

Pengembangan Aplikasi *Photobooth* Berbasis *Augmented Reality*

Frans Immanuel^{*1)}, Aris Puji Widodo^{*2)}

^{**}Jurusan Ilmu Komputer/ Informatika, Fakultas Sains dan Matematika,
Universitas Diponegoro

¹⁾frans.marpaung2904@gmail.com, ²⁾arispw@gmail.com

Abstrak

Photobooth adalah sebuah area yang disediakan oleh penyelenggara acara di mana tamu dapat berpose untuk difoto dan langsung mendapatkan hasilnya saat itu juga. *Photobooth* dilengkapi dengan angle (sudut), background, lighting, dan hal pendukung lainnya untuk memberikan hasil terbaik. Permasalahan muncul ketika pengguna ingin melakukan pengambilan foto dengan beragam latar belakang dimanapun dan kapanpun, serta dapat melakukan pengambilan foto dengan objek 3 dimensi pada *smartphone* Android. Solusinya adalah pengembangan aplikasi *photobooth* berbasis *Augmented Reality* pada Android yang memungkinkan pengguna dapat menggunakan fitur *Augmented Reality* pada foto yang akan diambil. Aplikasi menggunakan software Unity untuk meletakkan objek 3 dimensi dan build ke Android serta aplikasi ini menggunakan Vuforia sebagai software library yang berfungsi sebagai pengenalan dan pemunculan model 3 dimensi tanpa penanda. Aplikasi menggunakan objek ARCamera dari library Vuforia, yang digunakan untuk menampilkan objek 3 dimensi yang sudah diatur posisinya di belakang kamera. Begitu kamera diakses, pengguna dapat memilih objek yang ingin ditampilkan dan aplikasi akan menampilkan objek 3 dimensi dan menampilkan objek 3 dimensi di mana kamera di arahkan. Pengguna dapat mengambil hasil gambar kemudian membagikannya pada sosial media Facebook, dan hasil gambar berupa foto dengan ekstensi PNG.

Kata kunci : *Augmented Reality, Photobooth, Android, Smartphone*

Abstract

Photobooth is an area provided by the event organizer where guests can pose to be photographed and get immediate results right away. *Photobooth* is equipped with angle (sudut), background, lighting, and other supporting things to give the best results. Problems arise when the user wants to take picture with variative background whenever and wherever and can take picture with 3 dimension object in Android *smartphone*. The solution is the development of *Augmented Reality photobooth* based app on Android that allows users to use *Augmented Reality* feature on the photo to be taken. This app used Unity software to place the 3 dimension object and build to Android and the app used Vuforia as a library software that acts as an introduction and appearance of a 3 dimensional model without a marker. This app used ARCamera feature from Vuforia library that used to display 3 dimension object that has been position behind camera. When user chooses to access camera, user can choose which object to display and this app will display 3 dimension object and display it wherever the camera point. User can take a picture then share it on Facebook social media, and the result is a picture with a PNG extension.

Keywords : *Augmented Reality, Photobooth, Android, Smartphone*

1 PENDAHULUAN

Seiring perkembangan zaman, dunia fotografi memasuki babak-babak kemajuan. Baik dari segi perkameraan, yaitu kamera *video shooting*, maupun kamera foto yang lebih dikenal dan sering dipakai adalah kamera *Digital Single Lens Reflex (DSLR)*. Untuk bidang sudut (*angle*), *background*, *lightning*, dan hal pendukung lainnya juga mengikuti kemajuan itu.

Perusahaan jasa foto lebih banyak menyediakan foto studio, yaitu foto ditempatkan dengan *background* warna khusus dan menunggu hasil jadi cetak foto. *Photobooth* semakin mengikuti perkembangan, *photobooth* menyediakan pengambilan foto dengan beragam *background*. Akan tetapi, fleksibilitas adalah keinginan setiap orang, oleh karena itu diperlukan sistem aplikasi yang dapat melakukan pengambilan foto dengan beragam *background* dimanapun dan kapanpun, maka dengan fitur *Augmented Reality* pada *smartphone* Android, pengguna dapat melakukan pengambilan foto dengan objek 3 dimensi pada *smartphone* Android.

Augmented Reality (AR) adalah sebuah variasi dari lingkungan virtual atau lebih sering disebut *Virtual Reality (VR)*. *VR* adalah *virtual reality* atau realitas maya adalah teknologi yang membuat pengguna dapat berinteraksi dengan suatu lingkungan yang disimulasikan komputer, suatu lingkungan sebenarnya yang ditiru atau benar-benar suatu lingkungan yang hanya ada alam imajinasi. *AR* adalah penggabungan benda-benda nyata dan maya di lingkungan nyata, berjalan secara interaktif dalam waktu nyata, dan terdapat integrasi antar benda dalam tiga dimensi, yaitu benda maya terintegrasi dalam dunia nyata dan mendefinisikan sistem *AR* memiliki karakteristik sebagai berikut (Azuma, 1997); menggabungkan lingkungan nyata dan

virtual, berjalan secara interaktif dalam waktu nyata, dan berintegrasi dalam tiga dimensi.

Teknologi *VR* membuat pengguna tenggelam dalam sebuah lingkungan sintetik. Teknologi *VR* membuat pengguna tenggelam dalam lingkungan maya, pengguna tidak bisa melihat dunia nyata, sebaliknya, teknologi *AR* membuat pengguna dapat melihat dunia nyata, dengan objek-objek *virtual* yang ditambahkan ke dunia nyata. *AR* merupakan sebuah teknologi yang menggabungkan benda maya baik 2D maupun 3D ke dalam lingkungan nyata lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata (Roedavan, 2014).

Terdapat beberapa aplikasi sejenis yang sudah ada, yaitu aplikasi pengambilan gambar, sebagai contoh, *camera mx*, *camera360*, *candy camera*, *camera FV-5*, *snapchat*, dan lain-lain. Akan tetapi, masih sedikitnya aplikasi pengambilan gambar yang menambahkan fitur *Augmented Reality* pada zaman di mana objek 3 dimensi sedang marak-maraknya, pembahasan aplikasi sejenis dapat di lihat pada bab 3.

Secara umum, aplikasi ini membantu pengguna untuk melakukan pengambilan foto dengan tambahan fitur 3 dimensi, pengguna tidak harus datang ke foto studio, melainkan dapat mengakses langsung pada aplikasi *smartphone* Android mereka.

2 TINJAUAN PUSTAKA

2.1 AUGMENTED REALITY

AR adalah penggabungan benda-benda nyata dan maya di lingkungan nyata, berjalan secara interaktif dalam waktu nyata, dan terdapat integrasi antar benda dalam tiga dimensi, yaitu benda maya terintegrasi dalam dunia nyata. Tujuan *AR* adalah untuk menambahkan informasi dan arti kepada

sebuah objek atau ruang yang nyata, tidak seperti VR, AR tidak membuat sebuah simulasi kenyataan, sebaliknya, dibutuhkan sebuah objek atau ruang yang nyata sebagai fondasi dan teknologi *incorporate* yang menambahkan data kontekstual untuk memperdalam pemahaman seseorang terhadap suatu objek (Azuma, 1997).

Riset AR bertujuan untuk mengembangkan teknologi yang memperbolehkan penggabungan secara waktu nyata terhadap *digital content* yang dibuat oleh komputer dengan dunia nyata. AR memperbolehkan pengguna melihat objek maya tiga dimensi yang diproyeksikan terhadap dunia nyata (Haller, 2006).

Definisi AR sebagai suatu teknologi yaitu (Madden, 2011) :

1. Kombinasi dunia nyata dengan komputer grafis.
2. Memberikan interaksi dengan objek secara waktu nyata.
3. *Tracks* objek secara waktu nyata.
4. Memberikan pengenalan gambar atau objek.
5. Memberikan konteks waktu nyata atau data.

Pembuatan sistem AR membutuhkan (Shapiro, 2001):

1. Model 3D dari objek untuk digabungkan dengan dunia nyata.
2. Korespondensi antara dunia nyata dengan model 3D melalui kalibrasi.
3. *Tracking* digunakan menentukan sudut pandangan pengguna terhadap dunia nyata.
4. Waktu nyata *Display* yang digabung dengan citra asli dan juga grafik komputer yang dibuat berdasarkan model.
5. Waktu respon terhadap gerakan dan akurasi antara gambar dan grafik sangat mempengaruhi keefektifan sistem.

2.2 MARKER

Penulisan Metode yang dikembangkan pada AR saat ini terbagi menjadi dua metode, yaitu *Marker Based Tracking* dan *Markerless Augmented Reality*.

1. *Marker Based Tracking*

Marker biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih. Komputer akan mengenali posisi dan orientasi marker dan menciptakan dunia virtual 3D yaitu titik (0,0,0) dan tiga sumbu yaitu X, Y, dan Z. *Marker Based Tracking* ini sudah lama dikembangkan sejak 1980-an dan pada awal 1990-an mulai dikembangkan untuk penggunaan AR.

2. *Markerless AR*

Salah satu metode AR yang saat ini sedang berkembang adalah metode "*Markerless AR*", dengan metode ini pengguna tidak perlu lagi menggunakan sebuah marker untuk menampilkan elemen-elemen digital, dengan *tool* yang disediakan Qualcomm untuk pengembangan *augmented reality* berbasis *mobile device*, mempermudah pengembang untuk membuat aplikasi yang markerless (Qualcomm, 2012).

Macam-macam teknik yang dapat digunakan dengan menggunakan *Markerless Tracking* yaitu sebagai berikut :

1. *Face Tracking*

Dengan menggunakan teknik algoritma yang mereka kembangkan, komputer juga dapat mengenali wajah manusia secara umum dengan cara mengenali posisi mata, hidung dan mulut, kemudian akan mengabaikan objek-objek lain disekitarnya seperti pohon, rumah dan benda lainnya.

2. *3D Object Tracking*

Face Tracking hanya mengenali wajah manusia secara umum, teknik *3D Object Tracking* dapat mengenali semua bentuk

benda yang ada di sekitar, seperti mobil, meja, televisi, dan lain-lain.

3. *Motion Tracking*

Komputer dapat menangkap gerakan, *Motion Tracking* telah mulai digunakan secara ekstensif untuk memproduksi film-film yang mencoba mensimulasikan gerakan.

4. *GPS Based Tracking*

Teknik *GPS Based Tracking* saat ini mulai populer dan banyak dikembangkan pada aplikasi *smartphone* (iPhone dan Android), dengan memanfaatkan fitur GPS dan kompas yang ada di dalam *smartphone*, aplikasi akan mengambil data dari GPS dan kompas kemudian menampilkannya dalam bentuk arah yang kita inginkan secara waktu nyata, bahkan ada beberapa aplikasi menampilkannya dalam bentuk 3D.

2.3 UNITY 3D

Unity adalah sebuah bentuk teknologi terbaru yang meringankan dan memudahkan *game developer* dalam membuat *game* (Creighton, 2011). *Unity* adalah sebuah *game engine / game authoring tool* yang mendukung orang kreatif untuk membangun *game*. *Unity* memiliki basis pengguna yang banyak (lebih dari 40.000 orang pada bulan April 2001) dan komunitas pemakai yang aktif dimana memungkinkan semua orang awam bertukar pengalaman dari para veteran (Blackman, 2011).

Unity menjadikan proses produksi *game* menjadi lebih mudah / sederhana dengan memberikan himpunan dari langkah-langkah logika untuk membangun sebuah *game* scenario yang menarik. Mengetahui bahwa untuk *game-game* yang tidak begitu spesifik tipenya, *Unity* memberikan *canvas* kosong dan himpunan prosedur yang konsisten, untuk memberikan imajinasi

sebagai batas kreativitas. Dengan menerapkan kegunaannya dari sebuah konsep objek *game*, dapat dibuat menjadi bagian untuk membagi *game-game* menjadi objek yang mudah diatur, yang dapat dibuat menjadi banyak bagian komponen individual. Dengan membuat objek individual dalam *game* dan mengenalkan fungsionalitasnya dengan komponen yang ditambahkan, kita dapat secara tidak terbatas mengembangkan *game* dalam proses pekerjaan yang logis. Bagian komponen memiliki variabel, dan beberapa pengaturan tambahan untuk mengendalikannya, dengan mengatur variabel-variabel ini, kita dapat mengendalikan seluruh efek yang dimiliki komponen dalam objek kita (Goldstone, 2009).

2.4 VUFORIA

Vuforia merupakan *software library* untuk *Augmented reality*, yang menggunakan sumber yang konsisten mengenai *computer vision* yang fokus pada *image recognition*. Vuforia mempunyai banyak fitur-fitur dan kemampuan, yang dapat membantu pengembang untuk mewujudkan pemikiran mereka tanpa adanya batas secara teknis. Dengan *support* untuk iOS, Android, dan Unity3D, *platform* Vuforia mendukung para pengembang untuk membuat aplikasi yang dapat digunakan di hampir seluruh jenis *smartphone* dan *table* (Amin & Govilkar, 2015).

Beberapa fitur yang dimiliki Vuforia antara lain:

1. Dapat mempercepat pendeteksian target dengan kapasitas *tracking* 5 target secara simultan (Amin & Govilkar, 2015).
2. Dapat *tracking* secara efisien dalam kondisi cahaya rendah dan meskipun target tertutup sebagian (Amin & Govilkar, 2015)

3. Dapat melakukan *Extended tracking*, yang memungkinkan aplikasi untuk menjaga target pelacakan dan membantu mempertahankan referensi yang konsisten untuk pembesaran objek bahkan ketika target tidak lagi terlihat secara waktu nyata kamera (Amin & Govilkar, 2015).

Dataset adalah sekumpulan benda yang dapat dikenali yang di *download* dari *target management system*. SDK memungkinkan aplikasi untuk memuat, mengaktifkan, menonaktifkan dan membongkar kelompok dataset pada saat *runtime*. Dataset dapat mengandung *image target* dan *multi target*. Pemuatan dataset membutuhkan beberapa waktu dan oleh karena itu dianjurkan menggunakan *background loading* (Ibañez & Figueras, 2013).

2.5 OBJECT ORIENTED

Object oriented adalah sebuah pendekatan untuk pengembangan suatu perangkat lunak dimana dalam struktur perangkat lunak tersebut terdapat kumpulan objek yang saling berinteraksi dalam suatu sistem. Pemrograman berorientasi objek diciptakan untuk mempermudah pengembangan program dengan cara mengikuti model yang sudah ada dalam kehidupan nyata. Beberapa pertimbangan *object oriented* (O'Docherty, 2005):

1. Objek lebih mudah dipahami orang, karena objek berasal dari bisnis yang dibangun secara otomatis, dibandingkan dipengaruhi oleh prosedur berbasis komputer atau persyaratan penyimpanan data. Contohnya, pada sistem bank, seseorang membuat program akun bank, bank *tellers* dan konsumen dahulu, dibandingkan langsung melihat catatan akun, prosedur mengambil dan menyeter.

2. Spesialis dapat berkomunikasi dengan lebih baik, semakin berkembang zaman, *software industry* meningkatkan karir setiap orang bersama dengan pengetahuan dan pengalaman mereka. Sebagai contoh, *programmer* akan memperbaiki *bug* pada *script code* yang dibuatnya sendiri. Kemudian, *senior programmer*, menulis kodenya sendiri. *Designer*, menentukan kode mana yang perlu ditulis, kemudian *analyst* akan berbicara dengan pelanggan untuk mengetahui apa yang dibutuhkan, dan kemudian menuliskan apa yang dapat dikerjakan oleh sistem yang sudah selesai.

3. Kode dapat digunakan kembali dengan mudah, dengan pendekatan secara tradisional, dimulai dengan masalah apa yang perlu diselesaikan dan membiarkan masalah itu mendorong pengembangan.

4. *Object oriented* sudah matang dan terbukti, ini bukan metode baru. Konsep pemrograman muncul pada akhir 1960-an sementara metodologi sudah ada setidaknya satu dekade. Menerapkan objek dalam bidang seperti perangkat lunak, basis data, dan jaringan kini dapat dipahami dengan baik.

2.6 RATIONAL UNIFIED PROCESS

Rational Unified Process (RUP) adalah sebuah proses pengembangan sistem yang preskriptif dan terdefinisi dengan baik yang sering digunakan untuk mengembangkan sistem berbasis obyek atau komponen berbasis teknologi. RUP didasarkan pada prinsip-prinsip rekayasa perangkat lunak seperti *iterative*, *requirements-driven*, dan *architecture-centric* pada pendekatan pengembangan perangkat lunak.

Daur proses pengembangan perangkat lunak dalam RUP terdiri dari empat tahapan yang saling berkesinambungan dalam waktu, yang disebut fase. Setiap fase memiliki target

yang harus dicapai. Fase-fase dalam RUP adalah sebagai berikut (Rational Software Corporation, 1998):

1. *Inception*

Membangun pemahaman yang baik tentang sistem apa yang akan dibangun dengan mendapatkan pemahaman tingkat tinggi dari semua persyaratan dan menetapkan ruang lingkup sistem. Mengurangi banyak risiko bisnis, menghasilkan kasus bisnis untuk membangun sistem, dan mendapatkan dukungan dari semua *stakeholder*.

2. *Elaboration*

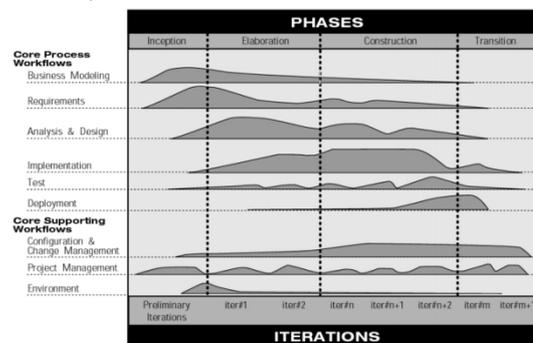
Mengurus banyak tugas yang paling sulit secara teknis: merancang, mengimplementasikan, menguji, dan membuat *baseline* arsitektur yang dapat dieksekusi, termasuk subsistem, antarmuka, komponen utama, dan mekanisme arsitektur, seperti cara menangani komunikasi antar proses atau persistensi. Mengatasi risiko teknis utama, seperti risiko pertentangan sumber daya, risiko kinerja, dan risiko keamanan data, dengan menerapkan dan memvalidasi kode aktual.

3. *Construction*

Melakukan sebagian besar implementasi saat beralih dari arsitektur yang dapat dieksekusi ke versi operasional pertama sistem. Menyebarkan beberapa rilis internal dan alfa untuk memastikan bahwa sistem dapat digunakan dan memenuhi kebutuhan pengguna. Akhiri fase dengan menggunakan versi beta yang berfungsi penuh dari sistem, termasuk pemasangan dan dokumentasi pendukung dan materi pelatihan (meskipun sistem kemungkinan masih akan membutuhkan penyesuaian fungsionalitas, kinerja, dan kualitas keseluruhan).

4. *Transition*

Memastikan bahwa perangkat lunak memenuhi kebutuhan penggunanya. Ini termasuk menguji produk dalam persiapan untuk rilis dan melakukan penyesuaian kecil berdasarkan umpan balik pengguna. Pada titik ini dalam siklus hidup, umpan balik pengguna berfokus terutama pada penyempurnaan produk, konfigurasi, instalasi, dan masalah kegunaan; semua masalah struktural utama seharusnya sudah dikerjakan lebih awal dalam siklus hidup proyek (Wesley, 2003).



Gambar 1 Diagram Fase RUP (Wesley, 2003)

3 BUSINESS MODELING AND REQUIREMENT

3.1 APLIKASI SEJENIS

Tahap analisis kebutuhan dilakukan untuk mengumpulkan data kebutuhan fungsional dari sistem. Proses pengumpulan kebutuhan fungsional dilakukan dengan riset pada beberapa aplikasi sejenis yang telah dibangun, kebutuhan fungsional dari setiap aplikasi dapat dilihat pada pembahasan ini.

3.1.1 CAMERA MX

Camera MX adalah aplikasi kamera berfitur lengkap yang menawarkan banyak opsi untuk mengambil gambar. Aplikasi ini memberi pengguna kontrol penuh terhadap resolusi dan memberikan visual yang jelas untuk pengambilan gambar yang maksimal. Dengan *Camera MX*, pengguna dapat membuat foto dan video animasi, dan juga menambahkan banyak efek, filter, bingkai,

dan lain-lain ketika mengedit hasil tangkapan foto. Fitur utamanya termasuk pemotretan langsung (*live shots*), *Graphics Interchange Format* (GIF), dan opsi fitur ‘*Shoot The Past*’, di mana pengguna dapat memilih momen yang tepat ketika mengambil foto. Aplikasi kamera ini tersedia secara gratis di *Google Play Store*, alamat URL aplikasi : https://play.google.com/store/apps/details?id=com.magix.camera_mx&hl=en

Fitur : Efek pada kamera, bingkai, dan *editing* pada gambar yang sudah diambil, *live shots*, *shoot the past*.

Kelebihan : Pada saat pengambilan gambar, dapat memberikan efek dan memberi bingkai, terdapat fitur *live shots* dan *shoot the past*.

Kekurangan : Tidak ada penambahan teknologi *augmented reality* dan tidak dapat langsung membagikan hasil ke sosial media.



Gambar 2 Aplikasi *Camera MX*

3.1.2 *CAMERA360*

Aplikasi ini menawarkan berbagai macam kamera yang dilengkapi dengan puluhan efek unik dari setiap jenis. Pengguna juga dapat mengintegrasikan efek langsung ke dalam gambar saat mengambilnya. Fitur lainnya termasuk stiker gerak, berbagai filter, editor foto, dan lain-lain. Aplikasi ini juga dapat memberikan *editing* untuk mencerahkan hasil gambar dan menghaluskan wajah. Aplikasi kamera ini tersedia secara gratis di *Google Play Store*, alamat URL aplikasi :

<https://play.google.com/store/apps/details?id=vStudio.Android.Camera360&hl=en>

Fitur : Efek pada kamera, bingkai, dan *editing* pada gambar yang sudah diambil.

Kelebihan : Pada saat pengambilan gambar, dapat memberikan efek dan memberi bingkai.

Kekurangan : Tidak ada penambahan teknologi *augmented reality*.



Gambar 3 Aplikasi *Camera360*

3.1.3 *CANDY CAMERA*

Aplikasi pengambilan gambar yang berfokus pada pengambilan foto *selfie*. Aplikasi kamera ini dilengkapi dengan banyak filter dan fungsi *beauty* yang mencakup alat rias wajah, efek pelangsingan wajah, *sticker*, dan lain-lain. Aplikasi ini juga dapat memberikan tambahan *sticker* dengan fitur *face recognition sticker*. Aplikasi ini juga dapat menggabungkan beberapa gambar menjadi satu dengan fungsi *collage*. Pengguna dapat memilih dari berbagai macam *grip* dan *styles*. Aplikasi kamera ini tersedia secara gratis di *Google Play Store*, alamat URL aplikasi :

<https://play.google.com/store/apps/details?id=com.joeware.Android.gpulumera&hl=en>

Fitur : Efek pada kamera, bingkai, dan *editing* pada gambar yang sudah diambil, dapat menggabungkan beberapa gambar menjadi satu dengan fungsi *collage*, *face recognition sticker*.

Kelebihan : Pada saat pengambilan gambar, dapat memberikan efek dan memberi bingkai. Dapat menambahkan *sticker* dengan fitur *face recognition sticker*. Setelah selesai mengambil gambar,

pengguna dapat menggunakan fitur *collage* untuk menggabungkan gambar.

Kekurangan : Tidak ada penambahan teknologi *augmented reality* dan tidak dapat langsung membagikan hasil ke sosial media.



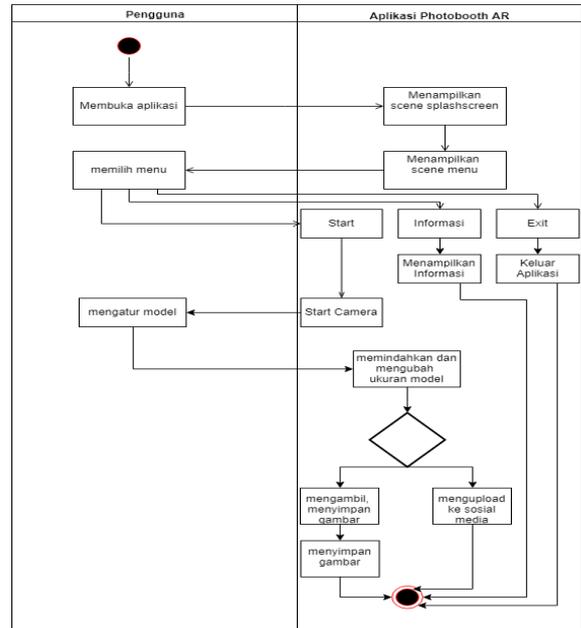
Gambar 4 Aplikasi Candy Camera

3.2 PEMODELAN BISNIS

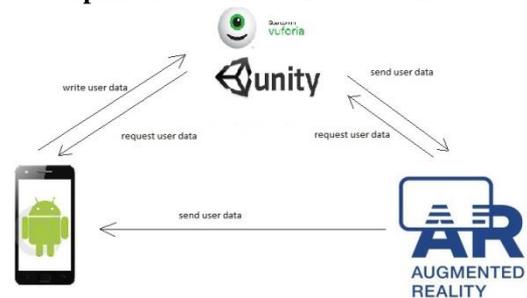
Berdasarkan kelebihan dan kekurangan dari beberapa aplikasi sejenis pada sub bab 3.1, masih sedikit sekali aplikasi pengambilan gambar yang menggunakan teknologi *augmented reality*. Maka dari itu, akan di buat aplikasi *photobooth* yang dapat menambahkan model AR 3D secara nyata dan dapat mengambil gambar serta mengunggahnya ke media sosial. *Activity* diagram aplikasi dapat dilihat pada gambar 5 dan arsitektur sistem dapat dilihat pada gambar 6.

Pada gambar 5, pertama kali pengguna membuka aplikasi, sistem akan menampilkan *scene splashscreen* yang akan menampilkan nama atau logo dari *developer* aplikasi. Setelah *countdown* selama 5 detik, aplikasi akan menampilkan halaman menu yang terdiri dari 3 pilihan, yaitu *start*, *information*, dan *exit*. Menu *information* akan menampilkan informasi dari aplikasi juga *credit* dari aplikasi. Menu *exit* akan menutup aplikasi. Menu *start*, akan memulai aplikasi, di mana sistem akan menyalakan kamera dan menampilkan *button capture*, *share*, juga *slider*. Pengguna kemudian mengatur model (memindahkan, mengubah ukuran model)

kemudian dapat mengambil gambar dan mengunggah ke media sosial.



Gambar 5 Activity Diagram Pengembangan Aplikasi Photobooth Berbasis AR



Gambar 6 Arsitektur Aplikasi Photobooth Berbasis AR

3.3 USE CASE

Setelah mendefinisikan kebutuhan, maka dalam subbab ini dilakukan pembuatan model *use case*. Model *use case* menghasilkan daftar aktor, dan daftar *use case*, yang menunjukkan hubungan antara aktor dengan *use case*.

Aktor merupakan pengguna yang akan berinteraksi dengan aplikasi. Aplikasi *photobooth* berbasis *augmented reality* memiliki 1 aktor, yaitu pengguna. Begitu aplikasi sudah selesai di kembangkan, maka admin akan melepas aplikasi tersebut sehingga hanya menyisakan pengguna

sebagai aktor yang mempunyai hak seutuhnya untuk menggunakan aplikasi tersebut, baik untuk mengambil gambar ataupun menyimpan gambar secara individual.

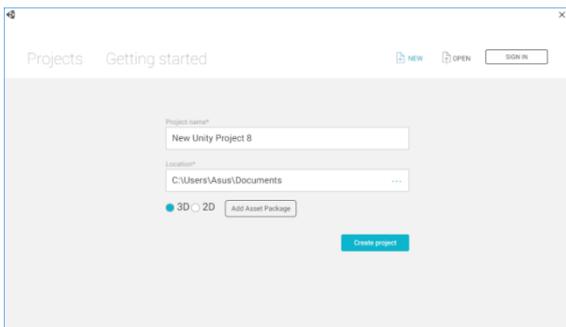


Gambar 7 Diagram Use Case Aplikasi *Photobooth* Berbasis AR

4 ANALYSIS, DESIGN, IMPLEMENTATION AND TESTING

Bagian ini menjelaskan tentang tahapan analisis, desain, implementasi dan testing pada pengembangn aplikasi *photobooth* berbasis AR. Sistem ini diimplementasikan dengan menggunakan bahasa pemrograman C# dan menggunakan perangkat lunak Unity dan *library* Vuforia.

Implementasi pengembangan aplikasi Gambar 8 sampai dengan Gambar 26.



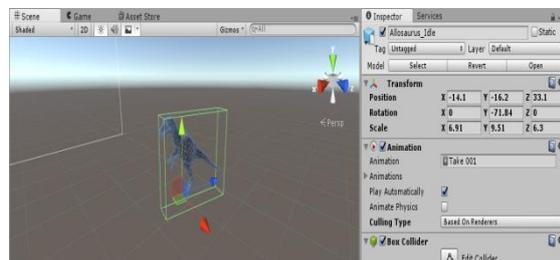
Gambar 8 Project baru pada Unity



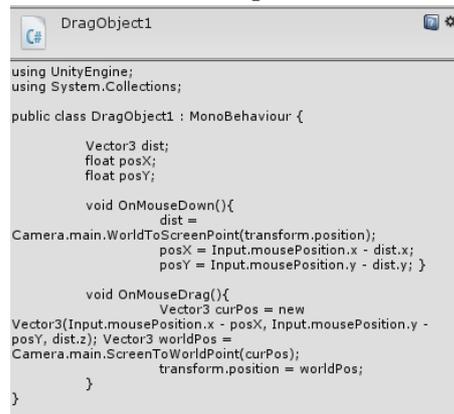
Gambar 9 Import Asset



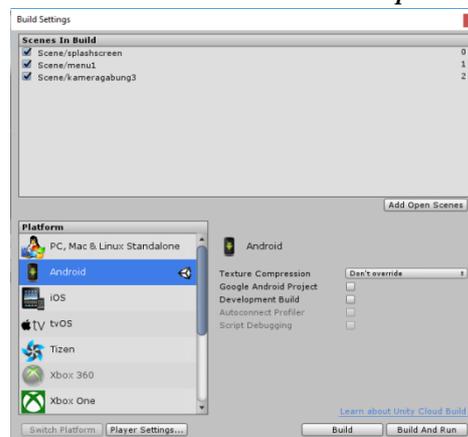
Gambar 10 Membuat Scene



Gambar 11 Menempatkan model AR



Gambar 12 Pembuatan Script

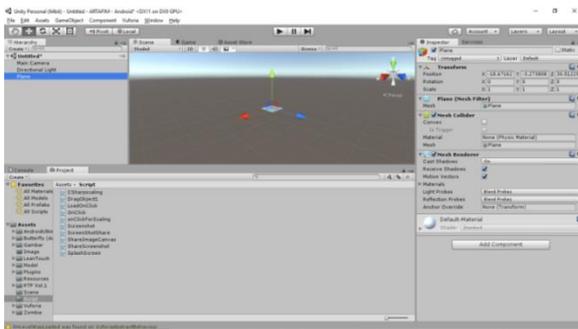


Gambar 13 Build Aplikasi

Pengembang aplikasi membuat project baru pada Unity, kemudian pengembang membuat alur bagaimana aplikasi akan

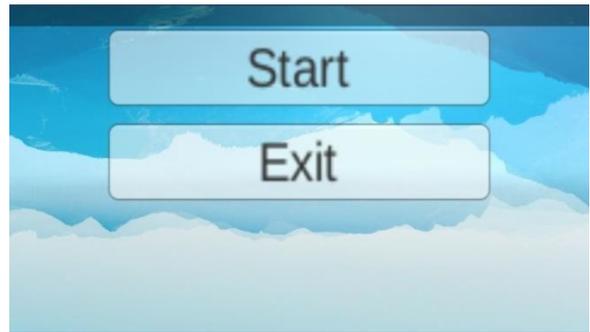
bekerja. Setelah membuat alur bagaimana aplikasi bekerja, pengembang membuat *scene* untuk tiap menu.

Scene splashscreen, pengembang memilih menu *GameObject*, dan memilih 3D *Object - plane*. *Plane* digunakan untuk meletakkan design dari *logo* atau gambar pengembang aplikasi yang bertujuan untuk memperlihatkan pengembang dari aplikasi tersebut. Pengembang kemudian meletakkan design logo, meletakkannya dibelakang kamera, dan mengatur besar dari logo. Pengembang kemudian meletakkan *script* yang digunakan untuk *me-load scene* berikutnya secara otomatis setelah 5 detik *scene splashscreen* ditampilkan.



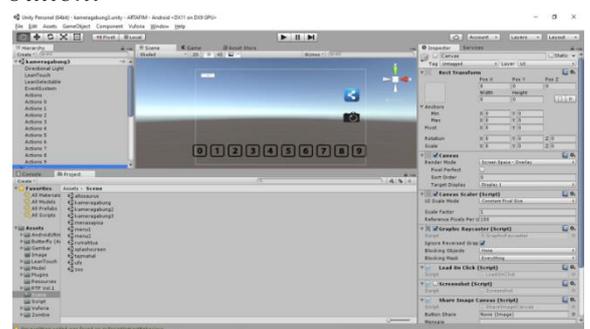
Gambar 14 Implementasi Antarmuka Halaman Kelola Tahun

Setelah membuat *scene splashscreen*, pengembang kemudian membuat *scene main menu*. Pengembang membuat *button* yang dapat di akses pada *GameObject - UI - button*, kemudian membuat 2 *button* yaitu untuk *Start* dan *Exit*. Pengembang kemudian mengatur ukuran dan posisi *button*, kemudian meletakkannya dibelakang kamera beserta *plane* yang digunakan untuk *background* dari *menu* tersebut. Setelah membuat dan mengatur *button*, kemudian pengembang memasukkan *script* ke dalam *object* yang digunakan untuk *me-load scene* yang akan dituju ketika *button* di sentuh.



Gambar 15 Pembuatan Menu

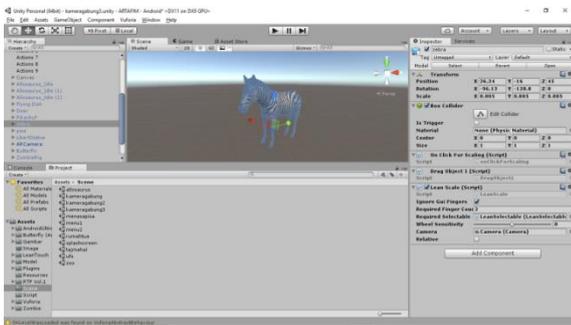
Pengembang kemudian membuat *menu* setelah *button start* disentuh, yaitu *menu* di mana kamera di akses yang akan menampilkan model AR. Pertama, pengembang menghapus *Camera* bawaan dari Unity pada *Hierarchy* dan kemudian meng-*drag ARCamera* dari *library Vuforia* ke *hierarchy*, *ARCamera* digunakan untuk menampilkan model AR. Kemudian pengembang membuat tampilan muka untuk *menu* ini (*button back*, *button capture*, *button share*, *slider* untuk model AR), kemudian sama seperti sebelumnya, pengembang meletakkan *script* pada *object* yang kemudian akan di *drag* pada masing-masing *button*.



Gambar 16 Membuat Interface

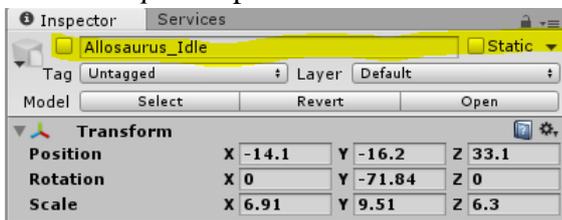
Pengembang kemudian meng-*drag* model AR ke *hierarchy*, karena aplikasi ini menggunakan *markerless* (tanpa penanda), pengembang harus meletakkan model AR dibelakang kamera agar model AR dapat muncul ketika kamera diakses dan *button* untuk menampilkan model AR di sentuh. (Jika menggunakan penanda, pengembang harus mengakses database Vuforia, untuk meletakkan model AR yang akan muncul beserta penanda yang akan digunakan untuk

meng-*scan* penanda tersebut agar model AR keluar).



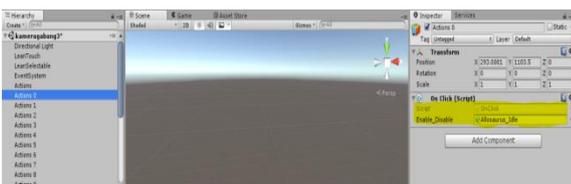
Gambar 17 Pengaturan Model AR

Pengembang kemudian membuat *create Empty*, yang dapat di akses melalui *GameObject – Create Empty*. *Empty* ini digunakan untuk melakukan aksi ketika button yang digunakan untuk menampilkan model AR disentuh. Agar model AR tidak langsung muncul begitu kamera di akses, maka model AR yang sudah di *drag* ke *hierarchy* harus di matikan terlebih dahulu dengan menghilangkan tanda centang melalui *inspector* pada tab sebelah kanan.



Gambar 18 Menonaktifkan Model AR

Pengembang kemudian men-*drag* model AR yang sudah dimatikan ke dalam *Empty* yang sudah dibuat tadi beserta script *OnClick* yang berguna untuk menampilkan model AR ketika *button* disentuh. Pengembang kemudian mengulangi hal yang sama untuk ke-10 model AR.



Gambar 19 Meletakkan Model AR dan Script Pada Empty

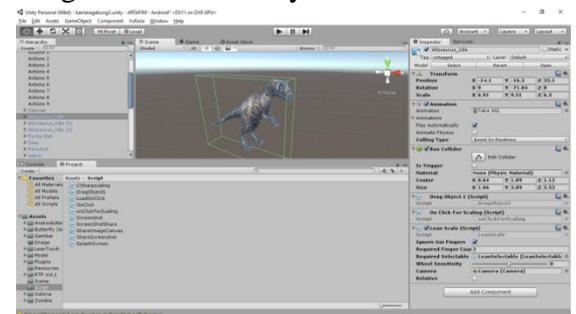
Pengembang kemudian membuat *Rect Transform* yang dapat di akses pada *Component – Layout – Rect Transform*. *Rect*

Transform digunakan untuk membuat slider sehingga banyaknya model AR dapat memenuhi layar. Pengembang mengatur besar dari area yang dapat di *slide* pada *content*, kemudian meletakkan ke-10 button model AR pada 1 *parent content*, kemudian meletakkan *content* pada *parent image* beserta *script scroll rect*.



Gambar 20 Pembuatan Slider

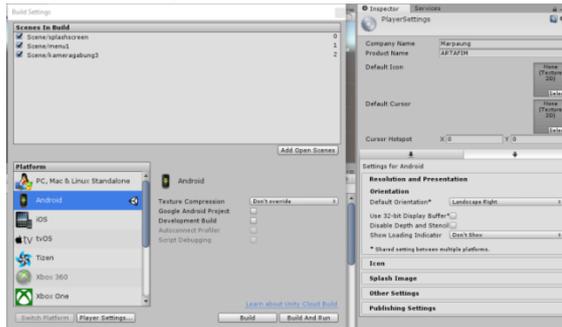
Pengembang kemudian membuka kembali model AR dan meng-*drag* *script drag object*, *on click for scaling*, dan *lean scale*. *Script drag object* digunakan untuk memindahkan model AR, *script on click for scaling* digunakan untuk mengatur besar dari model AR, dan *script lean scale* digunakan untuk mengatur besar dari model AR dengan menggunakan 2 jari. Sama seperti sebelumnya, *script* ini juga dibuat *Create Empty* sebagai aksi ketika perintah dijalankan. Pengembang membuat 2 *empty*, yaitu *Lean Touch* untuk perintah ketika disentuh dan *Lean Selectable* untuk memilih model AR. Pengembang kemudian memilih *Add component* pada tab inspector sebelah kanan, kemudian memilih *box collider*. *Box collider* digunakan untuk membuat area yang digunakan untuk memindahkan dan mengatur besar kecilnya model AR.



Gambar 21 Pembuatan Box Collider dan Meletakkan Script

Pengembang kemudian mengatur pengaturan dari aplikasi ini dengan mengakses *File –*

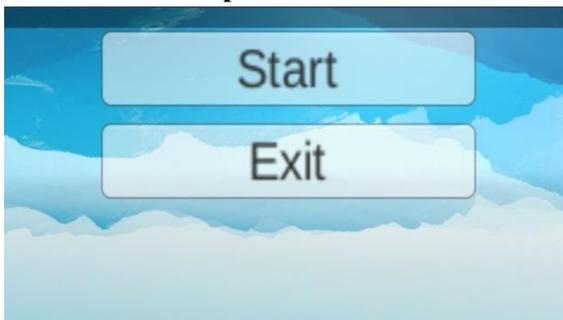
Build Setting. Pada pengaturan ini, pengembang dapat memilih *platform* yang dapat digunakan, kemudian mengatur resolusi, orientasi, nama perusahaan, nama produk, *icon* dari aplikasi dan lain-lain. Setelah selesai, pengembang dapat menekan *button build*, sehingga Unity akan menjadikan aplikasi dalam bentuk APK.



Gambar 22 Pembuatan Aplikasi Menjadi APK



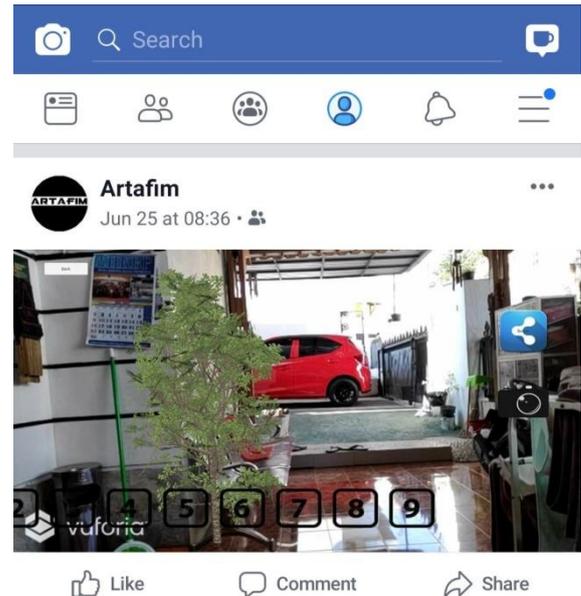
Gambar 23 Implementasi Antarmuka Scene Splash Screen



Gambar 24 Implementasi Antarmuka Scene Main Menu



Gambar 25 Hasil Model AR



Gambar 26 Implementasi Antarmuka Share to Social Media

5 KESIMPULAN DAN SARAN

5.1 KESIMPULAN

Kesimpulan yang dapat diambil dari penelitian tugas akhir ini adalah sebagai berikut.

1. Aplikasi ini di kembangkan pada perangkat Android sehingga memberikan fleksibilitas pada pengguna untuk melakukan pengambilan gambar di mana saja dan kapan saja dengan tambahan objek 3 dimensi, aplikasi akan di tampilkan dengan *landscape* kanan. Pembobotan dengan metode AHP yang menggunakan model hierarki berhasil memaparkan perbedaan tingkat kepentingan masing-masing kriteria.
2. Aplikasi ini dapat di *install* pada *smartphone* Android dengan *minimal requirement operating system* KitKat versi 4.4.2, sedangkan jika aplikasi ini di *install* pada *smartphone* Android dengan *operating system* sebelum KitKat versi 4.4.2, maka aplikasi tidak dapat di-*install* (*not supported*).
3. Aplikasi ini dapat menyajikan tampilan secara maksimal (tampilan *button* terlihat

semua, tidak terlihat setengah) apabila di *install* pada *smartphone* Android dengan resolusi 1080 x 1920 *pixels*.

5.2 SARAN

Saran yang dapat diberikan untuk penelitian selanjutnya adalah aplikasi dapat di-*build* untuk mode *portrait* juga *landscape* (*auto rotate*), sehingga pengambilan gambar dapat mengikuti keinginan pengguna dan menambah variasi pengambilan gambar.

DAFTAR PUSTAKA

- Amin, D., & Govilkar, S. (2015). Comparative Study Of Augmented Reality SDK's. *Computer Sciences & Applications (IJCSA)*, *V(1)*, pp. 11-26.
- Azuma, R. T. (1997). A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments*, *6(4)*, 355-385.
- Blackman, S. (2011). *Beginning 3D Game Development with Unity*. New York: Apress.
- Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The Unified Modeling Language User Guide* (2nd Edition ed.). Addison Wesley Professional.
- Creighton, R. H. (2011). *Unity 3.x Game Development by Example Beginners Guide*. Packt Publishing .
- Goldstone, W. (2009). *Unity Game Development Essentials*.
- Haller, M. B. (2006). Emerging Technologies of Augmented Reality: Interfaces and Design. *Hershey: IGI Publishing, (Edited Volumes)*, xiii+399pp.
- Ibañez, A. S., & Figueras, J. (2013). *Vuforia v1.5 SDK: Analysis And Evaluation Of Capabilities*. Escola d'Enginyeria de Telecomunicacio i Aeroespacial de Castelldefels. Master Thesis.
- Laganière, T. C. (2005). *Object-Oriented Software Engineering*. Shoppenhangers Road: McGraw-Hill Education.
- Madden, L. (2011). *Professional Augmented Reality Browsers for Smartphones*. (L. a. Wikitude, Ed.) Wrox. USA: Programming for junaio.
- O'Docherty, M. (2005). *Object-Oriented Analysis & Design Understanding System Development with UML 2.0*. England: John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex.
- Qualcomm. (2012). *Augmented Reality on Android*. Retrieved September Senin, 09, 2016, from https://myvodafone.secure.vodafone.co.za/cs/groups/public/documents/vodafone.co.za_portal_webassets/pocm01-521833.pdf
- Rational Software Corporation. (1998). *Rational Unified Process: Best Practices for Software Development Teams*. Lexington.
- Roedavan, R. (2014). *Unity Tutorial Game Engine*. Bandung: Informatika.
- Shapiro, L. &. (2001). Computer Vision. (P. Hall, Ed.) *Prentice Hall, 1st edition*, 530-532.
- Wesley, A. (2003). *Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*.