

**RANCANG BANGUN PERMAINAN VIRTUAL PET BERBASIS
AUGMENTED REALITY DENGAN OUTFIT SEBAGAI MARKER
(TAMAGO-CLOTH)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Antoni Rahardian Lubis

NIM:125150107111005



PROGRAM STUDI TEKNIK INFORMATIKA
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

RANCANG BANGUN PERMAINAN VIRTUAL PET BERBASIS *AUGMENTED REALITY*
DENGAN *OUTFIT* SEBAGAI MARKER (TAMAGO-CLOTH)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Antoni Rahardian Lubis
NIM: 125150107111005

Skripsi ini telah diuji dan dinyatakan lulus pada
25 Agustus 2016
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Issa Arwani, S.Kom, M.Sc
NIP:19830922 201212 1 003

Wibisono Sukmo Wardhono, S.T, M.T
NIK: 201008 820404 1 001

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawa, S.T, M.T, Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang,



Antoni Rahardian Lubis

NIM: 125150107111005

KATA PENGANTAR

Puji syukur kepada Tuhan Yesus Kristus, atas limpahan Rahmat dan Karunia-Nya, sehingga penulis dapat merampungkan skripsi dengan judul: RANCANG BANGUN PERMAINAN VIRTUAL PET BERBASIS *AUGMENTED REALITY* DENGAN *OUTFIT*. Ini untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar Sarjana Komputer Strata Satu pada Program Studi Teknik Informaika Fakultas Ilmu Komputer Universitas Brawijaya.

Penghargaan dan terima kasih yang setulus-tulusnya kepada Ayahanda tercinta Adian Salomo Lubis dan Ibunda yang kusayangi Sari Rahatni yang telah mencurahkan segenap cinta dan kasih sayang serta perhatian moril maupun materil. Semoga Tuhan Yesus selalu melimpahkan Rahmat, Kesehatan, Karunia dan keberkahan di dunia dan di akhirat atas budi baik yang telah diberikan kepada penulis.

Penghargaan dan terima kasih penulis berikan kepada Bapak Issa Arwani, S.Kom, M.Sc selaku Pembimbing I dan Bapak Hadriani Wibisono Sukmo Wardhono, S.T, M.T selaku Pembimbing II yang telah membantu penulisan skripsi ini. Serta ucapan terima kasih kepada :

1. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya.
2. Sahabat-sahabat saya Laskar Skripsi Asik dan rekan-rekan mahasiswa khususnya program studi S1 TIF.
3. Eka Indriani yang bersedia memberikan akomodasi selama menyelesaikan skripsi ini.

Akhir kata penulis menyadari bahwa dalam penulisan skripsi ini masih jauh dari kesempurnaan. Karena itu, penulis memohon saran dan kritik yang sifatnya membangun demi kesempurnaannya dan semoga bermanfaat bagi kita semua. Amiin

Malang, 11 Agustus 2016

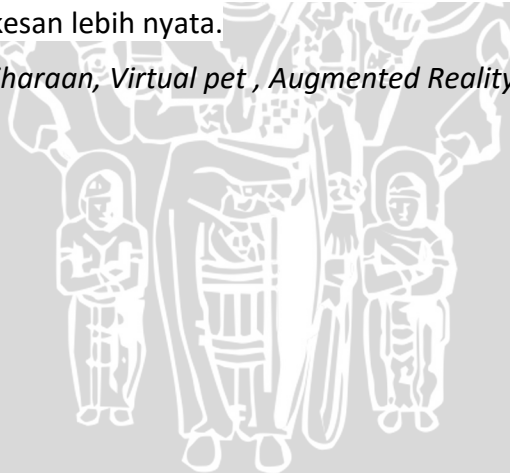
Penulis

toniinot10@gmail.com

ABSTRAK

Hewan peliharaan merupakan hewan yang dipelihara oleh manusia yang harus dirawat dan dipenuhi kebutuhan hidupnya serta memiliki tempat yang layak. Hewan peliharaan dapat dianggap sebagai teman bagi manusia karena pada umumnya hewan peliharaan memiliki karakter setia pada pemiliknya, dan dapat menghibur pemiliknya. Pada saat ini banyak orang yang mempunyai keinginan untuk memiliki hewan peliharaan namun tidak memiliki banyak waktu dan biaya untuk mengurus hewan peliharaan tersebut, sehingga mereka mengurungkan niatnya untuk memiliki hewan peliharaan. Terlebih lagi bagi mereka yang memiliki fobia terhadap hewan tertentu. Memelihara hewan peliharaan sebenarnya memiliki banyak keuntungan, antara lain membantu dalam mengatasi rasa sedih saat ditinggal orang terdekat, dan membantu dalam perkembangan anak. Dalam membantu mereka yang merasa kesulitan dalam memelihara hewan peliharaan, maka dibuatlah sebuah aplikasi *virtual pet*. *Virtual pet* merupakan hewan peliharaan dalam bentuk virtual dan tidak nyata, namun anda bisa merasakan bagaimana rasanya memelihara hewan sungguhan dalam aplikasi tersebut. Namun, aplikasi *virtual pet* yang sudah ada di pasaran selama ini masih menggunakan teknologi 2D, sehingga peneliti akan mengembangkannya ke dalam bentuk *Augmented Reality* (AR) agar sensasi memelihara hewan terkesan lebih nyata.

Kata Kunci : Hewan peliharaan, *Virtual pet*, *Augmented Reality* (AR).



ABSTRACT

Pets are treated animals and filled with their needs by humans and has the right to have a proper place. Pets can be regarded as a friend to humans because most pets have a loyal character to its owner, and they are able to entertain their owners. Nowadays, so many people crave to have a pet but they do not have much time and money to take care of the pet, so they will think it's better not to adopt it. Moreover, for those who have a phobia of certain animals. Raise a pet actually has many advantages, among others, help in overcoming sadness when someone left you, and assist in children growth. In helping those who find it difficult to raise a pet, then someone made a virtual pet application. Virtual pet is a pet in the form of virtual, but you can feel how to take care a real animal in such applications. However, the virtual pet application that is already on the market still using 2D technology, so that researchers will develop it to Augmented Reality (AR) technology, so that you can feel the sensation of taking care the pet more real.

Keywords : Pet, Virtual pet , Augmented Reality (AR).



DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	2
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Virtual pet	Error! Bookmark not defined.
2.2 <i>Augmented reality</i>	5
2.3 Blender 3D	6
2.4 Unity 3D	6
2.5 Vuforia.....	7
BAB 3 METODOLOGI	8
3.1 Studi Literatur	8
3.2 Analisa Kebutuhan Perangkat Lunak	9
3.3 Metode Perancangan Perangkat Lunak.....	9
3.4 Implementasi	9
3.5 Pengujian Perangkat Lunak.....	10
3.6 Pengambilan Kesimpulan.....	10
BAB 4 PERANCANGAN.....	11
4.1 Perancangan Game.....	11

4.1.1 Game Concept Document.....	11
4.1.2 Technical Design.....	14
4.1.3 Paper Prototyping	19
BAB 5 IMPLEMENTASI	22
5.1 Lingkungan Implementasi.....	22
5.1.1 Lingkungan Perangkat Keras	22
5.1.2 Lingkungan Perangkat Lunak	22
5.2 Batasan Implementasi	23
5.3 Implementasi Tampilan	23
5.3.1 Tampilan Logo	24
5.3.2 Tampilan Adopsi.....	24
5.3.3 Tampilan Image Target	25
5.3.4 Tampilan Utama Game	26
5.3.5 Tampilan Game Suit	26
5.3.6 Tampilan Shop.....	28
5.3.7 Tampilan Mini Game	28
5.4 Implementasi Kode Program	29
5.4.1 Source code untuk Menentukan Event dalam Peleccakan Image Target	29
5.4.2 Source Code untuk suit	34
5.4.3 Source Code penggunaan Makanan dan Minuman.....	39
5.4.4 Source code untuk mengurangi point hungry dan thirsty	42
5.4.5 Source Code Touch Event	43
BAB 6 PENGUJIAN	46
6.1 FPS Benchmark Testing.....	46
6.2 Playtesting.....	48
BAB 7 Penutup	56
7.1 Kesimpulan.....	56
7.2 Saran	56
DAFTAR PUSTAKA.....	58
LAMPIRAN	59

DAFTAR TABEL

Table 4.1 Konsep Utama Game.....	11
Table 4.2 Keterangan Karakter.....	12
Table 4.3 Keterangan Paper Prototye.....	19
Table 5.1 Spesifikasi Perangkat Keras	22
Table 5.2 Spesifikasi Perangkat Keras (<i>Mobile</i>)	22
Table 5.3 Spesifikasi Perangkat Lunak	23
Table 5.4 Gambar Image Target dan Kegunaanya.....	25
Table 5.5 Source Code Pelacakan Image Target dan Penentuan Event	29
Table 5.6 Source Code Suit.....	34
Table 5.7 Source Code penggunaan.....	39
Table 5.8 Source Code HungryandThirstyDecrease.....	42
Table 5.9 Source code destroybytouch.....	43
Table 6.1 Hasil Kuisisioner Playtesting Tamago-Cloth.....	48



DAFTAR GAMBAR

Gambar 2.1 Simplified Representation of a RV Continuum	5
Gambar 3.1 Blok Diagram Penelitian	8
Gambar 3.2 Iterative With Rapid Prototyping	9
Gambar 4.1 Diagram Alur Permainan	13
Gambar 4.2 Diagram Use Case permainan TAMAGO-CLOTH	14
Gambar 4.3 Activity Diagram Permainan Tamago Cloth Pilihan	16
4.4 Activity Diagram Permainan Tamago Cloth Suit	17
4.5 Activity Diagram Permainan Tamago Cloth Suit Memberi Makan dan Minum	18
Gambar 5.1 Tampilan Logo Game Tamago-Cloth	24
Gambar 5.2 Tampilan saat camera menscan marker babi six pack	24
Gambar 5.3 Tampilan saat camera menscan marker beans wears prada	25
Gambar 5.4 Tampilan Utama Game	26
Gambar 5.5 Tampilan saat camera menscan ada marker pet lain selain pet pemain	27
Gambar 5.6 Tampilan saat melakukan suit	27
Gambar 5.7 Tampilan Shop	28
Gambar 5.8 Tampilan Mini Game	28
Gambar 6.1 Pengujian FPS Benchmark Asus Zenfone 2	46
Gambar 6.2 Pengujian FPS Benchmark Samsung S4	47
Gambar 6.3 Pengujian FPS Benchmark Oppo R801	47
Gambar 6.4 Statistik playtesting	49
Gambar 6.5 Flow Graph method OnTrackableStateChanged	50
Gambar 6.6 Flow Graph method Incubate	52
Gambar 6.7 Flow Graph Method tambahHungry	53

BAB 1 PENDAHULUAN

1.1 Latar belakang

Memelihara hewan peliharaan memiliki banyak sekali keuntungan seperti, membantu dalam mengatasi rasa sedih saat ditinggal orang terdekat, membantu dalam perkembangan anak, bahkan anak-anak yang memiliki hewan peliharaan memiliki emosional yang berbeda dari orang yang tidak memilikinya [LSC-07]. Tetapi memiliki hewan peliharaan pun tidak mudah, terutama bagi orang yang sudah dewasa. Memberikan makan dan membersihkan hewan peliharaan bisa menjadi hal yang memberatkan bagi orang dewasa. Karena itu aplikasi virtual pet dibutuhkan untuk memecahkan masalah ini [CMD-12].

Sementara itu, virtual pet yang selama ini ada di pasaran adalah aplikasi virtual pet 2D yang masih sangat kuno. perkembangan teknologi informasi pengolahan citra AR (*augmented reality*) yang dapat menambahkan benda virtual dalam dunia nyata dapat membuat aplikasi virtual pet terasa lebih *real*.

Aplikasi Neko Atsume yang ada, memberikan interaksi yang dapat memenuhi kebutuhan dalam merawat hewan peliharaan, tetapi aplikasi ini berinteraksi masih dengan perantara layar handphone. Sedangkan untuk mendapat pengalaman yang lebih *real* diharapkan pengguna dapat langsung berinteraksi dengan *object* virtualnya [YTB-16].

Dalam penggunaanya, *user* perlu untuk mengakses virtual pet nya dimanapun dia berada. Sehingga perangkat *mobile* dibutuhkan dalam penggunaanya. Salah satu perangkat *mobile* dengan perkembangan aplikasi dan pengguna yang terbesar sekarang ini adalah android [ASM-13]. Sistem operasi ini menjalankan aplikasi inti yang dibangun sendiri tanpa melihat aplikasi pihak ketiga [DPA-13]

Tetapi, selain aplikasi ini dapat memberikan manfaat bagi penggunanya, diharapkan juga aplikasi ini dapat memberikan keuntungan berupa materil pada pengembangnya. Sayangnya, aplikasi berbayar belum cukup diminati di Indonesia, itu ditandai dengan tingginya pembajakan aplikasi di Indonesia yang mencapai 86% dengan kerugian sebesar Rp12,8 Triliun [RMN-13]. Sehingga, peneliti mencoba untuk mendapatkan keuntungan dengan melakukan penjualan *outfit* yang berisi marker awal yang dibutuhkan dalam penggunaan aplikasi. Untuk meningkatkan penjualan, sesuai prinsip ekonomi jumlah yang terbatas dengan kualitas yang baik akan menambah permintaan [UMN-NFS]. Maka, *outfit* akan dibuat terbatas dan unik satu dengan lainnya.

Dengan latar belakang tersebut, maka diambil judul Rancang Bangun Permainan Virtual pet Berbasis *Augmented Reality* Dengan *Outfit* Sebagai Marker (Tamagao-Cloth) yang diharapkan dapat memberikan keuntungan dan pengalaman yang baik untuk pengguna, memberikan keuntungan materil pada pengembang dan menciptakan pengalaman bermain yang baru bagi *virtual pet* . Aplikasi ini dilengkapi kemampuan untuk melakukan interaksi pada pemain

Tamago-Cloth yang lain, sehingga memungkinkan untuk mendapatkan pengalaman yang *real* bersama dengan orang lain.

1.2 Rumusan masalah

Berdasarkan permasalahan pada latar belakang di atas, maka rumusan masalah dikhususkan pada :

1. Bagaimana *gameplay virtual pet* berbasis AR dengan *outfit* sebagai marker (TAMAGO-CLOTH)?
2. Bagaimana cara menggabungkan antara teknologi pendeteksian marker milik vuforia dengan *outfit* yang menjadi marker?
3. Bagaimana implementasi permainan virtual pet berbasis AR dengan *outfit* sebagai marker (TAMAGO-CLOTH) pada perangkat mobile di platform android?
4. Bagaimana menguji kompatibilitas permainan TAMAGO-CLOTH dan penerimaan pengguna berdasarkan rancangan yang dibuat?

1.3 Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah :

Merancang dan mengimplementasikan game virtual pet berbasis AR dengan *outfit* sebagai marker (TAMAGO-CLOTH) dengan teknologi *augmented reality* yang dapat dipadukan dengan *outfit*.

1.4 Manfaat

Manfaat dari penelitian ini adalah :

1. Aplikasi ini dapat memberikan manfaat yang sama dengan memelihara hewan sungguhan, namun dengan harga yang lebih murah, perawatan yang lebih mudah, serta resiko yang lebih kecil namun dengan *experience* yang sama dengan memelihara hewan sungguhan.
2. Hasil penelitian ini diharapkan menjadi pedoman dalam pembuatan aplikasi *augmented reality* berbasis marker maupun markerless dalam lingkup yang lebih luas.
3. Aplikasi ini dapat menjadi aplikasi yang tidak hanya memberikan manfaat tetapi juga dapat menjadi bisnis yang dapat menghasilkan *profit* .
4. Menguji ketertarikan pengguna pada *virtual pet* berbasis *augmented reality*.

1.5 Batasan masalah

Agar permasalahan yang dirumuskan lebih terfokus dan tidak terjadi pelebaran topik pembahasan, maka penelitian ini dibatasi dalam hal :

1. Umpan balik yang digunakan untuk menampilkan karakter visual untuk pertama kali serta untuk melakukan kunjungan ke karakter *pet* lain menggunakan marker berbentuk *abstract* yang di tempel di atas *outfit*.
2. Perancangan dan implementasi AR pada sistem operasi Android.
3. Karakter yang dibuat merupakan karakter kartun/tidak nyata.

4. Hal yang bisa dilakukan oleh program ini hanyalah menemui hewan lain, memberikan tanda jika butuh perawatan, pertarungan antar hewan peliharaan, memberikan perawatan pada hewan peliharaan, membeli peralatan untuk hewan peliharaan dan melakukan mini game.

1.6 Sistematika pembahasan

Untuk mencapai tujuan yang diinginkan, sistematika penulisan skripsi ini adalah sebagai berikut :

BAB I Pendahuluan

Bab ini berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan pada penelitian Rancang Bangun Permainan Virtual pet Berbasis Augmented Reality Dengan Outfit Sebagai Marker (Tamagao-Cloth).

BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang tinjauan pustaka yang akan digunakan sebagai dasar teori serta penunjangnya. Mengenai *augmented reality* serta alat pendukungnya yang berguna sebagai teori dasar bagi penelitian Rancang Bangun Permainan Virtual pet Berbasis *Augmented Reality* Dengan *Outfit* Sebagai Marker (Tamagao-Cloth).

BAB III METODOLOGI

Membahas tentang langkah-langkah dalam merancang program mulai dari studi literatur, perancangan, implementasi, pengujian dan analisis.

BAB IV PERANCANGAN

Bab ini akan membahas tentang analisis kebutuhan dari perangkat lunak yang dibangun serta gambaran sistem secara keseluruhan sebelum diimplementasikan ke dalam bahasa pemrograman tertentu.

BAB V IMPLEMENTASI

Bab ini akan membahas tentang implementasi yang dilakukan dalam membuat aplikasi dari rancangan yang telah dibuat sebelumnya.

BAB VI PENGUJIAN

Bab Pengujian membahas proses dan hasil pengujian terhadap program yang sudah diimplementasikan serta melakukan analisa pada hasil pengujian sehingga dapat menghasilkan suatu kesimpulan.

BAB VII PENUTUP

Bab penutup berisi tentang kesimpulan yang didapat dari analisa hasil pengujian serta saran yang dapat digunakan untuk mengembangkan aplikasi lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini akan berisi kajian pustaka serta dasar teori yang dibutuhkan dalam penyusunan penelitian ini. Teori - teori dasar yang akan dibahas pada bab ini adalah *Virtual pet*, *Augmented reality*, Geolocation, Android, Blender, Unity, dan Vuforia.

2.1 Virtual pet

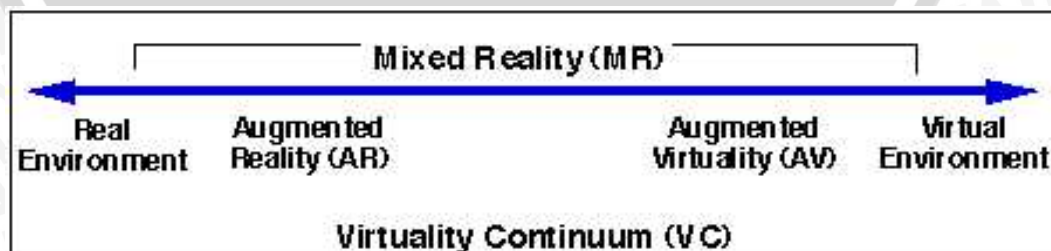
Virtual pet adalah suatu *entertainment* atau hiburan yang sangat terkenal sekarang ini. Mereka adalah pendamping manusia buatan yang dibuat menyerupai hewan peliharaan. Berinteraksi dengan *virtual pet* bisa digolongkan menjadi dua jenis. Jenis pertama adalah orang yang berinteraksi untuk tujuan tertentu seperti mendapatkan level tinggi agar bisa mengalahkan teman-temannya. Ada juga interaksi tanpa tujuan tertentu dimana pengguna dapat mengeksplor karakter *pet* dan menikmati membangun hubungan dengan virtual pet tersebut.

Ada beberapa hal yang biasanya dimiliki oleh sebuah permainan *virtual pet*, yaitu [WKP-16] :

- 1) Merespon saat ada panggilan
- 2) Merespon saat ada sentuhan
- 3) Melatih hewan peliharaan
- 4) Bermain dengan hewan peliharaan
- 5) Merubah tampilan *outfit* hewan peliharaan
- 6) Kompetisi atau pertarungan antar hewan peliharaan
- 7) Menemui hewan peliharaan lain
- 8) Memberikan tanda jika butuh perawatan

2.2 Augmented reality

Augmented reality atau yang biasa disebut dengan AR adalah pengelihan secara langsung maupun tidak langsung pada dunia nyata yang sudah ditambahkan dengan *object virtual* di dalamnya. AR adalah salah satu bagian dari *Mixed Reality* (MR) yang dapat kita lihat pada gambar 2.1 .



Gambar 2.1 Simplified Representation of a RV Continuum

SUMBER: [PMF-94]

Seperti yang kita lihat pada gambar 3.1. AR berada di sebelah kiri mendekati *Real Environment*. Sedangkan AV berada di sebelah kanan mendekati *Virtual environment*. Diharapkan dengan lebih menambahkan informasi dalam bentuk virtual kedunia nyata akan membuat pengguna lebih mudah [AAI-15]. Dalam penggunaannya AR digunakan di banyak bidang , di antaranya adalah :

- 1) Medik : Dokter dapat menggunakan AR sebagai visualisasi dan alat latihan dalam operasi.
- 2) Perakitan dan Reparasi : Instruksi menjadi lebih mudah dengan tampilan 3D yang menampilkan langkah demi langkah membuat instruksi menjadi lebih jelas.
- 3) Anotasi dan Visualisasi : AR digunakan untuk menganotasi objek dan *environment* dengan pengolahan informasi publik maupun private.
- 4) Perencanaan jalur robot : Dibandingkan untuk mengontrol robot secara langsung, akan lebih baik bila kita dapat mengontrol robot secara virtual.
- 5) Hiburan : Beberapa penampil yang menampilkan "Virtual Sets" yang menggabungkan real aktor dengan latar belakang virtual dalam real time dan 3D.
- 6) Pesawat Tempur : Pesawat tempur menggunakan HUD dan HMS untuk menghasilkan vektor grafik yang merepresentasikan apa yang pilot lihat di dunia nyata.

a. Marker

Marker Based Tracking dikembangkan sejak tahun 1980-an dan mulai dikembangkan dalam penggunaan AR paada awal 1990-an. Marker Biasanya merupakan ilustrasi hitam dan putih berbentuk persegi dengan batas hitam tebal dengan latar putih. Komputer akan mengenali posisi marker dan memunculkan objek 3D sesuai sumbu X, Y, dan Z nya.

2.3 Blender 3D

Blender 3D digunakan untuk merancang karakter binatang yang akan dimainkan di dalam game *virtual pet* ini.

2.4 Unity 3D

Unity 3D digunakan untuk melakukan perancangan permainan, adapun library yang digunakan adalah :

- 1) Random.Range : Digunakan untuk mendapatkan nilai random dengan batas atas dan bawah yang ditentukan
- 2) MoveTowards : Digunakan untuk memindahkan suatu object dr titik awal ke titik baru dengan kecepatan tertentu
- 3) Equals : Digunakan untuk membandingkan String

- 4) GetComponent<> : Digunakan untuk mengambil *component* sesuai kebutuhan.

2.5 Vuforia

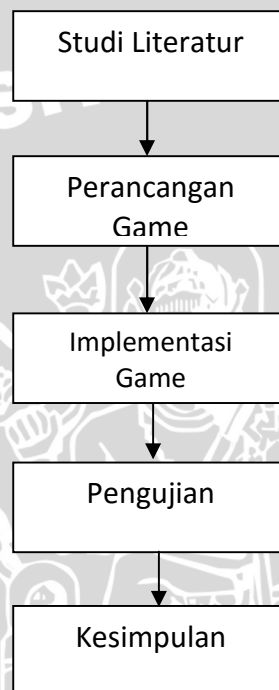
Vuforia digunakan untuk mendeteksi marker dan menampilkan objek pada marker sesuai dengan database yang ada. Adapun library yang digunakan adalah:

- 1) TrackableBehaviour : Digunakan untuk mengakses marker yang sedang dideteksi oleh vuforia
- 2) TrackableBehaviour.Status : Digunakan untuk memberi ataupun mendapatkan status suatu marker
- 3) CurrentStatus : Digunakan untuk mengetahui status marker pada waktu terbaru
- 4) ViewportToWorldPoint : Digunakan untuk merubah vector 3 dimensi dari *viewport* aplikasi ke dunia asli
- 5) GetSize marker : Digunakan untuk mengambil ukuran pixel



BAB 3 METODOLOGI

Pada bab ini dijelaskan langkah-langkah yang akan dilakukan dalam studi literatur, perancangan, implementasi dan pengujian aplikasi Rancang Bangun Permainan Virtual Pet Berbasis Augmented Reality Dengan Outfit Sebagai Marker (Tamagao-Cloth) yang akan dikembangkan. Kesimpulan dan saran akan disertakan sebagai catatan atas kemungkinan pengembangan aplikasi selanjutnya. Berikut diagram alir dari pengerjaan tugas akhir ini ada pada gambar 3.1 dibawah ini.



Gambar 3.1 Blok Diagram Penelitian

3.1 Studi Literatur

Studi Literatur merupakan dasar teori yang bertujuan untuk menunjang penulisan tugas akhir dan pengembangan aplikasi. Penelusuran bersumber dari media, buku, jurnal, website maupun hasil penelitian orang lain. Teori-teori pendukung ini akan meliputi :

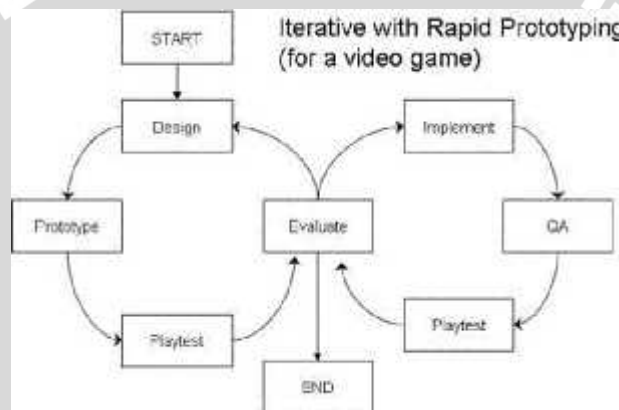
1. Virtual pet
2. Augmented Reality Marker
3. Blender
4. Unity
5. Vuforia

3.2 Analisa Kebutuhan Perangkat Lunak

Analisa kebutuhan berisi analisis spesifikasi perangkat lunak. Kebutuhan fungsional dari perspektif user akan dijabarkan oleh Use case diagram. Analisis kebutuhan dilakukan untuk mengetahui semua kebutuhan sistem yang kemudian dimodelkan ke dalam diagram. Analisis kebutuhan dilakukan dengan review dan *Paper Prototyping* yang dilakukan secara berulang-ulang. Dengan melakukan review tersebut, didapatkan kebutuhan fungsional yang diperlukan dalam pengembangan aplikasi AR *virtual pet* ini. Ditambah lagi peneliti akan melakukan analisa pada beberapa orang yang sudah pernah merawat hewan peliharaan. Pengumpulan data ini akan dilakukan dengan metode wawancara.

3.3 Metode Perancangan Perangkat Lunak

Metode perancangan yang akan digunakan dalam penelitian ini adalah Iterative dengan Rapid Prototyping.



Gambar 3.2 Iterative With Rapid Prototyping

SUMBER : [BBF-09]

Dengan metode ini, peneliti akan membuat paper prototype dan melakukan *testing* sebanyak tiga kali yang akan dilakukan oleh game tester secara acak, yang kemudian akan dievaluasi dan diimplementasikan ke dalam bentuk *mobile game* yang akan berjalan di atas platform android.

3.4 Implementasi

Implementasi perangkat lunak mengacu pada perancangan perangkat lunak. Implementasi perangkat lunak ini diawali dengan mendesain karakter binatang yang akan ditampilkan, lalu mengenalkan marker serta merancang interaksi antara objek virtual dengan dunia nyata. Kemudian implementasi dilakukan pada bahasa pemrograman C++ untuk membuat program dapat berjalan di Android. Kemudian dilanjutkan dengan mendesain database dengan menggunakan MySQL, dan yang terakhir adalah implementasi *user interface*.

3.5 Pengujian Perangkat Lunak

Pengujian dari aplikasi ini dilakukan untuk mengetahui performa serta faktor lain yang mempengaruhi dampak pada penggunaannya. Strategi yang digunakan dalam pengujian unit ini adalah Black-box testing (FPS), dan pengujian faktor lain dengan melakukan pemberian kuisisioner pada 25 orang pemain secara random.

3.6 Pengambilan Kesimpulan

Pengambilan kesimpulan dilakukan setelah semua tahapan pengujian selesai. Kesimpulan diambil dari hasil pengujian dan analisa setiap sistem yang telah dibangun. Yang terakhir adalah saran yang dimaksudkan untuk memperbaiki kesalahan yang terjadi dan sebagai pertimbangan untuk menyempurnakan aplikasi kedepannya.



BAB 4 PERANCANGAN

4.1 Perancangan Game

Perancangan game terdiri dari tiga tahap yaitu, *technical game design* yang akan menjelaskan teknis yang digunakan dalam pengembangan game, *game concept design* yang menjelaskan segala sesuatu yang terjadi di dalam game, dan *paper prototyping*.

4.1.1 Game Concept Document

Sub bab ini akan menjelaskan mengenai one-sheet dan ten-pager.

4.1.1.1 Konsep Utama Game

Game ini memiliki konsep utama yang terdiri dari judul, rating, platform, genre, dan USP (Unique Selling Point). Tabel 3.1 akan menjelaskan konsep game ini:

Table 4.1 Konsep Utama Game

No	Elemen	Keterangan
1	Judul Game	Tamago-Cloth
2	ESRB	Semua Umur
3	Platform	Android
4	Genre Permainan	Simulation
5	USP	1. Cara bermain yang menarik 2. Adanya merchandise yang bisa dibeli dan digunakan dalam permainan 3. Menggunakan teknologi <i>Augmented reality</i>

4.1.1.2 Gameplay

Pemain diharuskan untuk mengadopsi binatang pertamanya dengan cara men-*scan* image target pet yang dimilikinya. Pemain dapat melakukan perawatan pada binatangnya dengan cara men-*scan* image target peralatan perawatan yang tersedia, pemain dapat meningkatkan status pet dengan melakukan perlawanan dan pertarungan (suit) dengan pet lain, pemain dapat mengunjungi pet lain dengan cara men-*scan* image target pet lain, pemain juga dapat menaikkan status pet dengan membeli peralatan serta mendapatkan uang dengan memainkan mini game.

4.1.1.3 Karakter Dalam Game

Permainan ini memiliki tiga karakter yang berbeda. Tiap karakter memiliki nilai *attack*, *health point* dan *defense* yang berbeda. Setiap karakter juga dirancang dalam bentuk 3 dimensi. Tabel 3.2 akan memberikan deskripsi karakter dalam game ini:

Table 4.2 Keterangan Karakter

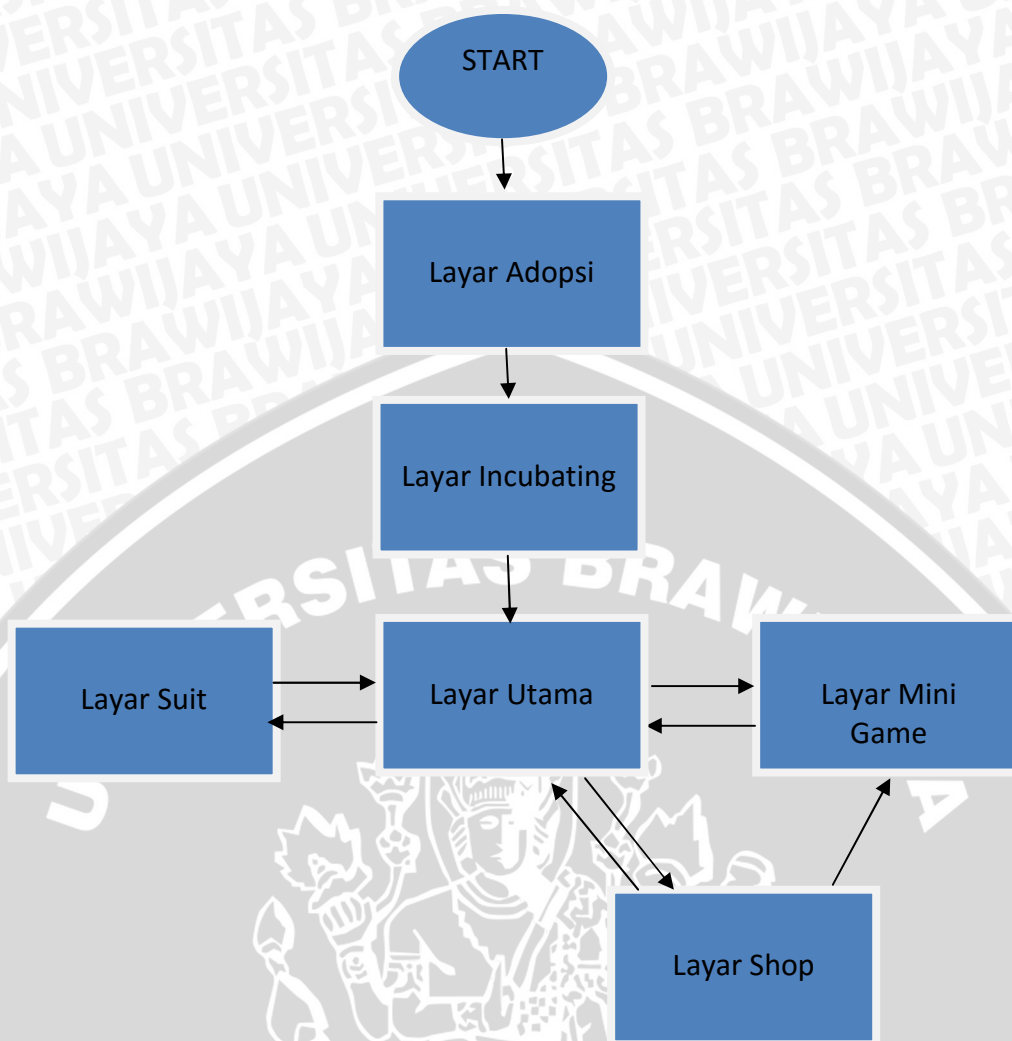
Karakter	Keterangan
	6pack-Piggy HP : 1300 Attack : 200 Defense : 150
	Bean-WearsPrada HP : 850 Attack : 300 Defense : 250

4.1.1.4 Kontrol Game

Untuk mengontrol game pemain hanya harus menekan layar pada pilihan pop-up yang keluar pada layar.

4.1.1.5 Game Flow

Gambar 3.3 menggambarkan alur game yang akan dimulai dari layar adopsi, setelah adopsi berhasil pemain akan dibawa ke layar utama. Di layar utama pemain diberikan pilihan untuk menuju ke layar Mini Game maupun shop. Dari layar shop pemain diberikan pilihan untuk kembali ke layar utama atau menuju layar mini game untuk mengumpulkan poin. Saat pemain lain masuk ke dalam *tracker* maka akan muncul tulisan *fight* yang bila ditekan akan membawa *player* ke layar suit. Pemain akan dibawa kembali ke layar utama saat permainan suit selesai.



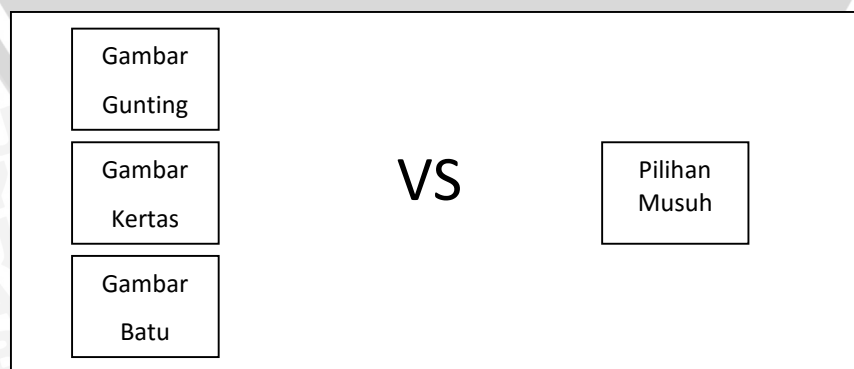
Gambar 4.1 Diagram Alur Permainan

4.1.1.6 Interface

Berikut beberapa rancangan game Tamago-Cloth :

1. Layar Bertarung

Layar ini berisi pilihan gunting, batu, kertas yang dapat dipilih untuk melakukan serangan. Gambar 3.6 akan menjelaskan interface bertarung.

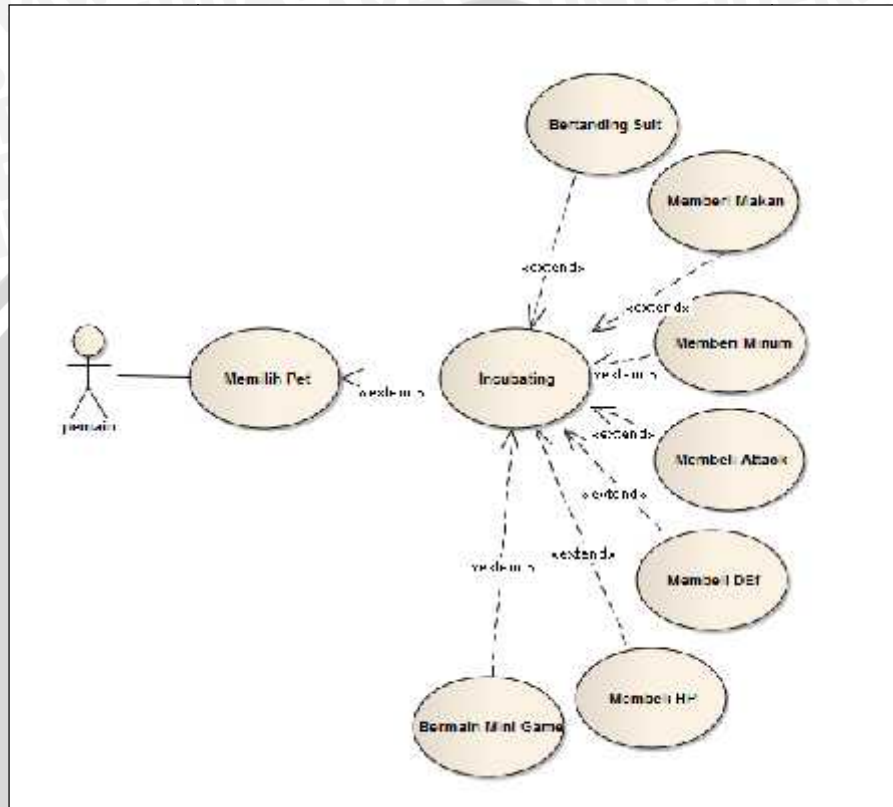


Gambar 3.6 Layar Permainan Suit

4.1.2 Technical Design

4.1.2.1 Use Case Diagram

Diagram Use Case pada perancangan game TAMAGO-CLOTH ini digunakan untuk memodelkan fungsionalitas dari game. Aktor dari permainan ini adalah pemain itu sendiri.



Gambar 4.2 Diagram Use Case permainan TAMAGO-CLOTH

4.1.2.2 Activity Diagram

Pembuatan activity diagram menggambarkan proses dan urutan aktivitas dari proses pada setiap use case yang ada. Berikut merupakan gambar dari masing-masing diagram activity.

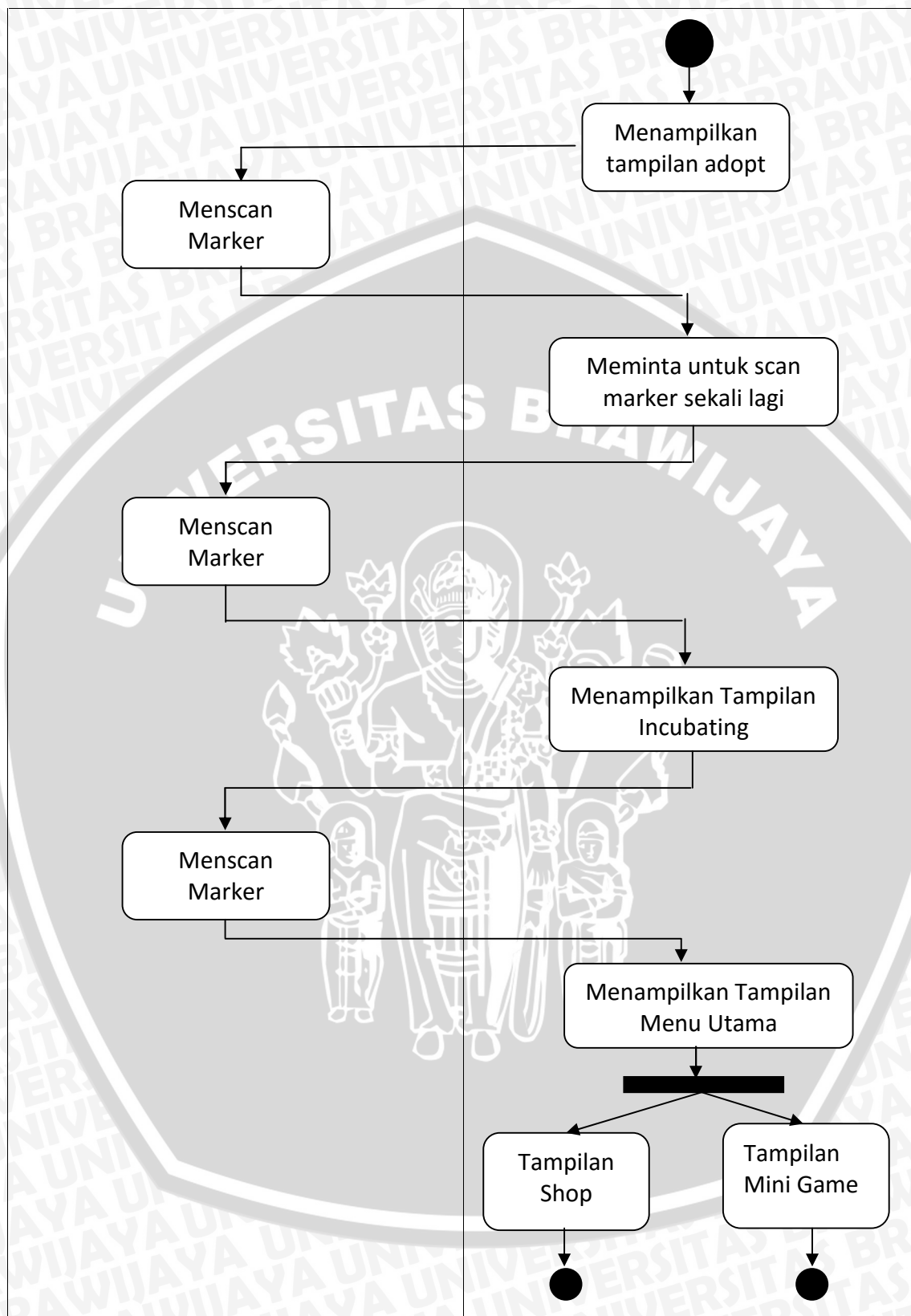
a) Permainan Tamago-Cloth pilihan

Pada gambar 4.3 menunjukkan urutan proses pada pilihan permainan. Saat pemain menjalankan permainan, sistem akan menampilkan tampilan adopsi, yang saat selesai dilakukan akan membawa pemain ke tampilan menu utama dimana terdapat dua pilihan stage.

Pemain

Sistem

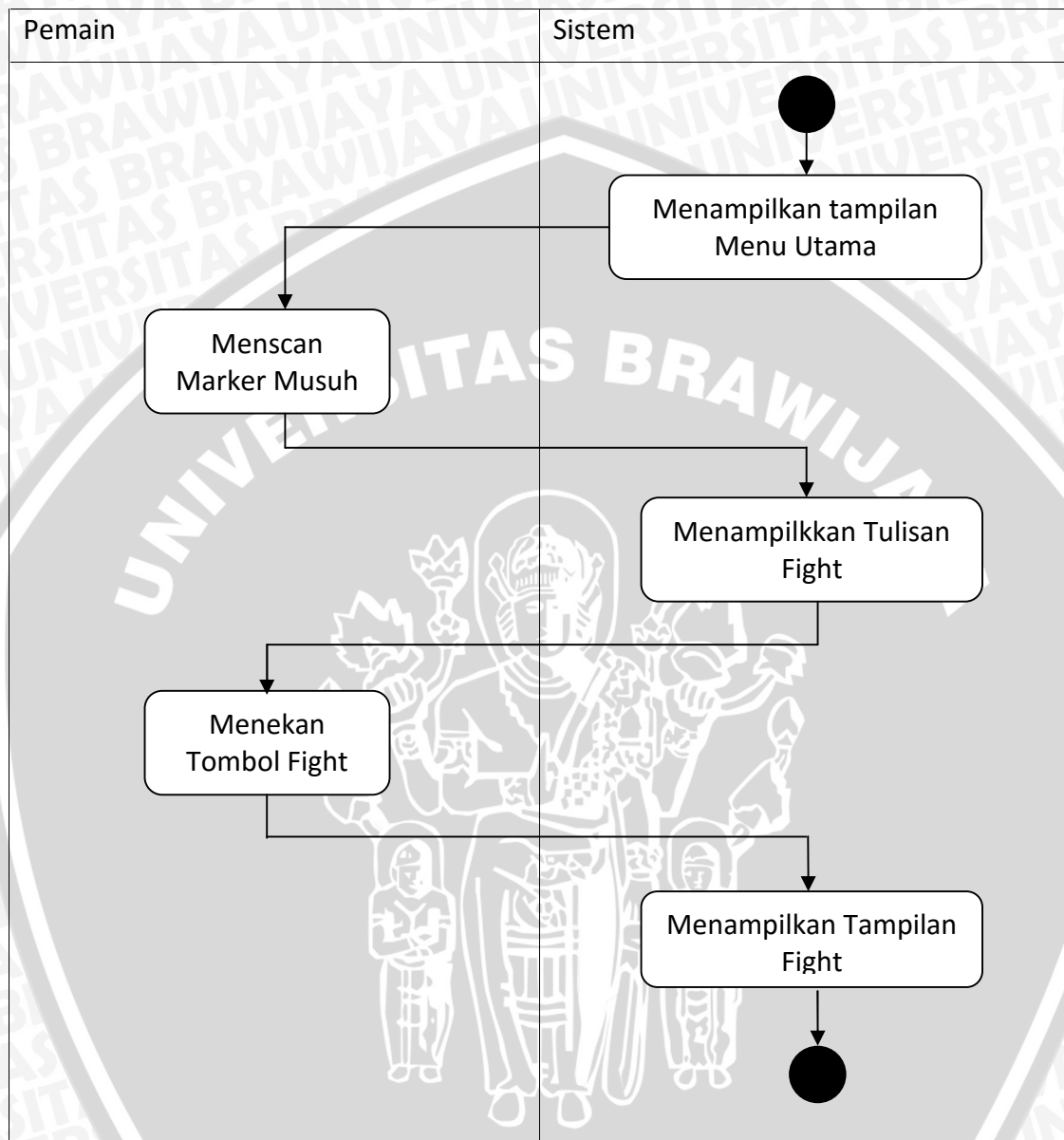




Gambar 4.3 Activity Diagram Permainan Tamago Cloth Pilihan

b) Permainan Tamago-Cloth Suit

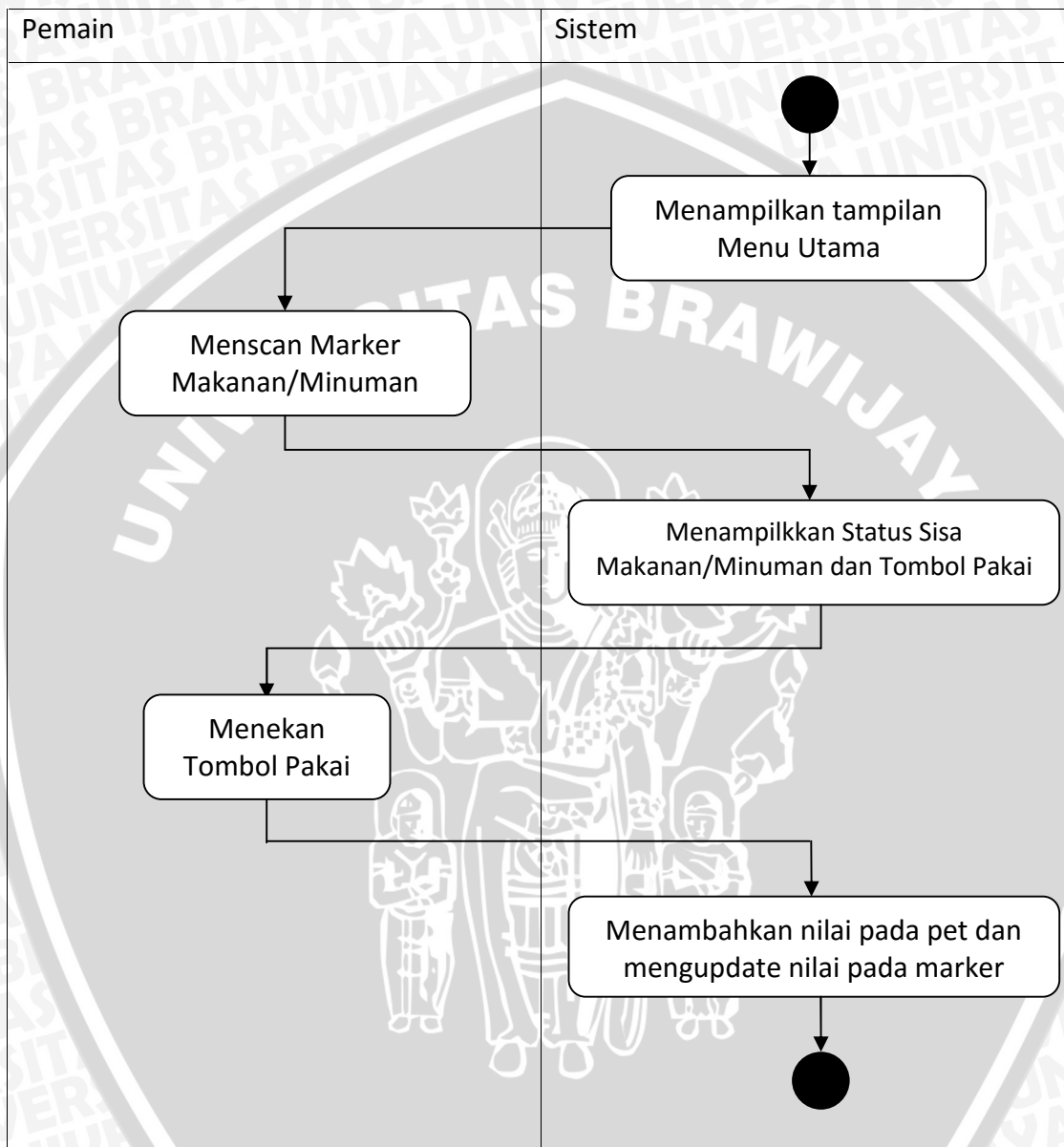
Gambar 4.4 akan menunjukkan urutan proses melakukan pertandingan suit.



4.4 Activity Diagram Permainan Tamago Cloth Suit

c) Permainan Tamago-Cloth memberi makan dan minum

Gambar 4.5 akan menunjukkan urutan proses melakukan pemberian makan dan minum.



4.5 Activity Diagram Permainan Tamago Cloth Suit Memberi Makan dan Minum

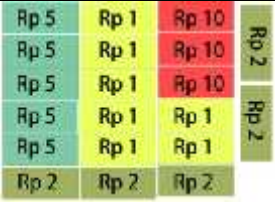
4.1.3 Paper Prototyping

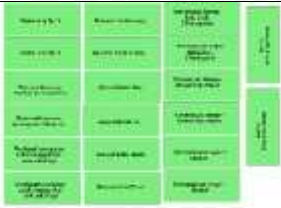
Dengan sebuah *paper prototype*, game dapat memberikan *gameplay* dan memberikan *user experience* yang mirip seperti game digital. Dengan *paper prototype* bisa mendapatkan keinginan user sebelum membuat digital game. Berikut ini terdapat beberapa proses paper prototyping.

4.1.3.1 Asset Paper Prototype

Beberapa asset yang digunakan dalam *paper prototype* ini adalah :

Table 4.3 Keterangan Paper Prototye

Nama Asset	Gambar Asset
Papan Permainan	
Karakter	
Uang	
Kartu Jujur Atau Tantangan	

Kartu Kesempatan	
Kartu perawatan peralatan	

4.1.3.2 Penjelasan Permainan

Paper prototype ini dapat dimainkan dengan peraturan seperti :

1. Kotak warna biru bertuliskan VS, berarti kedua pemain harus melakukan pertarungan suit. Pemenang 5 kali suit dapat mengurangi HP (Health Point Lawan) sebanyak 2 hitungan yang diakumulasi. *Critical damage* dapat terjadi apabila pemenang suit mendapat angka satu dari pelemparan dadu *critical*. *Critical damage* memberikan tambahan 10 poin untuk tim penyerang. Damage total didapat dari *attack* penyerang dikurangi oleh *defense* musuh. Permainan pada satu kotak berakhir setelah salah satu pemain memenagkan 5 putaran suit. Pemain yang menang dapat menggulirkan dadu sekali lagi, sedangkan yang kalah harus mundur 3 langkah. Pemain yang HP nya 0 akibat serangan terlebih dahulu dianggap *game over*.
2. Kotak kuning memiliki keuntungan dan kerugian menurut gambarnya :
 - a. Gambar kotoran membuat pet anda harus membuang kotoran sehingga anda akan ditahan 1 putaran.
 - b. Gambar anjing dengan tali pengekang membuat pet anda terkendali, sehingga anda bisa mengambil kartu kesempatan yang tersedia.
 - c. Gambar rumah mengartikan pet anda harus beristirahat selama 2 putaran.
 - d. Gambar peluit memberikan anda kesempatan untuk menggulirkan dadu sekali lagi.
3. Kotak hijau dengan gambar game controller membuat anda harus memainkan permainan *truth or dare* sesuai kartu yang anda tarik dari tumpukan kartu *truth or dare*.

4. Kotak dengan warna merah menandakan waktunya bagi pemain untuk merawat pet nya sesuai gambar yang tertera. Peralatan pet harus dibeli sesuai yang sudah disediakan. Pemain yang kehabisan uangnya terlebih dahulu dianggap kalah.
5. Setiap pemain harus melakukan 3 putaran sebelum bisa melakukan pertarungan (incubating mode). Bila pemain melakukan pertarungan sebelum *incubating* selesai dilakukan, maka pemain harus membayar denda sebesar 3 RP.

4.1.3.3 Kondisi Menang dang Kalah

Adapun kondisi menang dalam permainan ini adalah apabila memenuhi salah satu syarat dibawah ini :

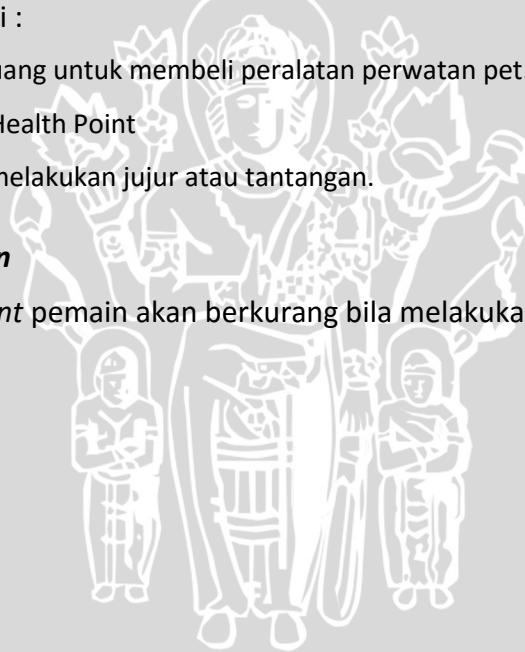
1. Pemain berhasil mencapai kotak finish.
2. Pemain berhasil membuat Health Point player lawan menjadi 0.

Adapun kondisi kalah dalam permainan ini adalah apabila memenuhi salah satu syarat dibawah ini :

1. Pemain kehabisan uang untuk membeli peralatan perawatan pet.
2. Pemain kehabisan Health Point
3. Pemain tidak mau melakukan jujur atau tantangan.

4.1.3.4 Kondisi Pemain

Kondisi *health point* pemain akan berkurang bila melakukan pertarungan suit dengan musuh.



BAB 5 IMPLEMENTASI

Bab ini akan berisi tentang penjelasan lingkungan perangkat keras dan lunak, batasan-batasan dalam implementasi, implementasi tampilan, dan implementasi kode program pada game.

5.1 Lingkungan Implementasi

Adapun lingkungan implementasi yang dijelaskan pada sub bab lingkungan implementasi perangkat keras adalah notebook, smartphone, dan perangkat lunak.

5.1.1 Lingkungan Perangkat Keras

Table 5.1 akan menampilkan spesifikasi notebook yang digunakan dalam merancang permainan tamago-cloth:

Table 5.1 Spesifikasi Perangkat Keras

Perangkat Keras	
Processor	Intel(R) Core(TM) i7-4510U CPU @ 2.00GHz
Memory	8.00 GB
Harddisk	1 TB
Graphic card	NVIDIA GEFORCE 840M

Adapun spesifikasi perangkat *mobile*, yaitu smartphone yang digunakan dalam implementasi game akan ditampilkan pada tabel 5.2.

Table 5.2 Spesifikasi Perangkat Keras (Mobile)

Perangkat Mobile	
Processor	Intel Atom Z3580 Quad core 2.3 GHz
Memory	4.00 GB RAM
Harddisk	32 GB
GPU	PowerVR G6430
Camera	13 MP

5.1.2 Lingkungan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan game ini akan ditampilkan pada table 5.3

Table 5.3 Spesifikasi Perangkat Lunak

Perangkat Lunak	
Operating Sytem	Windows 10 Home Single Language 64-Bit
2D Graphic Editor	Adobe Photoshop CS3
3D Graphic Editor	Blender 2.77
Basic Game	Unity 5.3.4f1
Bahasa Pemrograman	C#
Library <i>Augmented reality</i>	Vuforia 3.0

5.2 Batasan Implementasi

Adapun batasan-batasan dalam pembuatan game tamago-cloth ini adalah :

1. Game merupakan visualisasi yang diimplementasikan berdasarkan perancangan yang telah dibuat dan dijelaskan pada bab
2. Game ini dirancang dan dibangun menggunakan game engine unity dan library AR Vuforia
3. Game hanya dapat dimainkan pada perangkat *mobile* dengan basis operating system android
4. Objek maya pada game tidak akan muncul jika tidak menggunakan marker yang disediakan
5. Marker akan dicetak dan dilekatkan diatas *outfit*

5.3 Implementasi Tampilan

Bagian ini akan menjelaskan tentang tampilan-tampilan yang diimplementasikan pada game yang meliputi : tampilan logo, tampilan adposi, tampilan tarung, tampilan shop, tampilan home dan image target, tampilan mini game.

5.3.1 Tampilan Logo

Tampilan Logo menggunakan aplikasi 2D Adobe Photoshop. Berikut merupakan tampilan logo game yang digunakan dalam permainan ini.



Gambar 5.1 Tampilan Logo Game Tamago-Cloth

5.3.2 Tampilan Adopsi

Tampilan ini digunakan saat memulai permainan, pemain men-*scan* marker dan menemukan tombol adopsi saat marker yang terdeteksi adalah marker hewan. Tampilan terdapat pada gambar 5.2 seperti di bawah ini.



Gambar 5.2 Tampilan saat camera menscan marker babi six pack

Tampilan 5.2 didapatkan saat pemain ada pada layar adopsi dan software berhasil mendeteksi marker babi-sixpack.



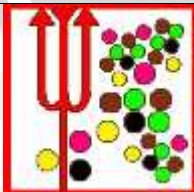
Gambar 5.3 Tampilan saat camera menscan marker beans wears prada

Tampilan 5.3 didapatkan saat pemain ada pada layar adopsi dan software berhasil mendeteksi marker minions wears prada.

5.3.3 Tampilan Image Target

Tampilan Image target berisi 4 gambar yaitu minuman, makanan, hewan1, dan hewan2. Marker menggunakan banyak warna agar marker lebih mudah dideteksi. Tabel 5.4 di bawah ini akan menjelaskan marker dan kegunaannya.

Table 5.4 Gambar Image Target dan Kegunaanya

Gambar Image Target	Kegunaan
	Marker untuk menampilkan minuman, digunakan untuk menambahkan poin thirsty pada pet sebanyak 10 poin. Bisa digunakan sebanyak 5 kali.
	Marker untuk menampilkan makanan, digunakan untuk menambahkan poin hungry pada pet sebanyak 10 poin. Bisa digunakan sebanyak 5 kali.
	Digunakan untuk menampilkan pet beans-wears prada.

	Digunakan untuk menampilkan pet babi sixpack.
	Digunakan untuk menampilkan player jenis 1 pada mini game
	Digunakan untuk menampilkan player jenis 2 pada mini game

5.3.4 Tampilan Utama Game

Tampilan ini merupakan tampilan utama dalam game ini, tampilan ini memberikan akses pada pemain untuk menuju ke tampilan shop dan mini game. Juga saat player men-*scan* marker musuh maka player akan di-*direct* menuju tampilan game suit.



Gambar 5.4 Tampilan Utama Game

Tampilan ini didapatkan saat marker devils wers prada terdeteksi oleh kamera handphone.

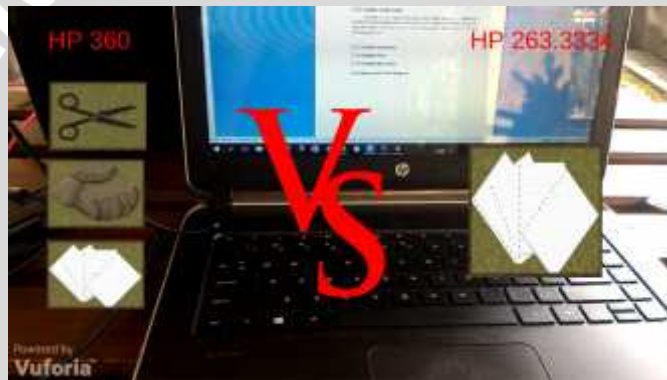
5.3.5 Tampilan Game Suit

Tampilan ini merupakan tampilan saat terjadi pertarungan suit, pemain diberikan tiga pilihan yang dapat dipilih untuk mengurangi HP musuh.



Gambar 5.5 Tampilan saat camera menscan ada marker pet lain selain pet pemain

Tampilan 5.5 didapatkan saat pemain menggunakan karakter devils wears prada dan software berhasil men-scan marker babi-sixpack.

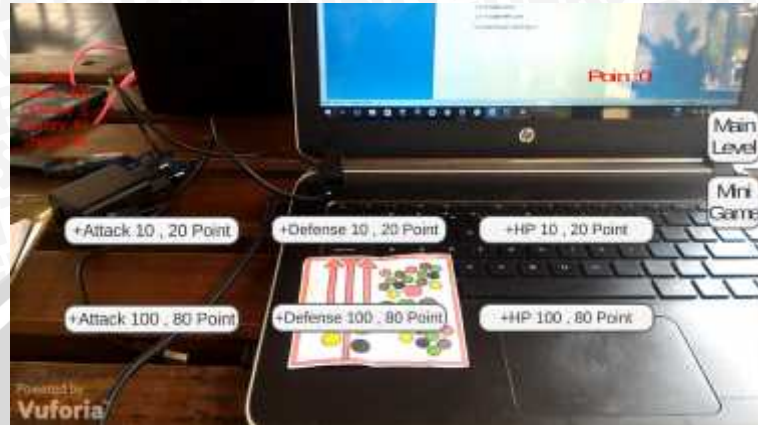


Gambar 5.6 Tampilan saat melakukan suit

Tampilan 5.6 didapatkan saat melakukan pertarungan, terdapat HP pemain di sebelah kiri atas dan HP lawan di sebelah kanan atas.

5.3.6 Tampilan Shop

Tampilan ini akan memberikan pilihan pada pemain untuk mengupgrade HP, Attack dan Defense dengan membayar dari hasil poin yang dikumpulkan di mini game .



Gambar 5.7 Tampilan Shop

Tampilan 5.7 didapatkan setelah menekan tombol shop dari tampilan menu utama.

5.3.7 Tampilan Mini Game

Tampilan ini akan menampilkan musuh dan peluru serta poin yang dikumpulkan dalam permainan.



Gambar 5.8 Tampilan Mini Game

Gambar 5.8 di atas didapatkan saat melakukan mini-game dengan devil wears prada sebagai karakternya. Tombol navigasi di pojok kanan bawah digunakan untuk menggerakkan karakter ke kiri maupun ke kanan, sedangkan tombol yang ada di ujung kiri bawah digunakan untuk menembakkan peluru.

5.4 Implementasi Kode Program

Pada sub bab ini akan dijelaskan mengenai implementasi code dari GAME TAMAGO-CLOTH. Dibawah ini merupakan beberapa *source code* pada game AI dan database pada game TAMAGO-CLOTH

5.4.1 Source code untuk Menentukan Event dalam Peleccakan Image Target

Table 5.5 Source Code Pelacakan Image Target dan Penentuan Event

1	using System.Collections;
2	using UnityEngine;
3	using UnityEngine.UI;
4	
5	namespace Vuforia
6	{
7	
8	public class edittrackable2 : MonoBehaviour,
9	ITrackableEventHandler
10	{
11	private TrackableBehaviour mTrackableBehaviour;
12	public string poop;
13	public string[] items;
14	public static string[] stacknama;
15	public string lala;
16	public Button fight;
17	private int cari;
18	public string namaPet;
19	public getNamaEnemy namaEnemy;
20	private string namaEnemyOK;
21	private penggunaan guna;
22	
23	IEnumerator Start()
24	{
25	WWW itemsdata = new
26	WWW("http://tugasskripsi.com/li.com/index.php");
27	yield return itemsdata;
28	string itemsdataString = itemsdata.text;
29	print(itemsdataString);
30	items = itemsdataString.Split(';');
31	mTrackableBehaviour =
	GetComponent<TrackableBehaviour>();

```

32         if (mTrackableBehaviour)
33         {
34             mTrackableBehaviour.RegisterTrackableEventHandler(this);
35         }
36
37         for (int x = 0; x <= 4; x++)
38         {
39             string poop = GetDataValue(items[x], "nama:");
40             lala = lala + "+" + poop;
41         }
42
43         stacknama = lala.Split('+');
44
45         GameObject gameControllerObject =
46         GameObject.FindGameObjectWithTag("ruler");
47         if (gameControllerObject != null)
48         {
49             namaEnemy =
50             gameControllerObject.GetComponent<getNamaEnemy>();
51             guna =
52             gameControllerObject.GetComponent<penggunaan>();
53         }
54         if (namaEnemy == null)
55         {
56             Debug.Log("Cannot find 'GameController'
57             script");
58         }
59     }
60
61     void Update()
62     {
63         namaPet = GetIDdanNama.namakirim;
64         namaEnemyOK = getNamaEnemy.namaEnemy;
65     }
66
67     string GetDataValue(string data, string index)
68     {
69         string value = data.Substring(data.IndexOf(index)
70         + index.Length);
71         if (value.Contains("|")) value =
72         value.Remove(value.IndexOf("|"));
73         return value;
74     }

```

```

72
73
74
75
76
77
78     public void OnTrackableStateChanged(
79         TrackableBehaviour.Status previousStatus,
80         TrackableBehaviour.Status newStatus)
81     {
82         if (newStatus ==
83             TrackableBehaviour.Status.DETECTED ||
84             newStatus == TrackableBehaviour.Status.TRACKED
85             ||
86             newStatus ==
87             TrackableBehaviour.Status.EXTENDED_TRACKED)
88         {
89             OnTrackingFound();
90         }
91         Else
92         {
93             OnTrackingLost();
94         }
95     }
96
97     private void OnTrackingFound()
98     {
99         cari = 0;
100
101         Renderer[] rendererComponents =
102         GetComponentsInChildren<Renderer>(true);
103         Collider[] colliderComponents =
104         GetComponentsInChildren<Collider>(true);
105
106         // Enable rendering:
107         foreach (Renderer component in rendererComponents)
108         {
109             component.enabled = true;
110         }
111
112         // Enable colliders:
113         foreach (Collider component in colliderComponents)
114         {
115             component.enabled = true;
116         }
117     }

```



```

112         Debug.Log("Trackable " +
mTrackableBehaviour.TrackableName + " foundlaal");
113
114
115
116         for (int k = 0; k <= 5; k++)
117         {
118             if (stacknama[k] ==
mTrackableBehaviour.TrackableName)
119             {
120                 cari = k;
121             }
122         }
123         string able = GetDataValue(items[cari - 1],
124 "identity:");
125
126         if (able.Equals("playable") &&
mTrackableBehaviour.TrackableName != namaPet)
127         {
128             fight.gameObject.active = enabled;
129             namaEnemy.setNamaEnemy(mTrackableBehaviour.name);
130         }
131         else if (able.Equals("notplayable"))
132         {
133             int keyCari = cari - 1;
134             guna.sisa(keyCari);
135         }
136     }
137     //
138     public void ok()
139     {
140         GetIDdanNama.namaEnemy kirim = namaEnemyOK;
141         Application.LoadLevel("suit");
142     }
143
144
145
146     IEnumerator WaitForRequest(www wwwk)
147     {
148         yield return wwwk;
149
150         if (wwwk.error == null)
151         {
152             print("WWW OK:" + wwwk.data);

```

```

153     }
154     Else
155     {
156         print("WWW error:" + wwwk.error);
157     }
158 }
159
160 private void OnTrackingLost()
161 {
162     Renderer[] rendererComponents =
163     GetComponentInChildren<Renderer>(true);
164     Collider[] colliderComponents =
165     GetComponentInChildren<Collider>(true);
166
167     // Disable rendering:
168     foreach (Renderer component in rendererComponents)
169     {
170         component.enabled = false;
171     }
172
173     // Disable colliders:
174     foreach (Collider component in colliderComponents)
175     {
176         component.enabled = false;
177     }
178
179     Debug.Log("Trackable " + mTrackableBehaviour.TrackableName + " lost");
180     fight.gameObject.active = false;
181 }

```

Penjelasan source code pada tabel diatas adalah sebagai berikut :

1. 24 - 29 --> menjelaskan cara mendownload data dari internet.
2. 31 - 35 --> inialisasi mtrackable.
3. 37 - 43 --> memasukan/membuat array stacknama.
4. 44 - 56 --> inialisasi gameobject guna & namaenemy
5. 60 - 62 --> mengambil data dari script lain.
6. 65 - 69 --> method yang digunakan untuk mengambil string dari array.
7. 78 - 92 --> method yang bekerja saat terjadi perubahan status scanning.
8. 94 -137 --> method saat terjadi perubahan status scanning ke found/tracked.

9. 116 - 124 --> mengidentifikasi object yang dicari.
10. 126 - 137 --> event yang terjadi berdasarkan hasil 116-124.
11. 139 - 143 --> method untuk memberikan nilai sekaligus mendirect pemain ke tampilan lain.
12. 146 - 158 --> method yang digunakan untuk memberikan tanda error dan berhasil saat melakukan koneksi ke internet atau sebuah link
13. 160 - 178 --> method saat terjadi perubahan status tracker ke hilang/lost.

5.4.2 Source Code untuk suit

Table 5.6 Source Code Suit

1	using UnityEngine;
2	using System.Collections;
3	using System;
4	using UnityEngine.UI;
5	
6	public class WEBin : MonoBehaviour
7	{
8	public string namague;
9	public string namamusuh;
10	
11	public float musuh;
12	public float Critical;
13	public float saya;
14	
15	private string[] items;
16	private string Namapeng;
17	private string[] stackNamapeng;
18	public float attakw;
19	public float attakm;
20	public float defw;
21	public float defm;
22	public float hpw;
23	public float hpm;
24	
25	public int cari;
26	
27	public UnityEngine.UI.Image testimage;
28	public UnityEngine.UI.Image testimage1;
29	public UnityEngine.UI.Image testimage2;
30	


```

31     public Button backtoMainGame;
32     public Text gameover;
33
34     public Boolean siap;
35
36     public Text UHP;
37     public Text EHP;
38
39     IEnumerator Start()
40     {
41         siap = false;
42         namague = GetIDdanNama.namakirim;
43         namamusuh = GetIDdanNama.namaEnemykirim;
44         testimage.enabled = false;
45         testimage1.enabled = false;
46         testimage2.enabled = false;
47
48         WWW itemsdata = new
49         WWW("http://tugasskripsi.com/li.com/index.php");
50         yield return itemsdata;
51         string itemsdataString = itemsdata.text;
52         print(itemsdataString);
53         items = itemsdataString.Split(';');
54
55         for (int x = 0; x <= 4; x++)
56         {
57             string poop = GetDataValue(items[x],
58             "nama:");
59             Namapeng = Namapeng + "+" + poop;
60
61             stackNamapeng = Namapeng.Split('+');
62             statusSaya();
63             statusMusuh();
64         }
65
66         string GetDataValue(string data, string index)
67         {
68             string value =
69             data.Substring(data.IndexOf(index) + index.Length);
70             if (value.Contains("|")) value =
71             value.Remove(value.IndexOf("|"));
72             return value;
73         }
74
75         public void statusSaya()

```

```
73     {
74         for (int k = 0; k <= 5; k++)
75         {
76             if (stackNamapeng[k] == namague)
77             {
78                 cari = k;
79             }
80         }
81         attakw = getAttack(cari);
82         hpw = getHP(cari);
83         defw = getDef(cari);
84     }
85
86     public void statusMusuh()
87     {
88         for (int k = 0; k <= 5; k++)
89         {
90             if (stackNamapeng[k] == namamusuh)
91             {
92                 cari = k;
93             }
94         }
95         attakm = getAttack(cari);
96         hpm = getHP(cari);
97         defm = getDef(cari);
98         siap = true;
99     }
100
101
102     public int getAttack(int indexke)
103     {
104         string attack;
105         attack = GetDataValue(items[indexke-1],
106 "attack:");
107         int attackInt = 0;
108         Int32.TryParse(attack, out attackInt);
109         return attackInt;
110     }
111     public int getHP(int indexke)
112     {
113         string HP;
114         HP = GetDataValue(items[indexke - 1],
115 "hp:");
116         int Hpint = 0;
117         Int32.TryParse( HP, out Hpint);
```

```

116         return Hpint;
117     }
118     public int getDef(int indexke)
119     {
120         string defense;
121         defense = GetDataValue(items[indexke - 1],
122 "defense:");
123         int defenseint = 0;
124         Int32.TryParse(defense, out defenseint);
125         return defenseint;
126     }
127
128     float damage(float critical, float defense,
129 float attack)
130     {
131         float attacksekarang = attack;
132         float damage;
133         float kemungkinandefense;
134         kemungkinandefense =
135         UnityEngine.Random.Range(1, 11);
136         float defensesekarang = defense /
137         kemungkinandefense;
138         if (critical == 1)
139         {
140             attacksekarang += 10f;
141         }
142         damage = attacksekarang - defensesekarang;
143         return damage;
144     }
145
146     public void war(int pilihansaya)
147     {
148         testimage.enabled = false;
149         testimage1.enabled = false;
150         testimage2.enabled = false;
151
152         saya = pilihansaya;
153         if (saya == musuh)
154         {
155             hpw -= 0;
156         }
157         else if (saya == 3 && musuh == 1)
158         {
159             hpw -= damage(Critical, defw, attakm);
160         }
161     }

```



```

158         else if (saya == 3 && musuh == 2)
159         {
160             hpm -= damage(Critical, defm, attakw);
161         }
162         else if (saya == 2 && musuh == 3)
163         {
164             hpw -= damage(Critical, defw, attakm);
165         }
166         else if (saya == 2 && musuh == 1)
167         {
168             hpm -= damage(Critical, defm, attakw);
169         }
170         else if (saya == 1 && musuh == 2)
171         {
172             hpw -= damage(Critical, defw, attakm);
173         }
174         else if (saya == 1 && musuh == 3)
175         {
176             hpm -= damage(Critical, defm, attakw);
177         }
178         if(musuh == 1)
179         {
180             testimage.enabled = true;
181         }
182         else if(musuh == 2)
183         {
184             testimage1.enabled = true;
185         }
186         else if(musuh == 3)
187         {
188             testimage2.enabled = true;
189         }
190     }
191
192     void Update()
193     {
194         if (hpw > 0 && hpm > 0)
195         {
196             musuh = UnityEngine.Random.Range(1, 4);
197             Critical = UnityEngine.Random.Range(1,
198 4);
199         }
200         else if (siap == true)
201         {
202             gameover.gameObject.active = true;

```

202	backtoMainGame.gameObject.active = true;
203	}
204	UHP.text = "HP " + hpw;
205	EHP.text = "HP " + hpm;
206	}
207	}

Penjelasan source code pada table 5.6 diatas sebagai berikut:

1. 41 - 46 --> menyiapkan inisialisasi awal.
2. 47 - 52 --> mengambil data online dari internet dan memasukkan ke array items.
3. 54 - 63 --> mengambil data dari array items dengan method statusaya() dan statusMusuh().
4. 64 - 70 --> method GetDataValue untuk mengambil data dari satu buah slot array.
5. 72 - 84 --> method statusSaya() untuk mengambil data attack, HP , def pemain saya.
6. 86 - 99 --> method statusMusuh() untuk mengambil data attack, HP dan def pemain musuh.
7. 102-125 --> method-method yang digunakan untuk mengambil nilai HP, Attack dan Defense pemain.
8. 127 -141 -->method yang menghitung damage serangan dengan perhitungan, critical point maupun defense point player lawan.
9. 143-190 --> method yang mengatur sesuai peraturan suit.
10. 193-203--> pengatur jalan permainan.
11. 204-206--> update tampilan HP pada UI

5.4.3 Source Code penggunaan Makanan dan Minuman

Table 5.7 Source Code penggunaan

1	public class penggunaan : MonoBehaviour {
2	public static string sisakirim;
3	public string tessimakirim;
4	public int sisakirim2;
5	public int jumlahhungryandthirsty;
6	public string attackkirim;
7	public string nama;
8	

```

9      public string[] items;
10     // Use this for initialization
11
12     IEnumerator Start()
13     {
14         jumlahhungryandthirsty = 10;
15         WWW itemsdata = new
16         WWW("http://tugasskripsi.com/li.com/index.php");
17         yield return itemsdata;
18         string itemsdataString = itemsdata.text;
19         print(itemsdataString);
20         items = itemsdataString.Split(';');
21     }
22     public void sisa(int keyCari)
23     {
24         attack kirim = GetDataValue(items[keyCari],
25         "attack:");
26         nama = GetDataValue(items[keyCari], "nama:");
27         sisakirim = GetDataValue(items[keyCari],
28         "defense:");
29         tessisakirim = sisakirim;
30         sisakirim2 = int.Parse(sisakirim);
31     }
32     public void tambahhungry()
33     {
34         if(sisakirim2 > 0) {
35             if (attack kirim.Equals("1"))
36             {
37                 HungryandThirstyDecrease.thirstypoint +=
38                 jumlahhungryandthirsty;
39             }
40             else if (attack kirim.Equals("2"))
41             {
42                 HungryandThirstyDecrease.hungrypoint +=
43                 jumlahhungryandthirsty;
44             }
45             sisakirim2 -= 1;
46             sisakirim = sisakirim2.ToString();
47             uploadnilai("def", sisakirim2);
48             StartCoroutine(updatenilai(1f));
49         }
50     }
51     Else
52     {
53         sisakirim = "HABIS !!!!";
54         uploadnilai("def", 0);
55     }

```



```

50     }
51     public void uploadnilai(string tujuan, int jumlahnya)
52     {
53         WWWForm form = new WWWForm();
54         form.AddField("nama", nama);
55         form.AddField("tujuan", tujuan);
56         form.AddField("jumlahnya", jumlahnya);
57         WWW wwwk = new WWW("http://tugasskripsi.comli.com/skripsiUpdate2.php",
58                               form);
59         StartCoroutine(WaitForRequest(wwwk));
60     }
61     IEnumerator WaitForRequest(WWW wwwk)
62     {
63         yield return wwwk;
64
65         if (wwwk.error == null)
66         {
67             print("WWW OK:" + wwwk.data);
68         }
69         Else
70         {
71             print("WWW error:" + wwwk.error);
72         }
73     }
74     string GetDataValue(string data, string index)
75     {
76         string value = data.Substring(data.IndexOf(index) +
77 index.Length);
78         if (value.Contains("|")) value =
79 value.Remove(value.IndexOf("|"));
80         return value;
81     }
82     IEnumerator updatenilai(float cooldown)
83     {
84         yield return new WaitForSeconds(cooldown);
85         WWW itemsdata = new WWW("http://tugasskripsi.comli.com/index.php");
86         yield return itemsdata;
87         string itemsdataString = itemsdata.text;
88         print(itemsdataString);
89         items = itemsdataString.Split(';');
90     }
91 }

```

Penjelasan source code pada tabel 5.7 di atas adalah sebagai berikut :

1. 12 - 19 --> inialisasi awal
2. 21 - 28 --> method mencari data dari sebuah array dengan slot array yang menjadi sebuah inputan
3. 29 - 50 --> method pengurangan nilai penggunaan marker dan penambahan nilai hungrypoint maupun thirstypoint serta penampilan tulisan habis saat marker 0. dan melakukan update ke database.
4. 51 - 59 --> method yang digunakan untuk mengupdate nilai ke database.
5. 60 - 72 --> method yang memberikan feedback saat terjadi kesalahan koneksi.
6. 73 - 78 --> method mengambil nilai tertentu dari sebuah slot array;
7. 79 - 87 --> method yang digunakan untuk mengambil data dari internet untuk pembaharuan data penggunaan;

5.4.4 Source code untuk mengurangi point hungry dan thirsty

Table 5.8 Source Code HungryandThirstyDecrease

	public class HungryandThirstyDecrease :
1	MonoBehaviour {
2	
3	public static int hungrypoint;
4	public static int thirstypoint;
5	private float waktu;
6	
7	public Text hungry;
8	public Text thirsty;
9	
10	void Start () {
	if (Application.loadedLevelName ==
11	"mainlevel" && hungrypoint == 0)
12	{
13	hungrypoint = 20;
14	thirstypoint = 20;
15	}
16	}
17	
18	Voif Update()
19	
20	waktu += Time.deltaTime;
21	
22	if (waktu > 5)
23	{
24	hungrypoint -= 3;
25	thirstypoint -= 5;
26	waktu = 0;

```

27     }
28
29     if(hungrypoint <= 10)
30     {
31         hungry.enabled = true;
32     }else if(hungrypoint > 10)
33     {
34         hungry.enabled = false;
35     }
36
37     if(thirstypoint <= 10)
38     {
39         thirsty.enabled = true;
40     }else if(thirstypoint > 10)
41     {
42         thirsty.enabled = false;
43     }
44
45     if(thirstypoint <= 0 && hungrypoint <= 0)
46     {
47         Application.LoadLevel("gameover");
48     }
49
50 }
51 }
52

```

Penjelasan source code 5.8 di atas adalah :

1. 11 - 17 --> inialisasi nilai awal hungrypoint dan thirstypoint.
2. 20 - 27 --> setiap 5 detik, makan hungrypoint akan dikurangi 3 dan thirstypoint akan dikurangi 5.
3. 29 - 49 --> pengaturan event yang terjadi bila hungrypoint dan thirstypoint bernilai dibawah 10, dan event yang terjadi jika hungrypoint dan thirstypoint kurang dari 0;

5.4.5 Source Code Touch Event

Table 5.9 Source code destroybytouch

1	using UnityEngine;
2	using System.Collections;


```
3 using UnityEngine.UI;
4
5 public class destroybytouch : MonoBehaviour {
6
7     private scoring scoreupdate;
8
9     void Start()
10    {
11        GameObject gameControllerObject =
12        GameObject.FindGameObjectWithTag("ruler");
13        if (gameControllerObject != null)
14        {
15            scoreupdate =
16            gameControllerObject.GetComponent<scoring>();
17        }
18        if (scoreupdate == null)
19        {
20            Debug.Log("Cannot find 'GameController'
21            script");
22        }
23    }
24
25    void OnTriggerEnter(Collider other)
26    {
27        if (other.name.Equals("tembok"))
28        {
29            Destroy(gameObject);
30        }
31        else if (other.name.Equals("peluru(Clone)") ||
32        other.name.Equals("peluru2(Clone)"))
33        {
34            Destroy(gameObject);
35            Destroy(other.gameObject);
36            scoreupdate.addScore(10);
37        }
38        }else if (other.name.Equals("PA_Warrior") ||
39        other.name.Equals("tank1"))
40        {
41            Destroy(gameObject);
42            Destroy(other.gameObject);
43            scoreupdate.gameOver();
44        }
45    }
46    else
47    {
48        Debug.Log(other.name);
49        Destroy(other.gameObject);
50    }
51 }
```

44	Destroy(gameObject);
45	}
46	
47	}
48	}
49	}

Penjelasan source code pada table 5.9 diatas adalah :

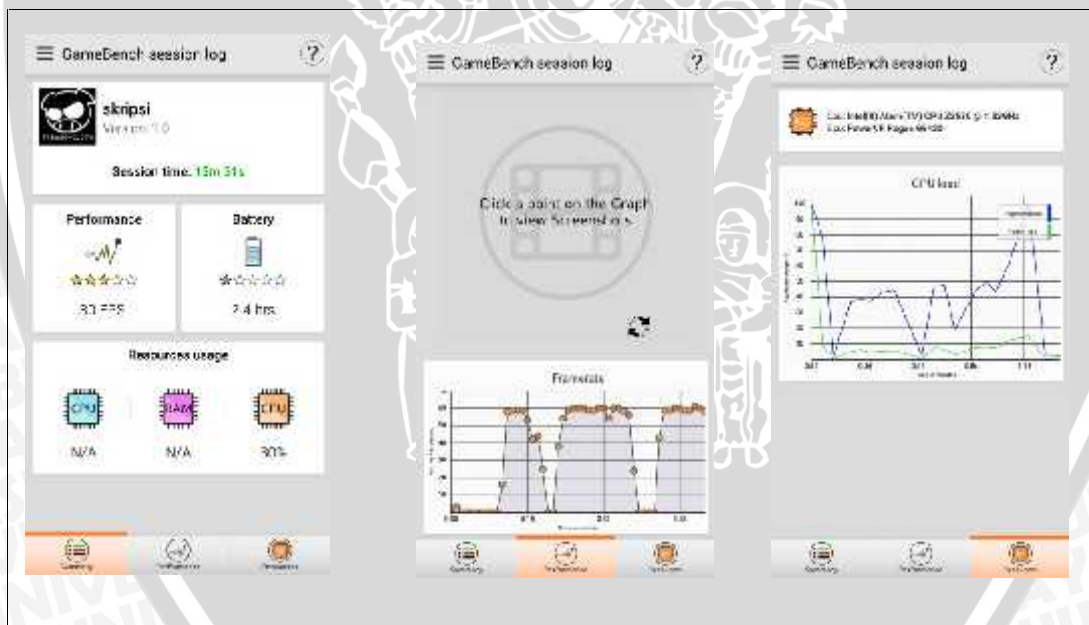
1. 10 - 19 --> inialisasi gameobject baru yang bernama scoreupdate dari komponen scoring pada object ruler.
2. 24 - 27 --> jika collider yang masuk dari object dengan nama tembok maka gameobject akan di destroy.
3. 28 - 32 --> jika collider yang masuk dari object dengan nama peluru(clone) atau peluru2(clone), maka gameObject akan di destroy bersamaan dengan peluru dan melakukan penambahan score sebanyak 10 point;
4. 34 - 39 --> jika collider yang masuk dari object dengan nama PA_Warrior atau tank1 maka gameObjet akan di destroy bersamaan dengan player yang akan di destroy juga. Lalu akan muncul tulisan gameOver di UI.
5. 41 - 44 --> saat gameObject menabrak yang lain selain yang diatur di atas maka gameObject dan Object lain yang bertabrakkan dengan yang akan di destroy.

BAB 6 PENGUJIAN

Bab ini berisi penjelasan tentang tahapan pengujian dari implementasi game Tamago-Cloth. Pengujian ini berfungsi untuk melihat performansi permainan dapat dijalankan di beberapa device berbeda, serta sebaik mana respon pemain dalam memainkan game ini. Pengujian ini akan menggunakan FPS benchmark sebagai pengujian performansi dan *compatibility* serta *playtesting* sebagai metode pengujian kepada pengguna, ditambah pengujian Basis Path sebagai pengujian *white box*.

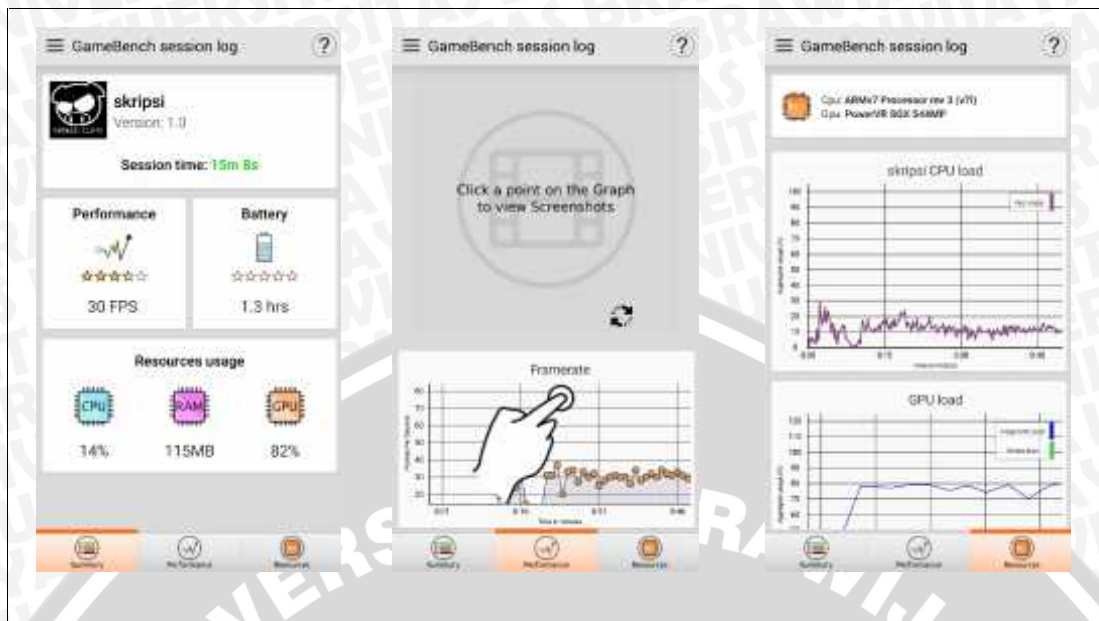
6.1 FPS Benchmark Testing

FPS (*Frame Per Second*) Benchmark Testing dilakukan untuk menguji performansi dari game Tamago-Cloth. Software yang digunakan dalam pengujian ini adalah GameBench. Software ini akan menganalisa penggunaan CPU, GPU, daya baterai serta FPS yang terjadi saat game dijalankan. Pengujian ini akan dilakukan di tiga *device* berbeda, yaitu Asus Zenfone 2, Samsung S4, dan Oppo R801. Pengujian dilakukan dengan memainkan game antara 15 menit hingga 17 menit sehingga didapat hasil sebagai berikut.



Gambar 6.1 Pengujian FPS Benchmark Asus Zenfone 2

Dalam Gambar 6.1 ditunjukkan bahwa game tamago-cloth yang dijalankan selama 15 menit 51 detik memiliki performa sebesar 30 FPS dan memakan sumber daya GPU sebesar 30%.



Gambar 6.2 Pengujian FPS Benchmark Samsung S4

Pada gambar 6.2 bisa kita ketahui bahwa game dimainkan selama 15 menit 8 detik, dan menghasilkan 30 frame per second serta menggunakan CPU sebanyak 14%, RAM sebesar 11.5 MB, dan GPU sebesar 82%.



Gambar 6.3 Pengujian FPS Benchmark Oppo R801

Oppo R801 memiliki spesifikasi hardware yang paling buruk dari dua handphone yang dicoba sebelumnya. Handphone ini hanya memiliki RAM sebesar 512 MB. Hal ini membuat hasil FPS yang dihasilkan pun tidak begitu baik, yaitu hanya 5 frame per second. CPU yang digunakan juga cukup tinggi hingga mencapai 22%, dan RAM yang digunakan mencapai 26 MB.

Dari ketiga pengujian ini didapatkan hasil rata-rata FPS sebesar 21,67 yang berasal dari perhitungan $(5+30+30)/3$. Disini juga peneliti mendapatkan kesimpulan bahwa game ini sebaiknya dimainkan oleh handphone dengan spesifikasi RAM diatas 1GB untuk mendapatkan pengalaman bermain yang baik. Karena nilai FPS yang dihasilkan oleh device ketiga dalam pengujian ini sangat rendah, sehingga permainan menjadi *lagging* dan tidak responsive. Dimana kita ketahui bahwa FPS adalah kemampuan hardware untuk mengulang gambar (frame) untuk setiap detiknya. Semakin kecil FPS yang dihasilkan, semakin buruk juga pengalaman yang akan diterima oleh pemain. Bisa kita lihat juga dari hasil pengujian pada *device* pertama dan kedua yang hasil penggunaan GPU nya lebih tinggi dari penggunaan CPU nya, dapat kita simpulkan bahwa game ini lebih bergantung pada pemrosesan GPU dibandingkan CPU. Nilai n/a pada *device* ketiga diakibatkan oppo r801 memang tidak memiliki spesifikasi GPU. Hal ini memperkuat hasil FPS rendah pada *device* ketiga.

6.2 Playtesting

Playtesting dilakukan setelah game sudah bisa dimainkan. Playtesting bertujuan untuk menguji apakah sebuah game menyenangkan, tujuan dari pembuatan game tersampaikan, dan apakah ada masalah dalam mekanisme game.

Kuisisioner berisi pertanyaan seputar kesenangan dan kepuasan pemain dalam bermain game tamago-cloth. Adapun pertanyaan yang tersedia pada kuisisioner *playtesting* adalah sebagai berikut :

- 1) Apakah *gameplay* yang dibuat cukup menarik?
- 2) Apakah *gameplay* yang dibuat dapat memberikan pengalaman memelihara hewan yang menyenangkan?
- 3) Apakah pemilihan *outfit* sebagai marker memberikan pengalaman yang baik bagi pemain?
- 4) Apakah karakter yang digunakan cukup menarik?
- 5) Apakah penggunaan kamera dalam permainan memberikan pengalaman bermain yang lebih baik ?

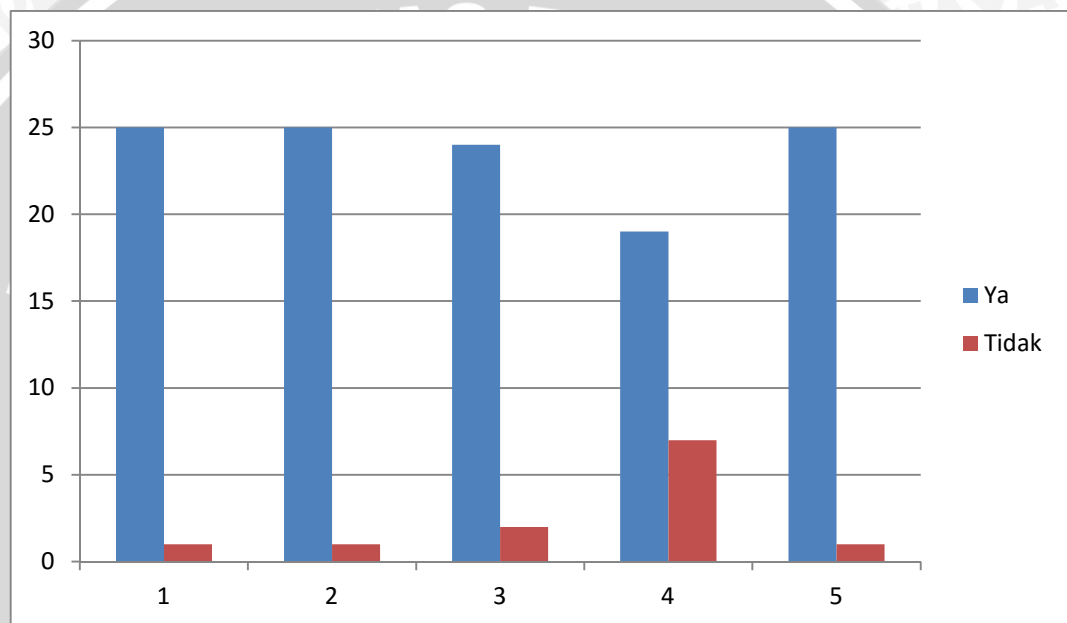
Pengujian dilakukan pada orang-orang yang ingin memelihara hewan secara random. Hasil yang didapat dari *playtesting* terhadap 26 orang mendapatkan hasil yang cukup baik. Adapun tabel 6.1 menunjukkan hasil dari *playtesting* game tamago-cloth .

Table 6.1 Hasil Kuisisioner Playtesting Tamago-Cloth

No.	Pertanyaan	Ya	Tidak
1	Apakah Gameplay yang dibuat cukup menarik?	25	1
2	Apakah Gameplay yang dibuat dapat memberikan pengalaman memelihara hewan yang	25	1

	menyenangkan?		
3	Apakah pemilihan <i>outfit</i> sebagai marker memberikan pengalaman yang baik bagi pemain?	24	2
4	Apakah karakter yang digunakan cukup menarik?	19	7
5	Apakah penggunaan kamera dalam permainan memberikan pengalaman bermain yang lebih baik?	25	1

Dari data yang disuguhkan pada tabel 6.1 maka kita dapat mendapatkan statistik seperti berikut ini :



Gambar 6.4 Statistik playtesting

Dari statistik diatas dapat kita lihat bahwa, pilihan 'ya' sangat dominan dalam kuisioner ini kecuali pada pertanyaan ke 4 dimana nilai 'tidak' cukup tinggi yaitu sebanyak 7 poin.

Dari hasil pengujian pengguna, didapatkan bahwa gameplay yang ditawarkan menarik. Penggunaan kamera sebagai daya tarik utama juga mendapatkan respon positif. Sedangkan karakter yang dibuat tampaknya kurang bisa diterima oleh pemain yang mengharapkan karakter yang lebih nyata. Pemilihan *outfit* sebagai maker juga dinilai cukup baik meskipun ada 2 orang yang merasa penggunaan *outfit* terkesan rumit dan membutuhkan biaya yang cukup tinggi.

6.3 Basis Path Testing

Basis path testing digunakan untuk mengukur tingkat kompleksitas dari algoritma hasil perancangan. Pada game Tamago-Cloth ini terdapat method-method yang akan diuji seperti OnTrackableStateChanged, Incubating, dan tambahHungry.

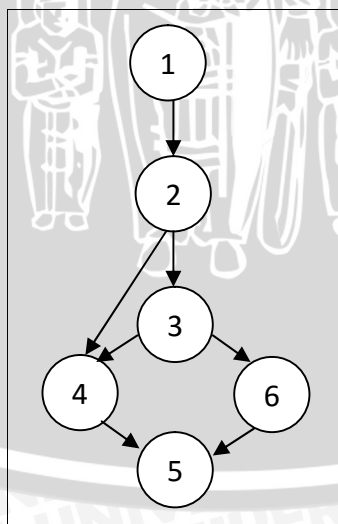
6.3.1 Basis Path Testing Method OnTrackableStateChanged

Jalur yang diharapkan adalah jalur yang dapat menampilkan objek berupa gambar jika image target terdeteksi dan jalur yang dapat menghentikan penampilan object ketika image taget tidak terdeteksi lagi atau lost. Algoritma OnTrackableStateChanged dijelaskan pada tabel 6.1.

Table 6.2 Pseudocode OnTrackableStateChanged

```
String poop , TrackableBehaviour.Status previousStatus,
TrackableBehaviour.Status newStatus ->1
IF newStatus == TrackableBehaviour.Status.Tracked || ->2
NewStatus == TrackableBehaviour.Status.Detected ->3
OnTrackingFound (); ->4
Else
OnTrackingLost (); ->6
End ->5
```

Gambar 6.5 Flow Graph method OnTrackableStateChanged



Dalam gambar 6. 5 cyclomatic complexity yang dihasilkan adalan $V(G) = 7\text{Edge} - 6\text{Node} + 2 = 3$.

Dengan hasil tersebut maka method ini memiliki 3 jalur independen yaitu :

1. Jalur 1 : 1-2-3-4-5
2. Jalur 2 : 1-2-4-5
3. Jalur 3 : 1-2-3-6-5

Uji kasus yang dieksekusi sesuai jalur independen. Adapun uji kasus akan dijelaskan pada tabel 6.3 .

Table 6.3 Uji Kasus pelacakan image target

Uji Kasus	Hasil Yang Diharapkan	Hasil Yang Diapat
Device tidak diarahkan ke image target	Object 3D tidak tampak	Object 3D tidak tampak
Device tidak diarahkan ke image target yang tepat	Object 3D tampak/muncul	Object 3D tampak/muncul

6.3.2 Basis Path Testing Method Incubating

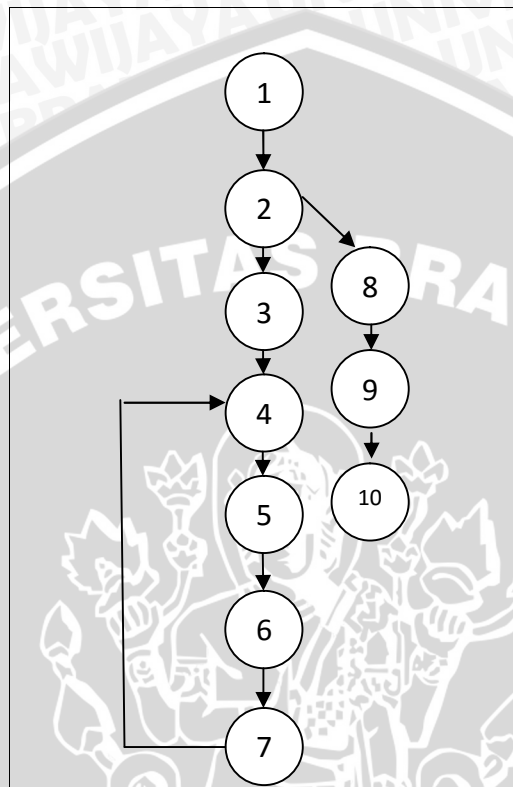
Jalur yang diharapkan adalah jika nama pet yang digunakan player memiliki nilai yang sama pada hasil scan maka, nilai inkubasi akan dikurangi 1 setiap detiknya, dan apabila nama pet dan nilai hasil scan tidak sama dan nilai tidak null maka nilai myegg akan dirubah menjadi false;

Table 6.4 Pseudocode Incubating

```

String namakirim, string namapet, float waktu, float
incubatetime = 10f , bool Myegg ->1
IF namakirim == namapet ->2
Myegg = true ->3
Waktu += Time.deltatime ->4
    IF waktu > 1 ->5
        Incubatetime -= 1 ->6
        Waktu = 0 -> 7
Else If namakirim != null ->8
Myegg = false -> 9
End ->10
    
```

Gambar 6.6 Flow Graph method Incubate



Dalam gambar 6.6 cyclomatic complexity yang dihasilkan adalah $V(G) = 10\text{Edge} - 10\text{Node} + 2 = 2$.

Dengan hasil tersebut maka method ini memiliki 3 jalur independen yaitu :

1. Jalur 1 : 1-2-3-4-5-6-7-4-5-6.....
2. Jalur 2 : 1-2-8-9-10

Uji kasus yang dieksekusi sesuai jalur independen. Adapun uji kasus akan dijelaskan pada tabel 6.5 .

6.5 Uji kasus pengurangan poin

Uji Kasus	Hasil Yang Diharapkan	Hasil Yang Diapat
Device menscan marker milik pemain	Waktu Inkubasi berkurang 1 poin setiap detik	Waktu Inkubasi berkurang 1 poin setiap detik
Device menscan marker milik pemain lain	Waktu Inkubasi tidak berkurang	Waktu Inkubasi tidak berkurang

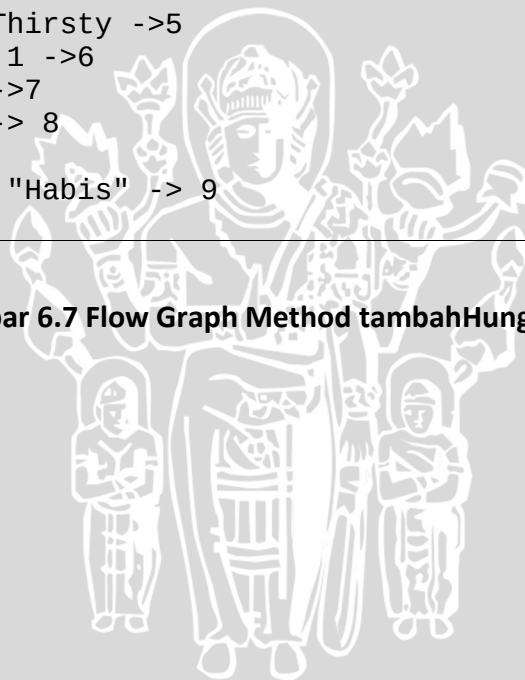
6.3.3 Basis Path Testing Method tambahHungry

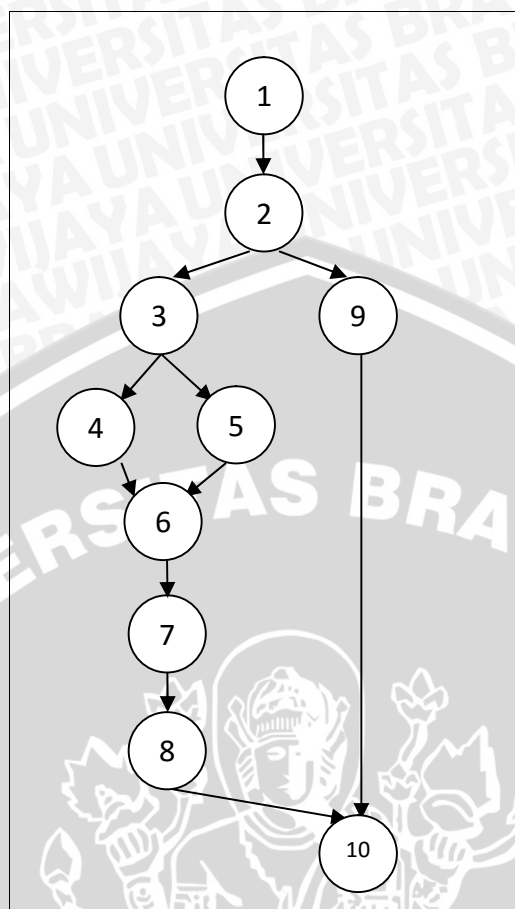
Jalur yang diharapkan adalah saat pemain menekan tombol gunakan pada layar maka sistem akan mengecek apakah masukan yang diberikan adalah makanan atau minuman dan apakah nilai sisa masih tersedia, bila masih maka nilai akan ditambahkan pada nilai makanan atau minuman, melakukan update nilai ke database dan mengambil nilai baru dari database, sedangkan jika nilai sisa 0 maka sistem akan memberikan tanda dengan tulisan "habis" di layar.

Table 6.6 Pseudocode tambahHungry

```
int Sisakirim, string attackkirim, string sisaKirim2 ->1
IF Sisakirim > 0 ->2
    IF attackkirim = 1 ->3
        += jumlahHungry ->4
    Else if attackkirim = 2
        += jumlahThirsty ->5
    Sisakirim -= 1 ->6
    Uploadnilai ->7
    Updatenilai -> 8
Else
    Sisakirim2 = "Habis" -> 9
End ->10
```

Gambar 6.7 Flow Graph Method tambahHungry





Dalam gambar 6. 7 cyclomatic complexity yang dihasilkan adalah $V(G) = 11\text{Edge} - 10\text{Node} + 2 = 3$.

Dengan hasil tersebut maka method ini memiliki 3 jalur independen yaitu :

1. Jalur 1 : 1-2-3-4-6-7-8-10
2. Jalur 2 : 1-2-3-5-6-7-8-10
3. Jalur 3 : 1-2-9-10

Uji kasus yang dieksekusi sesuai jalur independen. Adapun uji kasus akan dijelaskan pada tabel 6.5 .

Table 6.7 Uji Kasus PengecekanPoinSisa

Uji Kasus	Hasil Yang Diharapkan	Hasil Yang Diapat
Menekan tombol pakai pada marker makanan yang nilai sisanya >0	Poin sisa berkurang dan nilai hungrypoint bertambah	Poin sisa berkurang dan nilai hungrypoint bertambah
Menekan tombol pakai pada marker minuman yang nilai sisa nya > 0	Poin sisa berkurang dan nilai thirstypoint bertambah	Poin sisa berkurang dan nilai thirstypoint bertambah

Menekan tombol pakai
pada marker
makanan/minuman
yang nilai sisanya = 0

UI menampilkan tulisan
habis.

UI menampilkan tulisan
habis.



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil Implementasi dan pengujian game *augmented reality* berjudul tamago-cloth dapat ditarik kesimpulan sebagai berikut:

- 1) Gameplay Vitual Pet berbasis AR dengan outfit sebagai marker adalah sebagai berikut :

Pemain harus mengeluarkan uang untuk membeli perlengkapan untuk perawatan hewan peliharaan, dan untuk mendapatkannya harus melakukan mini-game ataupun pertarungan. Pemain dapat melakukan pertarungan dengan pemain lain melalui peraturan suit. Pemain harus melakukan inkubasi untuk masuk kedalam permainan. Melakukan mini-game dengan karakter sesuai keinginan pemain.

- 2) Menggabungkan teknologi pendeteksian marker milik vuforia dengan outfit dilakukan dengan cara memasukkan desain marker ke dalam database milik vuforia dan men-set marker tersebut pada object 3D yang ingin ditampilkan. Marker yang sudah berhasil dikenali akan dicetak dan ditempelkan diatas outfit.
- 3) Game Tamago-Cloth ini ber-genre simulation yang memvisualisasikan perawatan hewan peliharaan dengan animasi dan game yang diimplementasikan dalam bentuk augmented reality dengan menggunakan marker yang di sediakan.
- 4) Pengujian kompatibilitas dilakukan dengan cara melakukan pengujian FPS di 3 spesifikasi *handphone* yang berbeda. Dari hasil pengujian ini didapatkan bahwa permainan tamago-cloth tidak bisa dimainkan dengan *handphone* berspesifikasi RAM dibawah 1 GB dan tidak memiliki GPU.
- 5) Penerimaan pengguna pada permainan ini sangat baik. Hal itu bisa kita lihat dari baiknya *feedback* yang diberikan oleh pengguna dari pengujian playtesting yang sudah dilakukan.

7.2 Saran

Saran bagi pengembang selanjutnya :

- 1) Memberikan pilihan opsi serta fitur perawatan binatang yang lebih bervariasi. Seperti melakukan kebersihan binatang dan mengajak binatang peliharaan berjalan - jalan.



DAFTAR PUSTAKA

- [AAI-15] Andika Ady Irawan (2015). Rancang Bangun Aplikasi *Augmented reality* Penunjuk Arah Universitas Brawijaya Menggunakan Model Geolocation.
- [APR-13] Aan Erliasari , Paulus Insap Santosa dan Ridi Ferdiana (2013). *Augmented reality* Application for Book Promotion.
- [ARY-NHD] Ariansyah Nurhadi. Pengertian *Augmented reality*.
- [ASM-13] Asokan M (2013) . Android Vs iOS - An Analysis.
- [BBF-09] Brathwaite Brenda dan Schreiber Ian(2009). Challenges for Game Designer
- [BLR-16] About Blender 2016. [online] Tersedia di: <https://www.blender.org/about/> [diakses 26 januari 2016].
- [CMD-12] Catrinel Maria Danauta (2012). Virtual pet s: Interaction, Uses, Technology.
- [DPA-13] Deta Pratama AE (2013), Rancang Bangun Sistem Informasi Pariwisata Berbasis *Augmented reality* Pada Smartphone Android.
- [KCB-15] K Candra Brata (2015). Android Platform.
- [LSC-07] Lawson Shaun, Chesney (2007). The Impact of Owner Age On Companionship with Virtual pet s.
- [PMF-94] Paul Milgram and Fumio Kishino (1994). A Taxonomy Of Mixed Reality Visual Displats.
- [RMN-13] R. Maulana Nuradhi Wicaksana (2013). Dampak Negatif Pembajakan Software di Indonesia Dalam perpektif Ekonomi Kelembagaan.
- [RTA-97] Ronald T. Azuma (1997). A Survet of *Augmented reality*.
- [UMN-NFS] Ummu Nafisa . Need and Demand.
- [UNT-16] Manual User (2016). [online] Tersedia di: <http://docs.unity3d.com/Manual/> [diakses 26 januari 2016].
- [WKP-16] Common features of Virtual pet (2016) . [online] Tersedia di : https://en.wikipedia.org/wiki/Digital_pet [diakses 27 Januari 2016]
- [YTB-16] Video About Neko Atsume (2016) . [online] Tersedia di : https://www.youtube.com/results?search_query=neko+atsume [diakses 26 januari 2016]

LAMPIRAN

Kuisisioner Pengujian Playtesting Game Tamago-Cloth dari kuisisioner *playtesting* yang berjumlah 5 pertanyaan.

No.	Pertanyaan	Ya	Tidak
1	Apakah Gameplay yang dibuat cukup menarik?		
2	Apakah Gameplay yang dibuat dapat memberikan pengalaman memelihara hewan yang menyenangkan?		
3	Apakah pemilihan <i>outfit</i> sebagai marker memberikan pengalaman yang baik bagi pemain?		
4	Apakah karakter yang digunakan cukup menarik?		
5	Apakah penggunaan kamera dalam permainan memberikan pengalaman bermain yang lebih baik?		

