APLIKASI PERMAINAN TRADISIONAL INDONESIA "BENTENGAN" BERBASIS ANDROID

Agusta Wahyu Sputra
Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia
Jl.Kaliurang KM 14 Yogyakarta
Telp. (0274) 898444
E-mail: agusta.wahyu@gmail.com

ABSTRAK

Pada saat ini teknologi perangkat telekomunikasi berkembang dengan sangat cepat khususnya telpon pintar atau smartphone. Sistem operasi yang paling popular digunakan pada smartphone saat ini adalah sistem android. Dengan munculnya smartphone 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 diterapakan 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.

Kata kunci : smartphone. Android, game android, gadget

1. PENDAHULUAN

Seiring dengan berjalannya waktu teknologi berkembang dengan pesat nya 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 telpon maupun sms saja tapi telah menjadi sebuah komputer portable yang biasa disebut dengan *smartphone*.

Dalam kurun waktu beberapa tahun terakhir, perkembangan teknologi informasi terutama dalan 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 hingga 82 juta pada tahun 2014. (Teknojurnal,2011). Perkembangan teknologi yang semakin canggih ini disertai dengan munculnya sistem operasi yang terbaru untuk mengimbanginya. 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 dirumah 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. Dalam tugas akhir ini saya membuat 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.

2. LANDASAN TEORI

2.1 *Android*

Android adalah sistem operasi denga sistem tebuka (*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.

Dalam perkembangan hingga saat ini Android memiliki beberapa versi Android, berikut versi-versi Android dari pertama rilis hingga sekarang yaitu versi 1.0 dan 1.1, versi 1.2-1.5 (cupcake), versi 1.6 (donut), versi 2.0-2.1 (éclair), versi 2.2-2.2.3 (froyo) dan lain-lain. Sedangkan versi yang terbaru adalah versi 6.0 (marshmallow).

2.2 Game

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.
Dalam game sendiri terdapat beberapa tipe permainan atau sering disebut *genre*, antara lain adalah :

1. *Role Play Game (RPG)*
2. *Strategy*
3. *Simulation*
4. *Arcade*
5. *Adventure*
6. *Shooter games*
7. *Racing game*
8. *Fighting game*
9. *Board game*
10. *Sport*

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 sampi 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 sebagaia 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.

2.4 *App Inventor*

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 kira bisa membuat aplikasi tanpa mengedit / mengetik kode sedikitpun, hanya tinggal *drag & drop* symbol – symbol perintah dan fungsi.

3. ANALISIS SISTEM

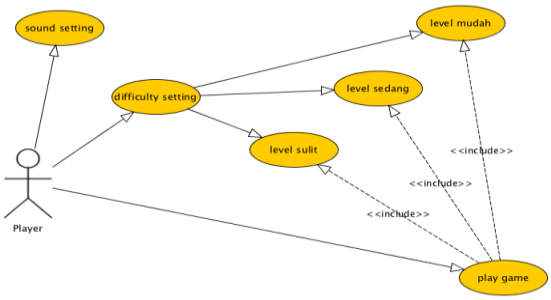
3.1 Gamabaran Umum Sistem

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

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

3.2 *Use Case Diagram*

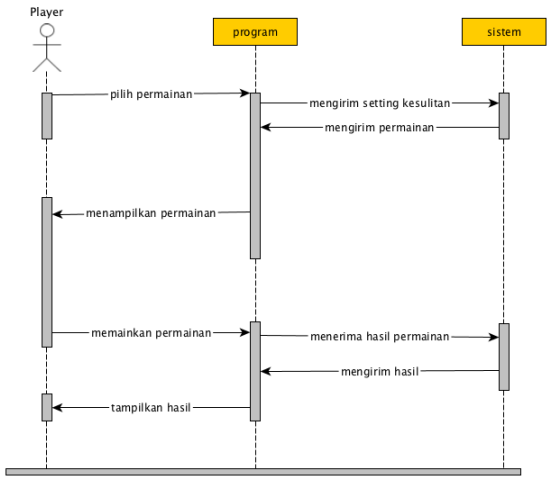
Use Case Diagram digunakan untuk menggambar apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamat luar. *Use Case* digunakan untuk menggambarkan fungsionalitas dari sistem yang dikembangkan. Dan juga mempresentasikan interaksi antara pengguna dengan sistem.



Gambar 1 *Use Case Diagram*

3.3 *Sequence Diagram*

Sequence diagram merupakan salah satu diagram interaksi yang menjelaskan bagaiaman 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

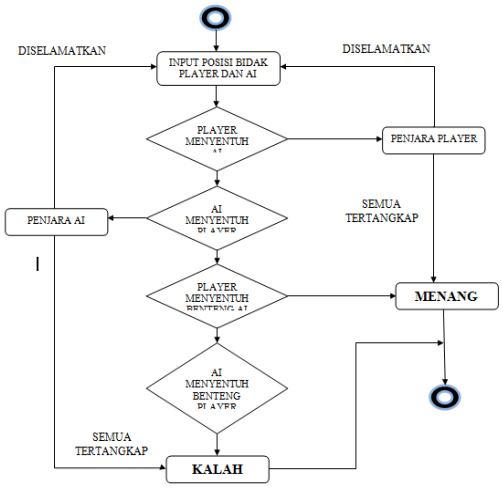


Gambar 2 *Sequence Diagram*

3.4 *Actifity diagram*

Actifity diagram atau diagram aktifitas adalah sebuah diagram yang menggambarkan aktifitas-aktifitas dalam sistem yang sedang dibuat. Beberapa

aktifitas itu adalah seperti awal setiap permainan, pilihan yang terjadi, dan kondisi berakhirnya permainan baik menang ataupun kalah. Pada diagram aktifitas juga dapat menggambarkar aktifitas yang lebih dari satu proses (parallel).

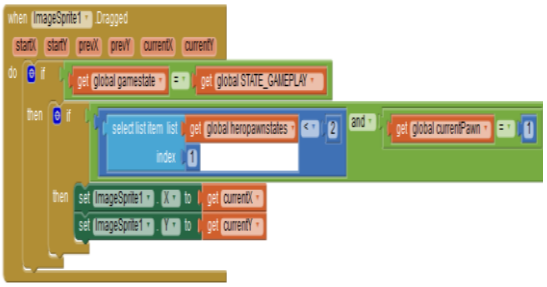


Gambar.3 Diagram Aktifitas

4. IMPLEMENTASI

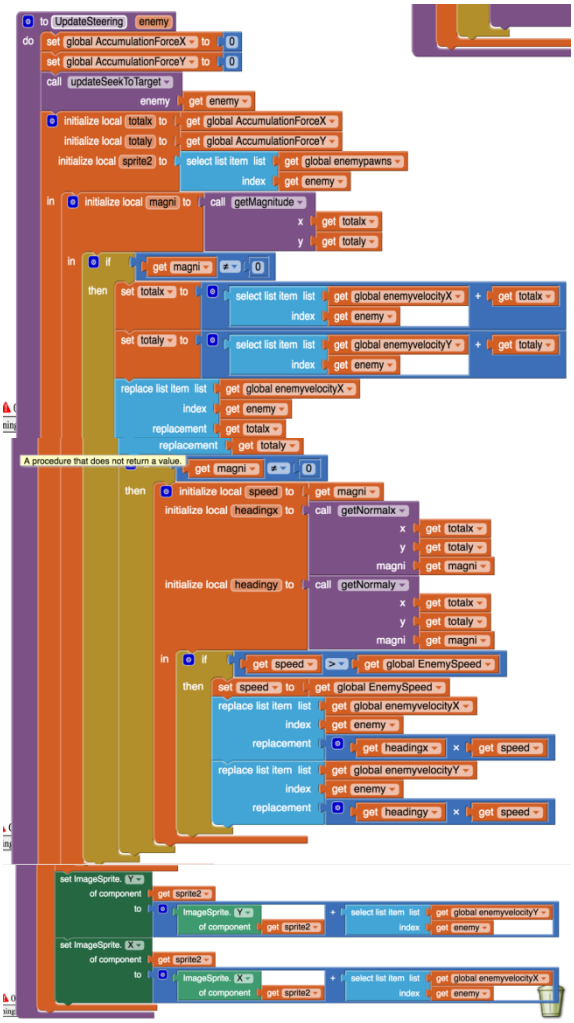
Hasil implementasi dari perancangan yang telah dibuat, pada implementasi ini dijelaskan bagaimana sistem yang telah dirancang sebelum nya kemudian di implementasikan menjadi sebuah aplikasi utuh. Aplikasi yang di kembangkan ini digunakan untuk mengenalkan permainan tradisional Bentengan.

Gambar merupakan blok *method* yang 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.



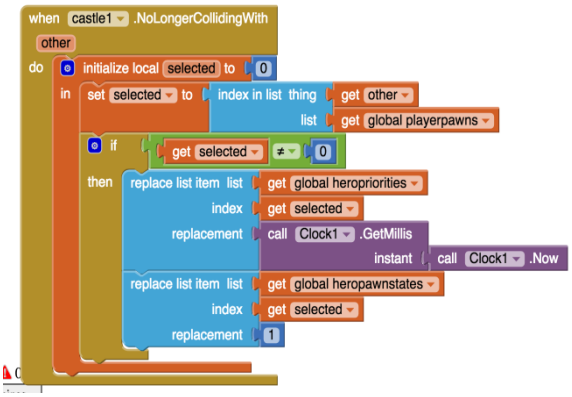
Gambar 4 Kode blok *when.ImageSprite1.Dragged*

Gambar 6 merupkan blok *method* 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.



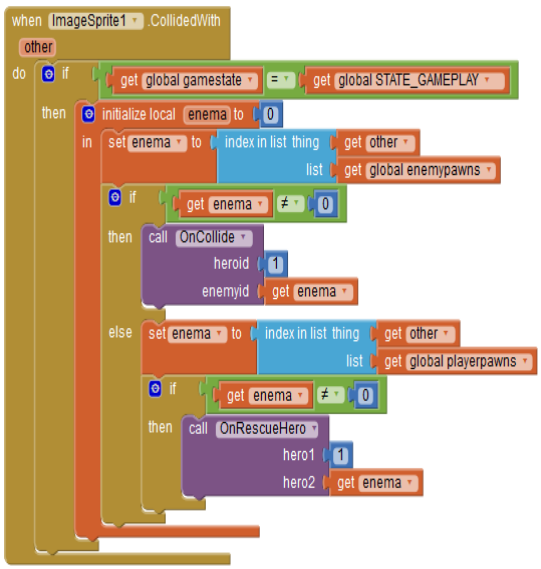
Gambar 5 Kode *block to.UpdateSteering.enemy*

Gambar 7 adalah *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.*



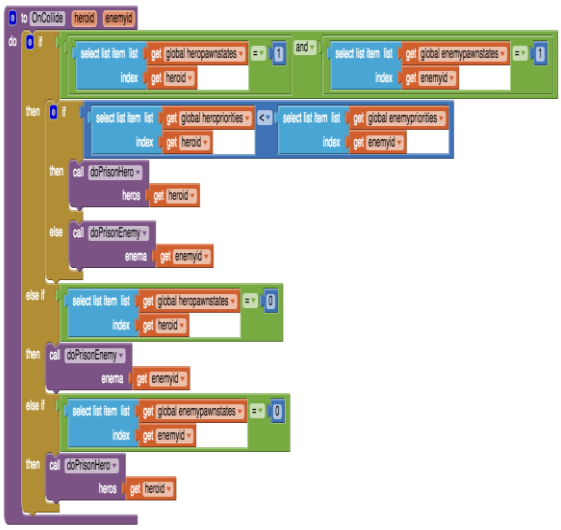
Gambar 6 Kode blok *when.castle1.NoLonger CollidingWith*

Untuk pola deteksi di atur pada blok *Collidedwith*. Sebagai contoh seperti pada gambar 8 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.



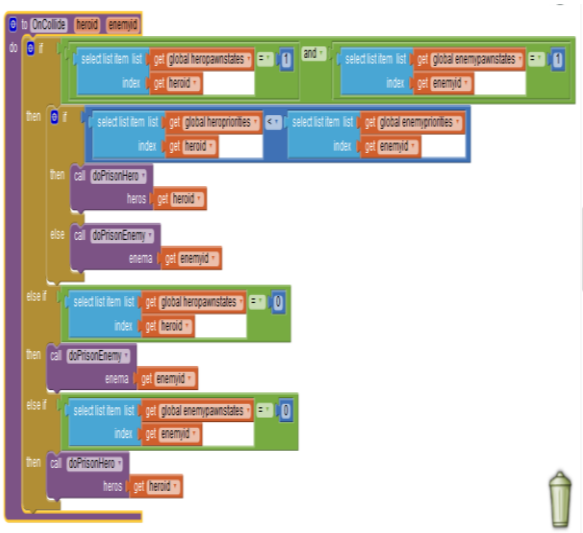
Gambar 7 Kode blok *when.ImageSprite1. Collidedwith*

Gambar 9 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*.



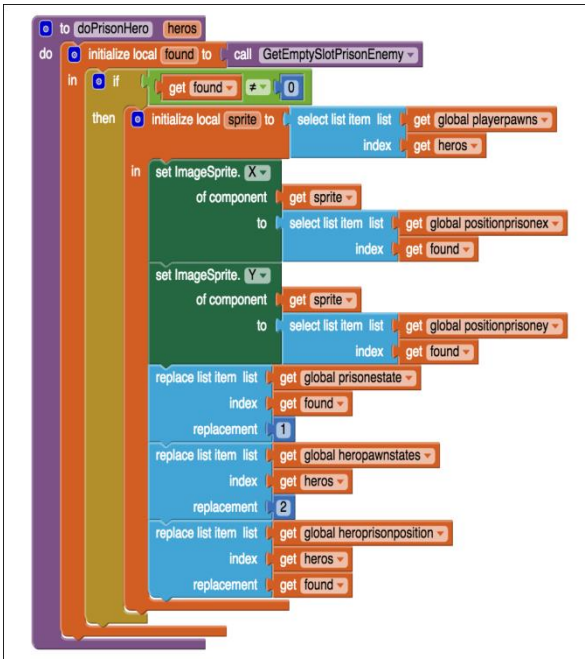
Gambar 8 Kode blok *to.OnCollide.heroid.enemyid*

Gambar 10 adalah method untuk memasukkan ke dalam 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.*



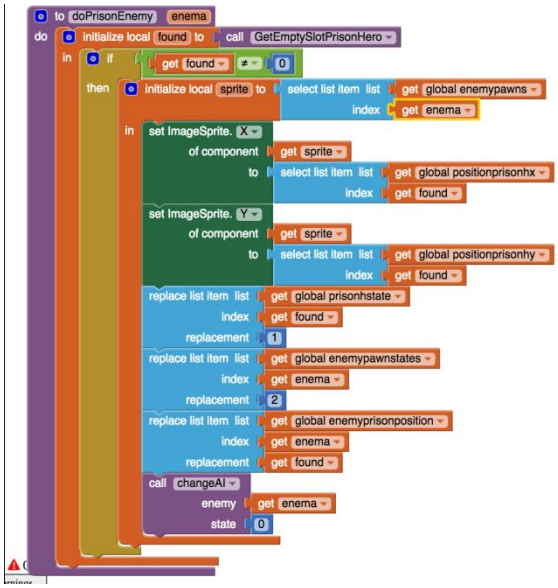
Gambar 9 Kode blok *to.OnCollide.heroid.enemyid*

Gambar 11 merupkan *method* untuk menentuan posisi *hero* saat dipenjara yaitu menggunakan *method to.doPrisonHero.heros.* Pada *method* tersebut digunakan untuk menentukan koordinat *hero* di penjara berdasarkan koordinat x dan y.



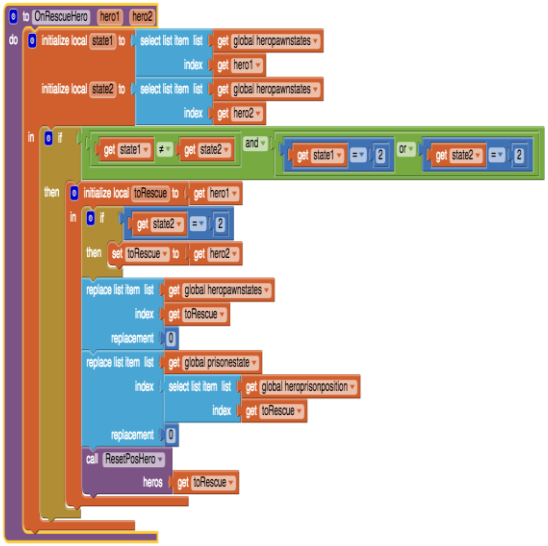
Gambar 10 Kode blok *to.doPrisonHero.heros*

Gambar 12 merupakan blok *method* untuk penentuan posisi *enemy* saat dipenjara yaitu menggunakan *method to.doPrisonEnemy.enema.* Pada *method* tersebut digunakan untuk menentukan koordinat *enemy* di penjara berdasarkan koordinat x dan y.



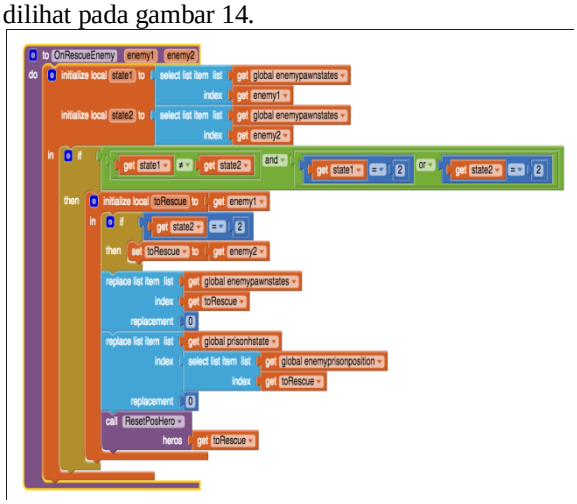
Gambar 11 Kode blok *to.doPrisonEnemy.enema*

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 dan 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 13.



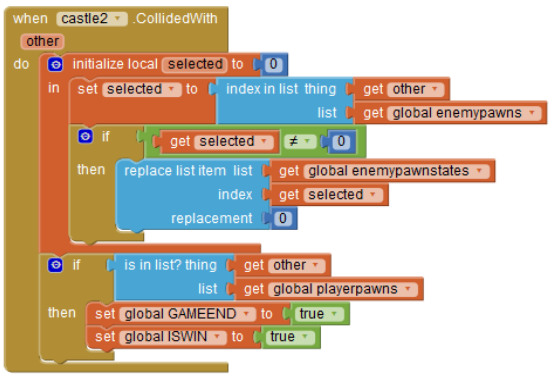
Gambar 12 Kode blok *to.OnRescueHero.hero1. hero2*

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 detil tentang *method to.onRescueEnemy.enemy1.enemy2* dapat dilihat pada gambar 14.



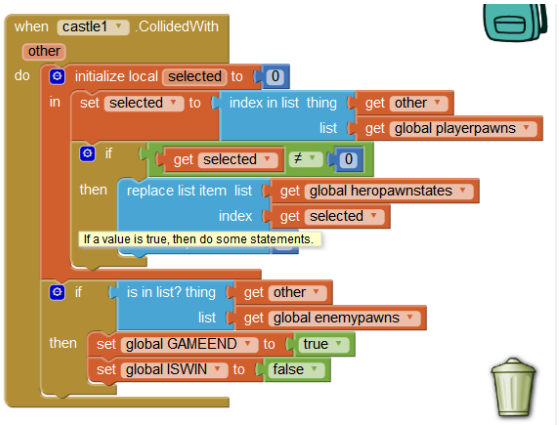
Gambar 13 Kode blok *to.OnRescueEnemy.enemy1. enemy2*

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* seperti pada gambar 14. Sedangkan untuk kondisi *enemy* menyentuh benteng *hero* menggunakan *method* blok *when.castle1.CollidedWith* seperti pada gambar 15.



.

Gambar 14 Status menang blok *when.castle2. collidedwith*



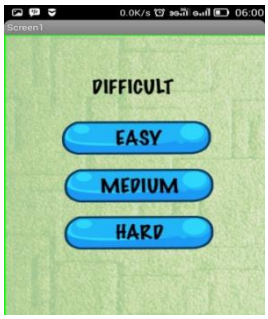
Gambar 15 Status kalah blok *when.castle1. collidedwith*

5. PENGUJIAN

Pengujian dilakukan dengan menggunakan *smartphone* android dengan kondisi normal. Saat aplikasi dibuka maka muncul halaman *Home.*

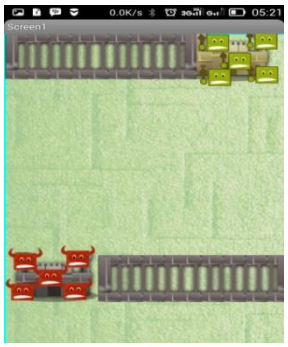
Gambar 16 Tampilan Home

Pengujian perintah pada saat tombol *difficult* di pencet untuk memilih tingkat kesulitan. Ditunjukkan pada gambar 17.



Gambar 17 Tampilan pilihan kesulitan

Pengujian perintah pada saat tombol *Play* pada halaman *Home* di pencet untuk memulai permainan. Ditunjukkan pada gambar 18.



Gambar 18 Tampilan Arena

6. KESIMPULAN

Setelah melalui tahap perancangan dan implementasi sistem, maka dapat diambil kesimpulan bahwa game berjalan dengan lancar, game lebih menarik dikarenakan tampilan yang menarik dan juga terdapat fitur *backsound.* Dan game bentengan ini tidak dapat menggunakan fungsi *multi touch* dikarenakan keterbatasan fungsi pada App Inventor.

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.

Wijayanto, Tri. 2015. *GAME TRADISONAL GOBAK SODOR BERBASI ANDROID*. Skripsi, Teknik Informatika, STIMIK AKAKOM, Yogyakarta.

Cioray, Dede. 2014. *Pengenalan "Undified Modeling Language/UML" Bagian 1. http://ebook.dede-gunawan.web.id/2014/12/ebook-uml-bahasa-indonesia.html. Diakses 20 Juli 2016.*

Dewanata, Rachmat Catur. 2012. *GAME EDUKASI PERTOLONGAN PERTAMA PADA KECELAKAAN* .Skripsi, Teknk Informatika, UII, Yogyakarta.

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.*