BAB II
TINJAUAN PUSTAKA

2.1 Teori Judul

2.1.1 Pengertian Penerapan

Menurut Ali (2017:51), "Implementasi (penerapan) merupakan penyediaan sarana untuk melaksanakan sesuatu yang menimbulkan dampak atau akibat terhadap sesuatu".

Sedangkan menurut Browne dan Wildavsky dikutip oleh Arinda (2018:19), "Implementasi (Penerapan) adalah perluasan aktivitas yang saling menyesuaikan."

Dari pengetian-pengertian diatas, dapat disimpulkan bahwa penerapan merupakan suatu kegiatan yang meniru suatu teori dalam bentuk gerakan atau aktivitas untuk mencapai suatu hasil.

2.1.2 Pengertian Teknologi Informasi

Menurut Pratiwi (2017:104), "Teknologi informasi adalah gabungan antara teknologi komputer dan telekomunikasi. Teknologi Informasi memanfaatkan komputer elektronik dan perangkat lunak komputer untuk mengubah, menyimpan, melindungi, memproses, mentransmisikan, dan memperoleh informasi secara aman".

Suratno (2016: 93), "Teknologi informasi adalah suatu teknologi yang berhubungan dengan pengolahan data menjadi informasi dan proses penyaluran data dan informasi tersebut dalam batas-batas ruang dan waktu".

Berdasarkan pengertian tersebut dapat disimpulkan Teknologi Informasi adalah suatu teknologi pengolahan yang dapat mengubah,menyimpan, melindungi, memproses, mentransmisikan, dan memperoleh informasi secara aman.

2.1.3 Pengertian Augmented Reality

Menurut Pamoedji, Maryuni, Sanjaya (2017:2) menyatakan, "Sebuah teknik yang menggabungkan benda maya dua dimensi maupun tiga dimensi ke dalam sebuah lingkup nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata."

Menurut Shumaker dan Lackey (2014:16) menjelaskan, "Teknologi *Augmented Reality* (AR) didasarkan pada informasi virtual yang tumpang tindih di ruang nyata. Teknologi AR memungkinkan untuk mencampur objek virtual yang dihasilkan oleh komputer dengan lingkungan nyata, menghasilkan lingkungan campuran yang dapat dilihat melalui perangkat teknologi apa pun secara real time."

Sedangkan menurut Dr. Grasset (2013:5), *"Augmented Reality* (AR) adalah penciptaan atau modifikasi dari realita kita, diperkaya dengan informasi digital (atau virtual), di layar komputer ataupun di perangkat seluler."

Dari pengertian-pengertian diatas, dapat disimpulkan bahwa *Augmented Reality* adalah teknologi yang menggabungkan gambar virtual 3D kedalam dunia nyata dan dapat digunakan dalam berbagai macam bidang ilmu.

2.1.4 Pengertian Promosi

Menurut KBBI, Promosi adalah perkenalan dalam rangka memajukan usaha, dagang, dan sebagainya.

2.1.5 Pengertian Rumah

Menurut Undang-undang Republik Indonesia Bab I (2011:3), "Rumah adalah bangunan gedung yang berfungsi sebagai tempat tinggal yang layak huni, sarana pembinaan keluarga, cerminan harkat dan martabat penghuninya, serta aset bagi pemiliknya."

Sedangkan Menurut Wicaksono (2009:2), "Rumah tidak hanya sekedar tempat berlindung dari hujan dan terik matahari, tetapi juga symbol status sosial bagi pemiliknya dan sumber inspirasi."

Dari pengertian-pengertian diatas, dapat disimpulkan bahwa rumah berfungsi sebagai bangunan yang di tinggali dan juga sebagai status social pemiliknya.

2.1.6 Pengertian Android

Menurut Kadir (2018:1), "Android adalah system operasi berbasis linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan computer tablet."

Menurut Safaat (2015:1), "Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencap sistem operasi, *middleware* dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka.

Berdasarkan pengertian tersebut dapat disimpulkan *Android* adalah sistem operasi untuk *mobile* yang berbasis *linux* yang dirancang sebagai telepon pintar.

2.1.7 Pengertian Prototype

Menurut Rosa dan Shalahuddin (2016:31) "Prototype adalah model yang dapat digunakan untuk menyambung ketidakpahaman pelanggan mengenai hal teknis dan memperjelas spesifikasi kebutuhan yang di inginkan pelanggan kepada pengembang perangkat lunak."

2.1.8 Pengertian Aplikasi Penerapan Teknologi *Augmented Reality* Sebagai Media Promosi Penjualan Rumah Berbasis Android Menggunakan Metode *Prototype*

Aplikasi Penerapan Teknologi *Augmented Reality* Sebagai Media Promosi Penjualan Rumah Berbasis Android Menggunakan Metode *Prototype* merupakan aplikasi yang bertujuan untuk membantu perumahan Bukit Mas Residence dalam memasarkan perumahan. Aplikasi dapat menampilkan perumahan dalam bentuk 3D secara *real-time*.

2.2 Teori Khusus

2.2.1 Metode Transformasi Geometri

Menurut Menurut Sembiring Suwah dan Marsito (2017:111), Transformasi Geometri adalah suatu pemetaan dari bidang Kartesius ke bidang tersebut, yang disajikan dalam geometri analitik atau geometri koordinat.

Ada dua macam transformasi yaitu transformasi dua dimensi dan transformasi tiga dimensi. Pada Augmented Reality menggunakan transformasi tiga dimensi. Pada dasarnya yang membedakan antara transformasi dua dimensi dengan tiga dimensi adalah pada transformasi tiga dimensi terdapat satu buah sumbu tambahan yang mewakili jarak pandanf kedalam. Sumbu ini biasa disebut dengan sumbu z.

Transformasi tiga dimensi dapat berupa rotasi, Berbeda dengan rotasi dua dimensi yang menggunakan titik pusat (0, 0) sebagai pusat perputaran, rotasi pada objek tiga dimensi menggunakan sumbu koordinat sebagai pusat perputaran. Karena pada objek tiga dimensi terdapat 3 sumbu koordinat, maka terdapat 3 macam rotasi yang dapat dilakukan, yaitu:

a. Rotasi sumbu x

$$Rx = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & Cos(\theta) & Sin(\theta) & 0 \\ 0 & -Sin(\theta) & Cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

b. Rotasi sumbu y

$$Ry = \begin{pmatrix} Cos(\theta) & 0 & -Sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ Sin(\theta) & 0 & Cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

c. Rotasi sumbu z

$$Rz = \begin{pmatrix} Cos(\theta) & Sin(\theta) & 0 & 0 \\ -Sin(\theta) & Cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotasi searah jarum jam jika nilai θ negatif, dan berlawanan arah jarum jam jika nilai θ positif. Titik hasil rotasi diperoleh dari :

Q = P.M

Q = [Xq, Yq, Zq, S] ; matriks 1x4 berisi titik hasil rotasi

P = [Xp, Yp, Zp, S] ; matriks 1x4 berisi titik yang akan dirotasi

(S adalah skala global bernilai 1)

M = Matriks 4x4, seperti berikut :

$$M = \left(\begin{array}{cccc} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \\ \hline m_{30} & m_{31} & m_{32} & m_{33} \end{array}\right) \rightarrow \left(\begin{array}{cc} R_{3x3} & P_{3x1} \\ \hline f_{1x3} & 1x1 \end{array}\right)$$

R_{3x3} = matriks rotasi

P_{3x1} = vektor posisi asal sistem koordinat rotasi

f_{1x3} = transformasi perspektif

1x1 = faktor skala global

2.2.2 *Unified Modeling Language*

Menurut Sukamto (2018:13), *Unified Modeling Language* (UML*)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

2.2.3 Class Diagram

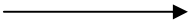
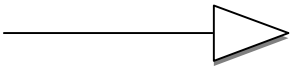
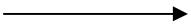
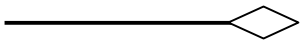
Menurut Salahuddin (2014) Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebuat atribut dan metode atau operasi.

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

Tabel 2.1 Simbol Class Diagram

No.	Simbol	Deskripsi
1	Kelas nama_kelas + atribut + operasi()	Kelas pada struktur sistem
2	Antarmuka / *interface* name_interface	sama dengan konsep *interface* dalam pemograman berorintasi objek.

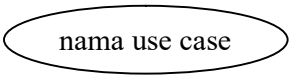
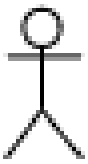
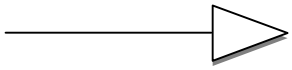
Lanjutan Tabel 2.1 Simbol Class Diagram

No.	Simbol	Deskripsi
3	Asosiasi / *association*	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity.*
4	Asosiasi berarah / *directed association* 	Relasi antarkelas dengan makna kelas yang saat digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.*
5	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6	Kebergantungan / *dependency* 	Relasi antarkelas dengan makna kebergantungan antar kelas.
7	Agregasi / *aggregation* 	Relasi antarkelas dengan makna semua-bagian (*whole-part)*

2.2.4 Use Case Diagram

Menurut Salahuddin (2014) *Use case* atau diagram *use case* merupakan pemodelan untuk kelakukan (*behavior)* sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor denagan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

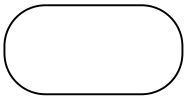
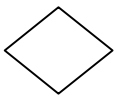
Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Deskripsi
1	*Use case* 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan anta unit atau aktor; biasanya dinyatakan dengan menggunkana kata kerja di awal fase nama *use case*
2	Aktor / *actor* 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunkanan kata benda di awal fase nama aktor
3	Asosiasi / *association* 	Komunikasi antara aktor dan *use case* yang berpatisipasi pada *use case* atau *use case* memiliki interkasi dengan aktor
4	Ekstensi / *extend* < < extend> > 	Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip *ingeritance* pada pemograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case*
5	Generalisasi / *generalization* 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	Menggunakan / *include* / *uses* < < include > > 	Relasi *use case* tambahan ke sebuah *use case* di mana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini

2.2.5 Activity Diagram

Menurut Salahuddin (2014) Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Tabel 2.3 Simbol Activity Diagram

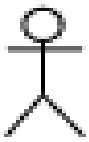
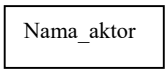
No	Simbol	Deskripsi
1	Start / status awal (*Intial State)*	Start atau intial state adalah state atau keadaan awal pada saat sistem mulai hidup
2	End / status akhir (*final state)*	End atau final state adalah state keadaan akhir dari daur hidup suatu sistem.
3	Event	Event adalah kegiatan yang menyebabkan berubahnya status mesin.
4	State 	Sistem pada waktu tertentu. State dapat berubah jika ada event tertentu yang memicu perubahan tersebut.
5	Decision 	Decision mengambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu.

2.2.6 Sequence Diagram

Menurut Salahuddin (2014) Diagram sekuen menggambarkan kelakukan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case berserta metode-metode yang dimiliki kelas yang diintansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada use case.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak use case yang didefiniskan maka diagram sekuen yang harus dibuay juga semakin banyak.

Tabel 2.4 Simbol Sequence Diagram

No	Simbol	Deskripsi
	Aktor  atau 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibua itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanyannya dinyatakan menggunakan kata benda di awal frase nama aktor.

2.3 Teori Program

2.3.1 Pengertian Bahasa C#

Menurut Yulius (2013:1) C# Adalah "bahasa pemrograman *object oriented* dan memiliki *class library* yang sangat lengkap yang berisi prebuilt component, sehingga memudahkan programmer untuk men-*develop* program lebih cepat." Basis dari Bahasa pemrograman C# sebenarnya adalah C++, Karena banyaknya kekurangan dan kelemahan dari bahasa pemrograman C++, maka akhirnya dibuat

bahasa pemrograman Java, dengan fitur dan model yang lebih kompleks dari C++. Java memiliki banyak peminat, Akan tetapi setelah diketahui bahwa ternyata dalam Bahasa pemrograman Java juga masih memiliki banyak kelemahan, akhirnya java dikembangkan menjadi C#, dimana dalam bahasa pemrograman C# tersusun dari 70% Java, 20% C++ dan 10% baru. Sehingga hal ini dapat menciptakan suatu bahasa pemrograman baru yang menutup kelemahan C++, maupun Java.

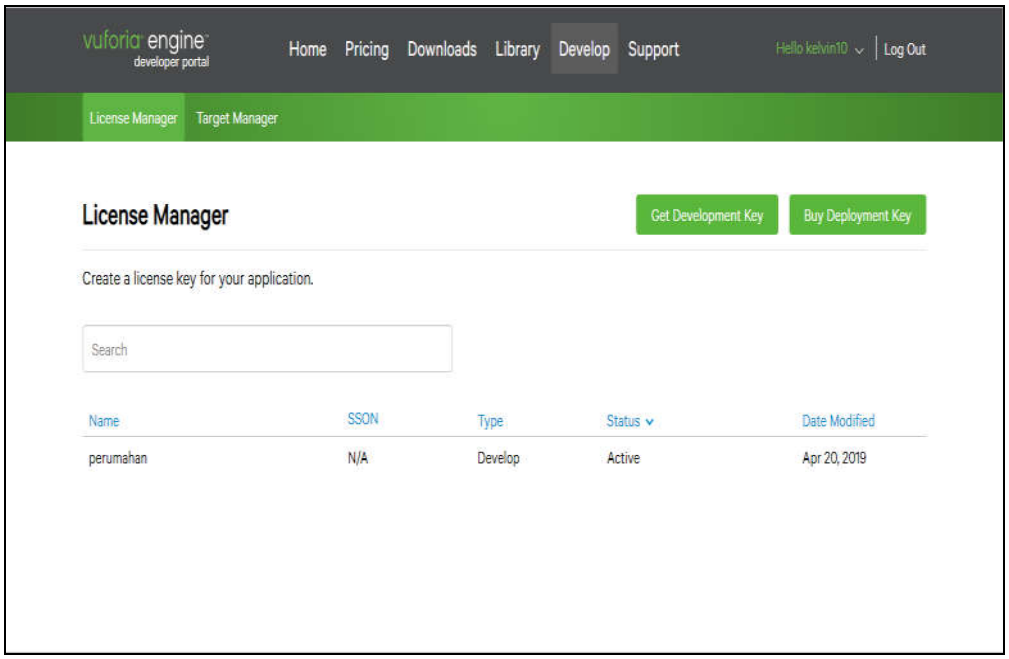
Gambar 2.1 Bahasa C#

2.3.2 Pengertian *Vuforia*

Menurut Andre Kurniawan Pamoedji, Maryuni dan Ridwan Sanjaya (2017:4) "Vuforia adalah salah satu *plugin* yang dapat membantu Anda dalam membuat sebuah aplikasi *Augmented Reality*." Aplikasi Vuforia menggunakan peralatan mobile sebagai layer untuk melihat kedalam dunia *Augmented* sehingga dunia nyata dan virtual dapat terlihat secara bersamaan. Kelebihan dari Vuforia SDK yaitu deteksi objek dapat secara local dan *cloud* melalui internet, dapat mengenali lebih dari 1 juta *image target* secara simultan, pelacakan bersifat *robust tracking* (*Augmentation* melekat pada objek sehigga tidak mudah hilang).

Tahapan penting dalam membuat aplikasi AR dengan Vuforia yaitu mengunggah *image target* atau *target word* untuk dijadikan target objek yang akan dilacak. *Image target* dapat diakses dengan aplikasi mobile dengan 2 cara :

1. Akses dari sebuah *cloud target database* menggunakan layanan web.
2. Mengunduh dalam sebuah *device target database* untuk di-*bundle* dengan aplikasi.



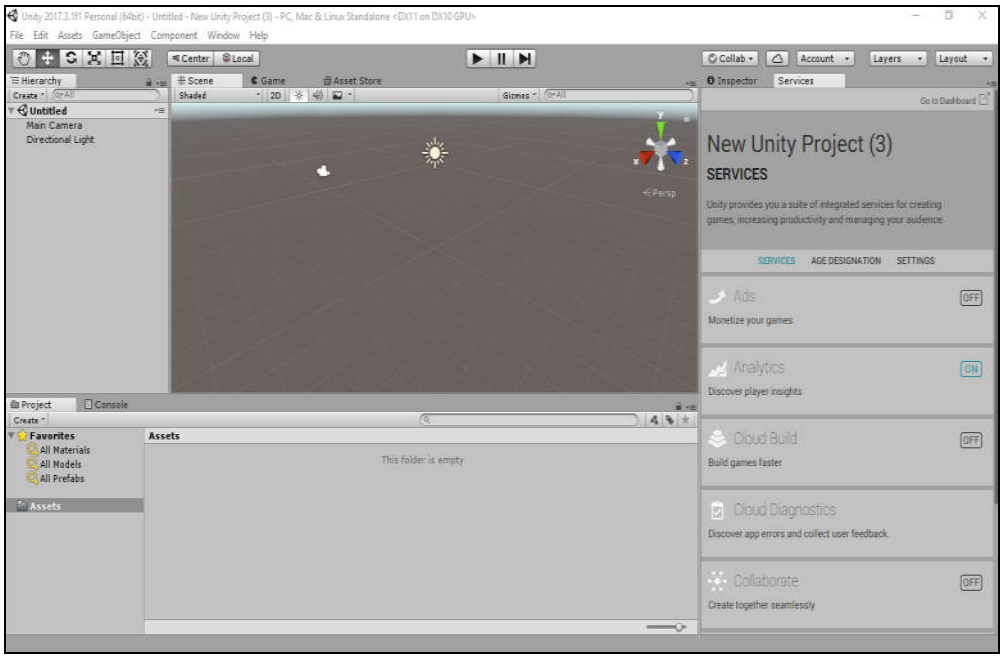
Gambar 2.2 User interface Vuforia

2.3.3 Pengertian *Unity* 3D

Menurut Andre Kurniawan Pamoedji, Maryumi dan Ridwan Sanjaya (2017:15) "Unity 3D adalah sebuah software pemrograman yang digunakan untuk membuat berbagai macam aplikasi." Unity 3D adalah sebuah game engine yang berbasis multiplatform. Unity dapat digunakan untuk membuat sebuah game yang bisa di gunakan pada perangkat computer, smartphone, iPhone, PS3, Bahkan X-BOX. Unity merupakan sebuah tool yang terintegrasi untuk membuat game, arsitek bangunan dan simulasi.

Unity 3D ini dikembangkan oleh Unity Technologies yang dibangun di tahun 2004 oleh David Helgason, Nicholas Francis dan Joachim Ante. Game engine ini di bangun atas kepedulian mereka terhadap indie developer yang tidak bisa membeli game engine karena terlalu mahal. Fokus perusahaan ini adalah membuat sebuah perangkat lunak yang bisa di gunakan oleh semua orang, khususnya untuk membangun sebuah game. Pada tahun 2009, Unity di luncurkan secara gratis dan pada April 2012 Unity mecapai popularitas tertinggi dengan lebih dari satu juta developer terdaftar di seluruh dunia.

Unity juga memiliki IDE (Integrated Development Environment) yaitu MonoDevelop yang bertujuan untuk menginteregrasikan semua script yang di buat kedalam unity, sehingga dapat langsung diproses. Game engine unity dikembangkan dengan menggunakan Bahasa pemrograman C#/C++ dan juga mendukung berbagai macam Bahasa pemrograman yang lainnya seperti Javascript.



Gambar 2.3 User interface Unity