status kalah. Untuk kondisi *hero* menyentuh benteng *enemy* menggunakan blok *when.castle2.Collidedwith* . Sedangkan untuk kondisi *enemy* menyentuh benteng *hero* menggunakan *method* blok *when.castle1.CollidedWith*.



Gambar 4.13 Status menang dengan blok *when.castle2.collidedwith*



Gambar 4.14 Status kalah dengan blok *when.castle1.collidedwith*

4.2 Pengujian Aplikasi

Pengujian aplikasi game Bentengan ini menggunakan *Smartphone* Android dengan merek Lenovo bertipe A859. Untuk tampilan menu-menu aplikasi dapat dilihat pada lampiran yang terdiri dari tampilan Home atau menu utama, Tampilan pilahan *Sound* posisi Off dsan On, tampilan pemilihan tingkat kesulitan, tampilan saat menang dan juga tampilan saat kalah.

Tampilan menu utama dapat dilihat pada gambar L.1, pada menu utama pemain dihadirkan beberapa pilihan menu. Pilihan menu tersebut antara lain tombol *Play* untuk memulai permainan , tombol *Sound On/Off* untuk menghidupkan atau mematikan suara *backsound*, tombol *Difficult* untuk masuk ke dalam pemilihan tingkat kesulitan dan juga terdapat tombol *Exit*.

Tampilan saat tombol *Play* di pencet dan masuk ke dalam arena permainan dapat dilihat pada gambar L.2. Pada arena terdapat 5 pemain *Hero* dan 5 pemain

Enemy. Disini player dapat menggerakkan pemain *hero* untuk mengalahkan *enemy*. Player menggerakkan pemain *hero* yang bertujuan untuk menyentuh benteng *enemy* tanpa tertabarak pemain *enemy*.

Tampilan menu pemilihan *Difficult* dapat dilihat pada gambar L.3. Pada tampilan menu terdapat 3 pilihan tingkat kesulitan yaitu *easy*, *medium* dan *hard*. Tingkat kesulitan yang dipilih berpengaruh terhadap tingkat kecepatan gerak *enemy*.

Saat player dan enemy berada di arena seperti pada gambar L4. Tampilan pada saat pemain enemy dan player berada di penjara seperti pada gambar L.5 Saat player menang dan berhasil mengalahkan *enemy* maka akan muncul status menang seperti pada gambar L.6 dan saat player kalah terhadap *enemy* maka akan muncul status kalah seperti pada gambar L.7.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan uraian dan pembahasan yang telah dilakukan pada bab-bab sebelum nya dan juga setelah melalui tahap perancangan dan implementasi sistem, maka dapat diambil beberapa kesimpulan yaitu :

1. Game Bentengan memiliki 3 pilihan kesulitan yaitu *easy, medium, hard* yang dapat disesuaikan dengan keinginan pemain.
2. Game ini hanya dimainkan sendiri atau *single player* melawan *enemy* atau komputer.
3. Setiap player baik user dan *enemy* memiliki 5 pemain pada awal permainan.

5.2 Saran

Berdasarkan pengujian dan juga setelah melalui proses pengembangan sistem ini maka terdapat beberapa saran yang dipandang perlu dalam proses pengembangan berikut nya.

1. Ditambahkan *timer* untuk melihat berapa lama waktu permainan berlangsung.
2. Game Bentengan ini perlu dikembangkan agar kedepannya dapat dimainkan secara bersamaan secara langsung oleh 2 player.
3. Penambahan fitur *highscore* untuk menentukan siapa yang memperoleh *score tertinggi*.

DAFTAR PUSTAKA

- Prasetito, Ahmad Fajar. *APPINVENTOR UNTUK PEMULA !!!*.
<http://kambing.ui.ac.id/onnopurbo/ebook/ebook-SU2013/SuryaUniv-Appinventor-bagi-pemula-by-Ahmad-Fajar-Prasetiyo.pdf> . Diakses 30 Juni 2016.
- BYP. 2014. *Membuat Game Android Dengan APP Inventor*.
<http://bypyudha.blogspot.co.id/2014/11/tni-au-ingin-kewenangan-menyidik.html>. Diakses 2 Juli 2016.
- Dirgantara, Danang. *Ebook Membuat Aplikasi Android di MIT App Inventor (Step by Step Tutorial)*. <http://simponydaun.blogspot.co.id/2015/12/cara-membuat-game-dengan-app-inventor.html>. Diakses 2 Juli 2016.
- Wijayanto, Tri. 2015. *GAME TRADISONAL GOBAK SODOR BERBASIS ANDROID*. Skripsi, Teknik Informatika, STIMIK AKAKOM, Yogyakarta.
- Cioray, Dede. 2014. *Pengenalan “Unified Modeling Language/UML” Bagian 1*.
<http://ebook.dede-gunawan.web.id/2014/12/ebook-uml-bahasa-indonesia.html>. Diakses 20 Juli 2016.
- Kemkominfo: *Pengguna Internet di Indonesia Capai 82 Juta*, 2014,
https://kominfo.go.id/content/detail/3980/kemkominfo-pengguna-internet-di-indonesia-capai-82-juta/0/berita_satker. Diakses 20 Juli 2016.
- Alifary, Krisan. 2016, *Game sebagai sarana edukasi*,
<https://krisanalifari.wordpress.com/2016/01/16/game-sebagai-sarana-edukasi/>. Diakses 20 Juli 2016.

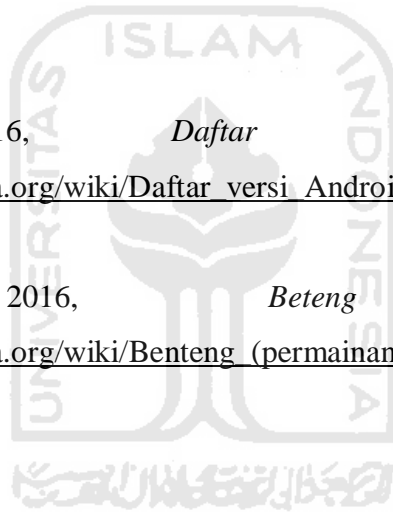
Putra, Tridea, 2016, *Klasifikasi Game (Pengantar Teknologi Game)*
<http://trideaps.blogspot.co.id/2016/04/klasifikasi-game-pengantar-teknologi.html>. Diakses 16 Juli 2016.

Imansaiki,2012,*Macam-macam Genre/Jenis Dalam Game*,
<http://imansaiki.blogspot.co.id/2012/03/macam-macam-genre-jenis-dalam-games.html>. Diakses 20 Juli 2016.

Wikipedia1, 2016, *Sistem Operasi Android*,,
[https://id.wikipedia.org/wiki/Android_\(sistem_operasi\)](https://id.wikipedia.org/wiki/Android_(sistem_operasi)). Diakses 18 Juli 2016.

Wikipedia2, 2016, *Daftar Versi Android*,
https://id.wikipedia.org/wiki/Daftar_versi_Android. Diakses 18 Juli 2016.

Wikipedia3, 2016, *Beteng (Permainan)*,
[https://id.wikipedia.org/wiki/Benteng_\(permainan\)](https://id.wikipedia.org/wiki/Benteng_(permainan)). Diakses 18 Juli 2016.



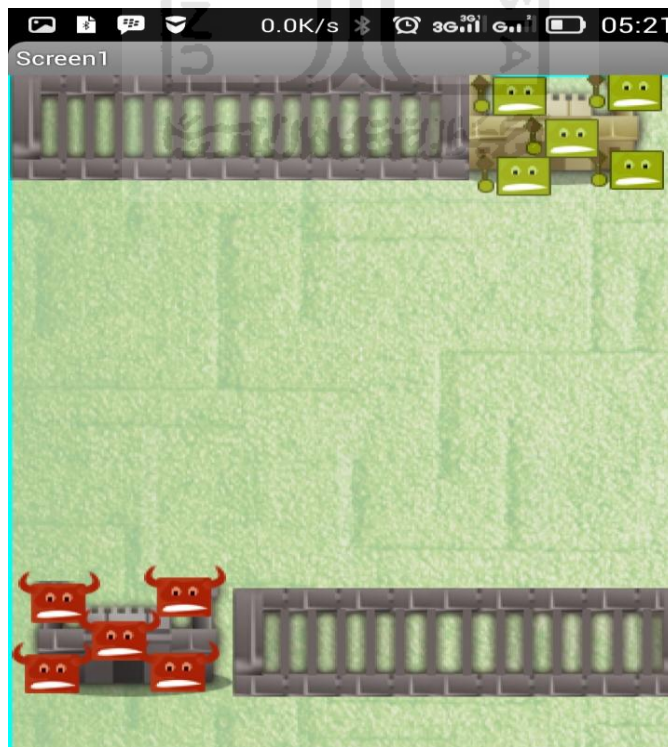
LAMPIRAN

Pengujian Aplikasi

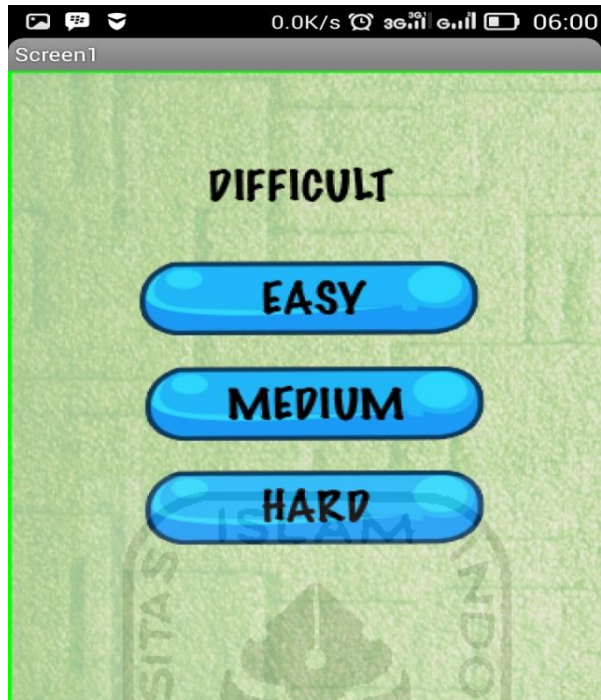




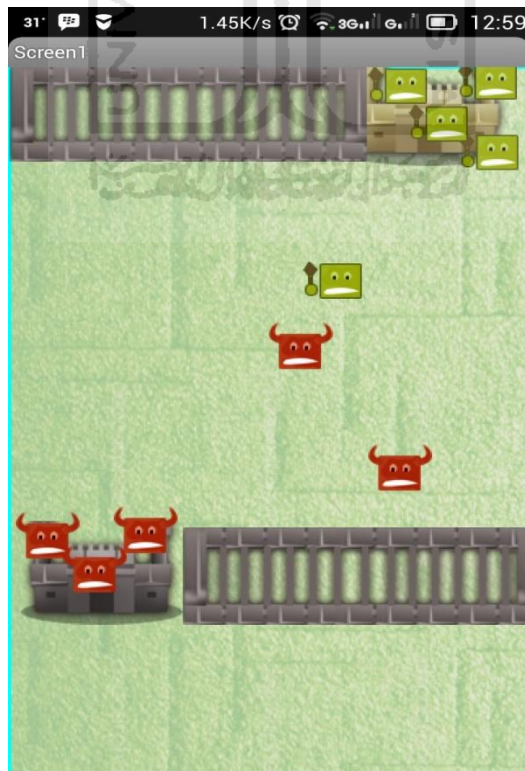
Gambar L.1 Tampilan Menu Utama



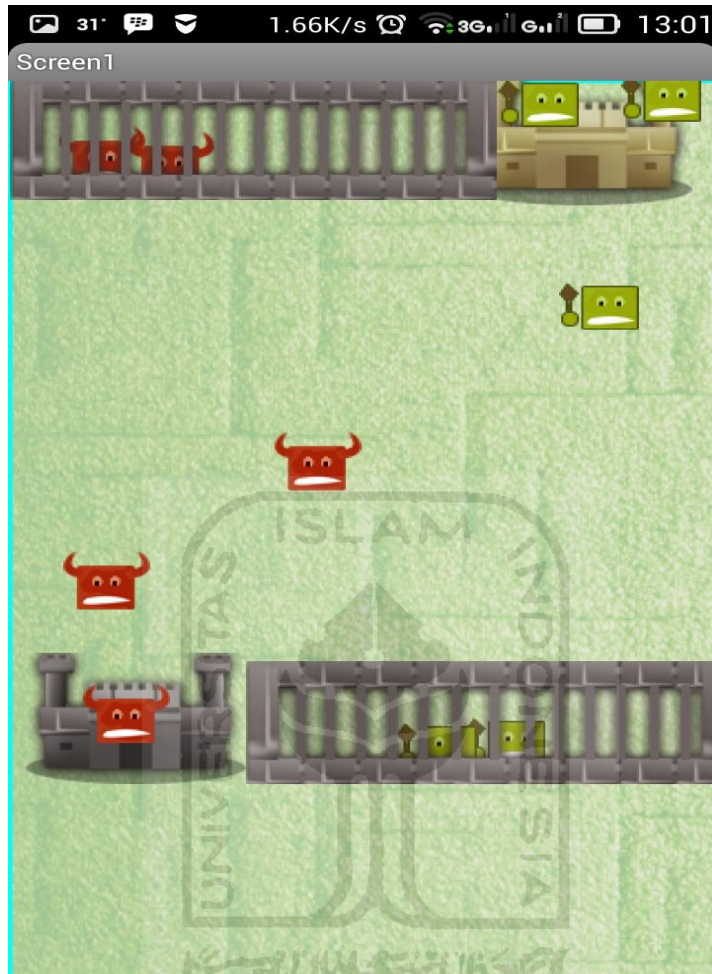
Gambar L.2 Tampilan Arena Permainan



Gambar L.3 Tampilan Pilihan Tingkat Kesulitan



Gambar L4 Tampilan saat pemain player dan enemy maju



Gambar L5 Tampilan saat pemain player dan enemy di penjara



Gambar L.6 Tampilan Saat Menang



Gambar L7 Tampilan Saat Kalah