

PERANCANGAN PERANGKAT LUNAK REALITAS MAYA KAMPUS TEKNIK ELEKTRO UNIVERSITAS DIPONEGORO BERBASIS C#

Lukas Aditratika^{*)}, Kodrat Iman Satoto, and R. Rizal Isnanto

Jurusan Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

^{*)}E-mail: lukas_aditratika@hotmail.com

Abstrak

Pada kampus Jurusan Teknik Elektro Universitas Diponegoro terdapat dua buah gedung utama. Seringkali para pengunjung, mahasiswa baru, tamu-tamu, dan mereka yang baru pertama kali datang ke kampus Teknik Elektro Universitas Diponegoro, mengalami kesulitan dalam hal navigasi dan mencari lokasi dari ruang-ruang yang ada di gedung kampus tersebut dikarenakan banyaknya ruangan dan lantai di tiap-tiap gedung tersebut. Salah satu cara untuk mempermudah mereka menemukan lokasi yang mereka cari adalah dengan membuat sebuah perangkat lunak realitas maya yang dapat memberi navigasi dan menunjukkan tempat yang dimaksud dengan mudah. Studi ini menghasilkan sebuah perangkat lunak yang mampu membantu pengguna dalam menjelajah kampus Jurusan Teknik Elektro Universitas Diponegoro secara realitas maya. Perangkat lunak ini dilengkapi dengan fitur jelajah dan jelajah otomatis yang menjawab kebutuhan utama penjelajahan kampus Jurusan Teknik Elektro Universitas Diponegoro.

Kata kunci: perangkat lunak, realitas maya, C#, Unity3D

Abstract

There are two main buildings in Electrical Engineering Department in Diponegoro University. Often, visitors, new students, guests, and those who first come to Electrical Engineering buildings, feel hard to navigate, and find locations of the rooms they are looking for, because of the fact that there are abundant numbers of rooms and floors in each of the buildings. One of the ways to ease those who find difficulties in finding locations in Electrical Engineering Building, is to design a virtual reality software that enables the users to navigate and find the places easily. This study created a virtual reality software that is able to help users navigate around Electrical Engineering Department Buildings in Diponegoro University. This software is equipped with navigation and auto-navigation features which answer the primary needs of navigation in Electrical Engineering Department building in Diponegoro University.

Keywords: software, virtual reality, C#, Unity3D

1. Pendahuluan

Seiring dengan kemajuan teknologi informasi, penggunaan sistem informasi berbasis jaringan sangat diminati. Salah satu layanan berbasis jaringan yang diberikan di kampus Teknik Elektro Universitas Diponegoro adalah Sistem Informasi Akademik (SIA) yang memudahkan mahasiswa dan dosen dalam proses administrasi seperti pengisian kartu rencana studi, penyajian transkrip nilai, pencarian data mahasiswa dan dosen serta beberapa fasilitas penunjang seperti info beasiswa, lowongan pekerjaan, dan lain sebagainya. Namun begitu, dengan segala layanan yang terus ditingkatkan, belum ada fasilitas dalam sistem informasi tersebut yang memudahkan para pengunjung *website*

Teknik Elektro Universitas Diponegoro untuk benar-benar dapat melihat seperti apa kira-kira bentuk fisik dari gedung Teknik Elektro Universitas Diponegoro selain hanya sebatas foto-foto yang terdapat di *website* tersebut. Di kawasan Jurusan Teknik Elektro pula, pengunjung kampus Jurusan Teknik Elektro Universitas Diponegoro seringkali mengalami kesulitan mencari sebuah lokasi ruangan saat berada di sana untuk pertama kalinya.

Dalam Penelitian ini dirancang sebuah perangkat lunak realitas maya yang mampu menjawab keingintahuan mereka yang dengan keterbatasan waktu dan tempat tidak bisa berada di kampus Teknik Elektro Universitas Diponegoro untuk nantinya bisa melihat atau menjelajah

kampus Teknik Elektro kita ini secara realitas maya sehingga mereka memiliki gambaran yang jelas akan apa saja yang bisa ditemukan di kampus Teknik Elektro ini.

2. Metode

2.1. Realitas Maya

Realitas maya atau *virtual reality* (VR) adalah teknologi yang membuat pengguna dapat berinteraksi dengan suatu lingkungan yang disimulasikan oleh komputer (*computer-simulated environment*), suatu lingkungan sebenarnya yang ditiru atau benar-benar suatu lingkungan yang hanya ada dalam imajinasi. Lingkungan realitas maya terkini umumnya menyajikan pengalaman visual, yang ditampilkan pada sebuah layar komputer atau melalui sebuah penampil stereoskopik.

Para pemakai dalam perangkat lunak realitas maya dapat saling berhubungan dengan suatu lingkungan sebetulnya atau sebuah artifak maya baik melalui penggunaan alat masukan baku seperti keyboard dan mouse, atau melalui alat multimodal seperti: sarung tangan terkabel dan ban jalan segala arah. Lingkungan yang ditirukan dapat menjadi mirip dengan dunia nyata, sebagai contoh, simulasi untuk pilot atau pelatihan pertempuran, atau dapat sangat berbeda dengan kenyataan, seperti di permainan realitas maya. Dalam praktik, sekarang ini sangat sulit untuk menciptakan pengalaman realitas maya dengan kejernihan tinggi, karena keterbatasan teknis atas daya proses, resolusi citra dan lebar pita komunikasi. Bagaimanapun, pembatasan itu diharapkan untuk secepatnya diatasi dengan berkembangnya pengolah, pencitraan dan teknologi komunikasi data yang menjadi lebih hemat biaya dan lebih kuat dari waktu ke waktu.

Penggunaan realitas maya didalam kehidupan sehari-hari memiliki potensial diantaranya dalam aplikasi pusat pengunjung (*visitor center applications*) sebuah gedung, dimana pengunjung dapat merasakan pengalaman *real-time* dalam waktu yang bersamaan.

2.2. Unity3D

Unity adalah perangkat lunak terintegrasi yang digunakan untuk membuat atau membangun permainan video tiga dimensi atau konten-konten interaktif lainnya seperti: visualisasi bangunan, atau animasi tiga dimensi *real-time*. Unity dapat berjalan baik di Microsoft Windows maupun di komputer dengan MacOS. Dan hasilnya dapat berjalan di Windows, Mac, Xbox 360, PlayStation 3, Wii, iPad, iPhone, Android, dan juga Linux. Unity juga dapat menghasilkan aplikasi *browser* dengan *plugin* Unity Web Browser yang mendukung Windows dan Mac. Bahasa pemrograman yang digunakan dalam Unity beragam mulai dari C++, JavaScript, Boo, dan C#. Berikut ini merupakan penjabaran terminologi yang terdapat dalam Unity.

2.2.1. Projects

Sebuah aplikasi yang dikembangkan dalam Unity akan selalu terdiri dari *Projects*. *Projects* ini akan menampung segala elemen dari *projects* seperti model, *script*, level, menu, *scene*, dan lain-lain.

Setiap *Project* terdiri dari satu adegan atau lebih. Di dalam adegan ini akan terdiri dari satu level aplikasi tetapi banyak elemen utama antarmuka pengguna, seperti menu dan teks.

2.2.2. GameObjects

Sebuah *GameObject* adalah bangunan dasar dari Unity. Sebuah *GameObject* adalah sebuah wadah bagi bermacam bentuk fungsionalitas yang berbeda yang disebut *Component*. Sebuah *GameObject* umumnya akan terdiri dari lebih dari satu buah *Component*. Semua *GameObject* memiliki sebuah *TransformComponent* yang menentukan posisi dan orientasinya.

Sebuah *GameObject* memiliki kekuatan untuk dapat menampung sebuah *GameObject* lain, yang berfungsi seperti parenting dalam sebuah hierarki. Sebuah *GameObject* yang terdapat didalam sebuah *GameObject* lain dinamakan *child GameObject*.

2.2.3. Components

Components adalah bagian dari sebuah *GameObject*. Tanpa sebuah *Component*, sebuah *GameObject* akan tampak kurang menarik. Sebuah *Component* dapat mewakili sebuah entitas yang dapat terlihat, misalnya: *meshes*, material, ataupun *particle system*. Jenis *Component* yang lain juga bisa dalam bentuk abstrak, misalnya kamera dan pencahayaan, yang mana tidak memiliki model secara fisik untuk direpresentasikan.

2.2.4. Assets

Assets merupakan bagian-bagian dari object yang dapat kita gunakan untuk segala keperluan. *Assets* dapat berupa sebuah material sederhana ataupun tekstur, file audio, atau sebuah prefab.

Seluruh *Assets* dalam sebuah *Project* akan ditampilkan dalam *Project Pane*. Sebuah *asset* yang dipakai dalam sebuah *scene*, akan tampak dalam *Hierarchy Pane*, yang menerangkan segala konten dari sebuah *scene*.

2.2.5. Prefabs

Prefabs merupakan kata lain dari istilah "*Prefabricated GameObjects*". Prefabs merupakan sebuah *Asset* yang dipakai menjadi sebuah *template*. Segala perubahan pengaturan sebuah Prefab yang terjadi dalam *Project*

Pane akan berpengaruh terhadap prefabs yang terdapat dalam adegan. Prefabs ideal digunakan untuk elemen-elemen yang sering dipakai dalam adegan, seperti: pintu, lemari, kursi, dan lain-lain.

2.3. Spesifikasi Sistem

Aplikasi realitas maya yang dinamakan “E-Landscape” ini dibuat ke dalam bentuk aplikasi tiga dimensi yang menggunakan sudut pandang orang pertama. Dalam sudut pandang orang pertama, tidak tampak aktor yang bergerak dan berada dalam ruang tiga dimensi, namun seolah-olah layar menunjukkan apa yang dilihat oleh aktor tersebut secara langsung. Penggunaan sudut pandang orang pertama ini bertujuan untuk memudahkan pemakai aplikasi ini untuk merasakan dengan nyata apa yang terdapat didalam aplikasi realitas maya ini

2.4. Diagram Perancangan

2.4.1. Diagram Fungsional

2.4.1.1. Diagram Use-Case

Diagram *use case* adalah sebuah diagram yang digunakan untuk menunjukkan tampilan grafis dari fungsionalitas yang diberikan oleh sistem dilihat dari sisi aktor, tujuan aktor, dan hal yang berkaitan dengan *use case* yang ada.

Aktor yang terdapat dalam sistem adalah pengguna. Pengguna disini adalah siapa saja yang menggunakan aplikasi. Aktor *user* biasa ini akan disebut sebagai pengguna dalam pembahasan ini, karena memang konteks aplikasi ditujukan kepada pengguna umum. Gambar 1 menunjukkan diagram *use case* untuk aktor pengguna.



Gambar 1. Diagram *use case* untuk aktor pengguna

Target pengguna aplikasi ini adalah calon mahasiswa Jurusan Teknik Elektro Universitas Diponegoro, dan masyarakat luas yang ingin mengetahui seperti apa tempat-tempat dan ruangan di gedung kampus Teknik Elektro ini. Aplikasi ini dirancang untuk dapat berjalan di dalam *web browser*. Tujuannya agar cakupan pengguna dari aplikasi ini semakin luas.

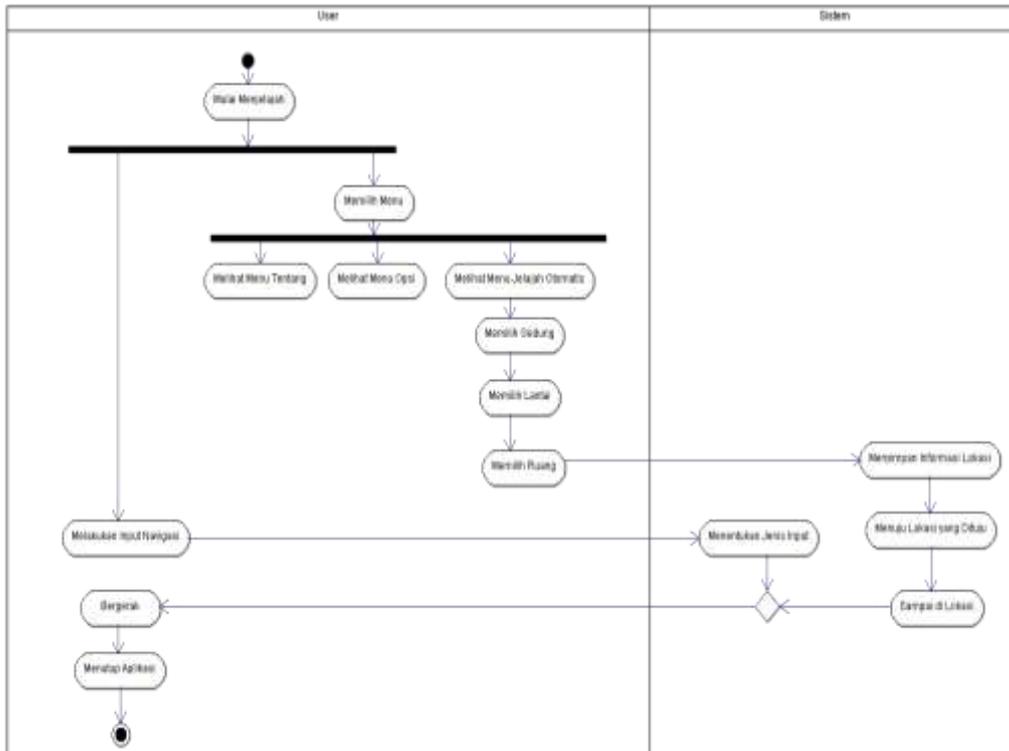
2.4.1.2. Diagram Aktivitas

Diagram aktivitas memodelkan tingkah laku dalam sebuah proses. Diagram ini juga merupakan salah satu bagian dari diagram fungsional, dan juga seperti halnya dengan diagram *use case*, diagram aktivitas juga digunakan pada fase analisis.

Dalam fase desain, diagram ini juga masih akan tetap digunakan. Diagram aktivitas dalam perangkat lunak

realitas maya ini dibagi menjadi bagian user dan bagian sistem. Gambar 2 merupakan diagram aktivitas dari aplikasi realitas maya.

dan juga obyek pendukung seperti tanaman, suara, dan cahaya.



Gambar 2. Diagram aktivitas proses kerja aplikasi realitas maya

2.4.2. Diagram Struktural

Diagram ini diwakilkan oleh diagram kelas yang merupakan sebuah tampilan statis struktur program. Diagram kelas menggambarkan struktur dengan memperlihatkan kelas dari sistem, baik atribut maupun relasi dengan kelas lain.

Kelas adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Kelas menggambarkan keadaan (atribut/property) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). Kelas diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam perangkat lunak yang sedang kita kembangkan. Diagram kelas menggambarkan struktur dan deskripsi kelas beserta hubungan kelas satu sama lain.

Obyek 3D merupakan komponen yang utama didalam aplikasi realitas maya ini. Obyek 3D menjadi penting karena berperan dalam memvisualkan bentuk nyata objek kedalam bentuk virtual. Obyek-obyek tersebut tidak hanya terbatas pada bentuk gedung Elektro, tetapi juga aktor, struktur lanskap apakah bergelombang atau datar,

Komponen lain yang tidak kalah pentingnya adalah antarmuka pengguna (GUI). Antarmuka pengguna menjadi penting karena memberikan tampilan fungsionalitas dari aplikasi realitas maya ini. Antarmuka membuat pengguna tahu fitur-fitur apa sajakah yang terdapat didalam aplikasi realitas maya ini.

Antarmuka yang terdapat di dalam sistem ini digabungkan menjadi sebuah menu utama. Dimana di dalam menu utama tersebut, seluruh menu dapat diakses. Di dalam GUI menu utama, terdapat tombol GUI Opsi, GUI jelajah otomatis, dan GUI tentang.

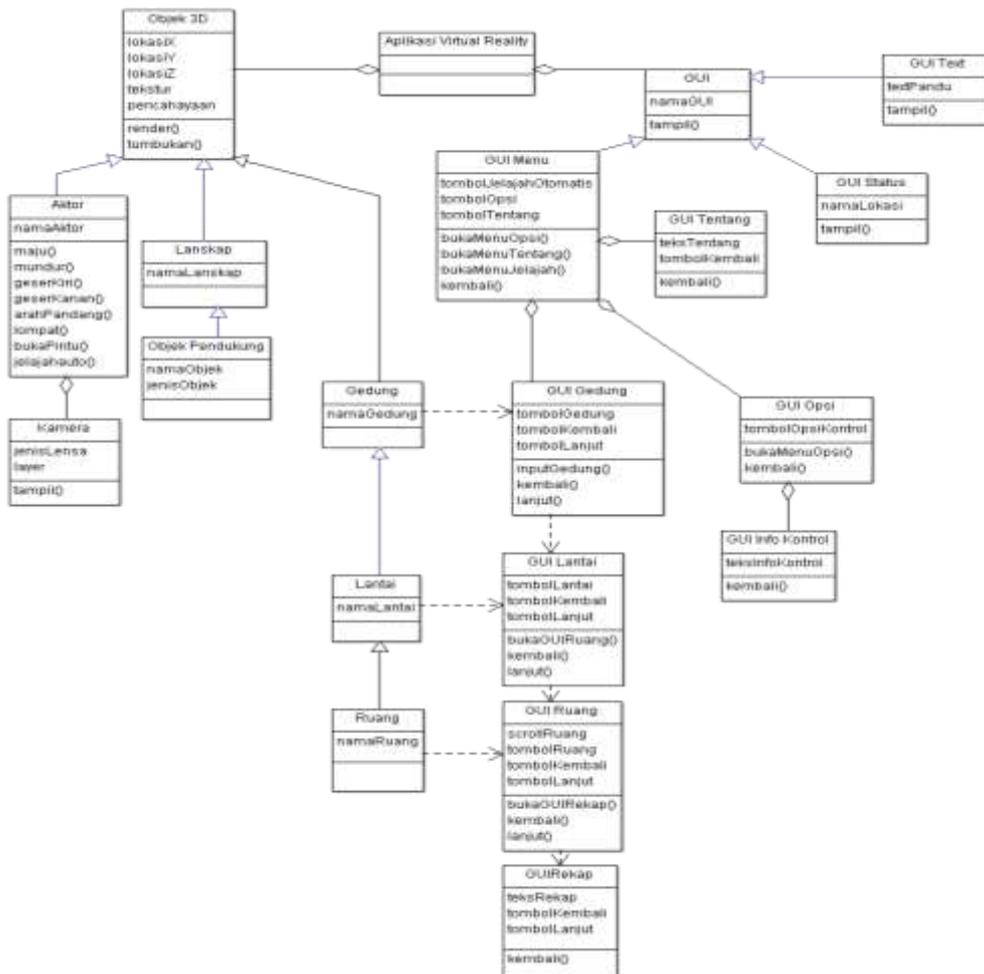
GUI jelajah otomatis terdiri dari kelas GUI Gedung, yang memberikan tampilan gedung apa sajakah yang dapat kita pilih. Kemudian selanjutnya kelas GUI lantai, yang menunjukkan lantai yang dapat kita pilih. Terakhir kelas GUI ruang, dimana kita dapat memilih ruang yang kita tuju.

Kelas GUI opsi memberikan tampilan menu Opsi yang terdiri dari Info Kontrol. Info Kontrol berada dalam kelas GUI Info Kontrol. Sedangkan kelas GUI tentang menginisialisasi obyek GUI tentang, yang berisi informasi seputar aplikasi realitas maya ini.

GUI yang juga merupakan komponen utama adalah GUI Status, yang berada di dalam kelas GUI Status. GUI status ini memberikan informasi berupa teks yang dapat dibaca di dalam kotak teks di dalam tampilan layar, berupa informasi tentang lokasi dimana pengguna berada di dalam kawasan Teknik Elektro Universitas Diponegoro ini. Di saat pengguna menjelajah kawasan Teknik Elektro Universitas Diponegoro, pengguna pasti membutuhkan informasi dimana dia berada saat ini. GUI status bertugas mengirimkan informasi posisi untuk pengguna. Gambar 3 menunjukkan diagram kelas dari aplikasi realitas maya.

Diagram urutan digunakan untuk menunjukkan interaksi antar pengguna dengan objek-objek lain yang terkait. Diagram ini lebih menitikberatkan pada urutan pesan yang dikirimkan antar pengguna dan obyek berdasarkan waktu. Biasanya diagram ini digunakan untuk menggambarkan skenario dalam suatu *use-case* yang ingin dijelaskan secara detail atau juga dapat digunakan untuk menggambarkan sebuah proses dalam *use-case*.

Diagram urutan menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna,



Gambar 3. Diagram kelas dari aplikasi realitas maya

2.4.2. Diagram Tingkah Laku

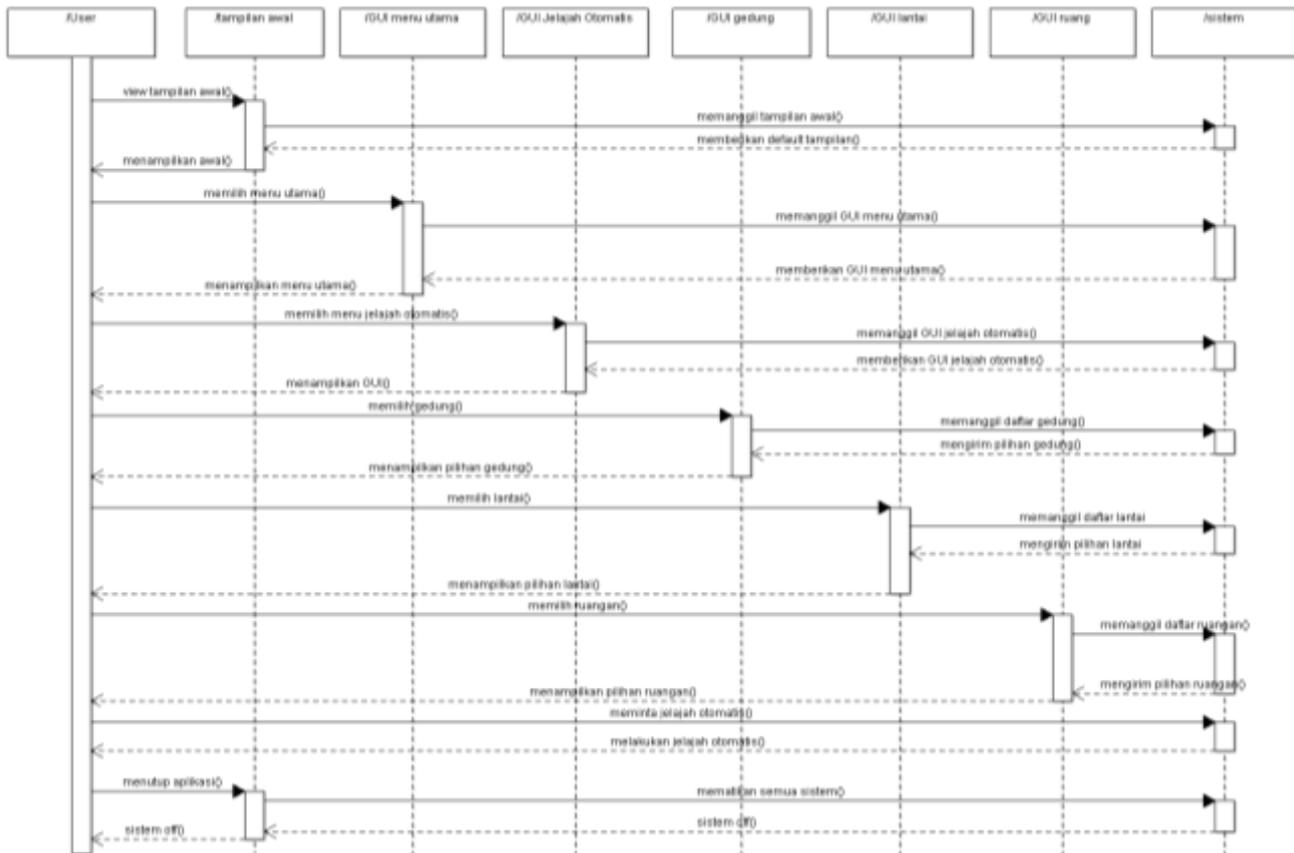
2.4.2.1. Diagram Urutan

Diagram urutan adalah sebuah diagram interaksi di dalam UML yang menunjukkan proses suatu operasi dengan lainnya dengan fokus ke arah urutan dari proses mana yang mendahului maupun yang tertinggal.

tampilan, dan sebagainya) berupa pesan yang digambarkan terhadap waktu. Diagram urutan terdiri antar dimensi vertikal (waktu) dan dimensi horisontal (objek-objek yang terkait). Diagram urutan biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu. Diawali dari apa yang memicu aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal, dan output apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. Pesan digambarkan sebagai

garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi/metoda dari class. Balok Aktivasi menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message. Gambar 4 menunjukkan diagram urutan aplikasi realitas maya untuk proses jelajah otomatis.

Selain dengan menggunakan tombol papan ketik, navigasi juga dilakukan dengan menggunakan masukan *mouse*. Masukan *mouse* digunakan untuk menentukan arah pandangan pada layar navigasi. Masukan *mouse* yang digunakan adalah kombinasi gerakan *mouse* ke atas dan ke bawah. Gerakan *mouse* keatas berarti gerakan pandangan ke atas, begitupun sebaliknya.



Gambar 4. Diagram urutan proses kerja jelajah otomatis

3. Hasil dan Analisa

3.1. Pengujian Unit

3.1.1. Pengujian Kemampuan Navigasi

Setelah program realitas maya ini berjalan, maka pengguna akan langsung dapat berinteraksi dengan dunia virtual dalam kawasan Teknik Elektro Universitas Diponegoro. Navigasi dilakukan dengan menekan tombol pada papan ketik. Tombol **W** sebagai representasi gerakan maju. Tombol **S** sebagai representasi gerakan mundur. Tombol **A** sebagai representasi gerakan ke kiri. Tombol **D** sebagai representasi gerakan ke kanan, dan masukan dari gerakan *mouse* sebagai representasi dari arah pandangan. Adapun pelengkap gerakan yaitu melompat dapat dilakukan dengan menekan tombol spasi pada papan ketik.

Semua proses navigasi arah pandangan dilakukan dengan senarai yang dapat dijabarkan sebagai berikut.

```
void Update ()
{
    if (axes == RotationAxes.MouseXAndY)
    {
        float rotationX =
        transform.localEulerAngles.y +
        Input.GetAxis("Mouse X") * sensitivityX;

        rotationY += Input.GetAxis("Mouse Y") *
        sensitivityY;
        rotationY = Mathf.Clamp (rotationY,
        minimumY, maximumY);

        transform.localEulerAngles = new Vector3(-
        rotationY, rotationX, 0);
    }
    else if (axes == RotationAxes.MouseX)
```

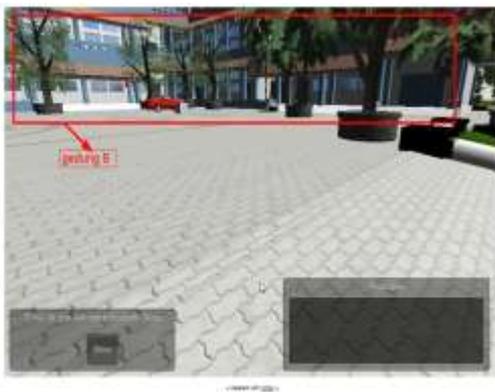
```

{
transform.Rotate(0, Input.GetAxis("Mouse X") * sensitivityX, 0);
}
else
{
rotationY += Input.GetAxis("Mouse Y") * sensitivityY;
rotationY = Mathf.Clamp (rotationY, minimumY, maximumY);
transform.localEulerAngles = new Vector3(-rotationY, transform.localEulerAngles.y, 0);
}
}
}
    
```

Gambar 5 dan 6 menampilkan pengujian kemampuan navigasi.



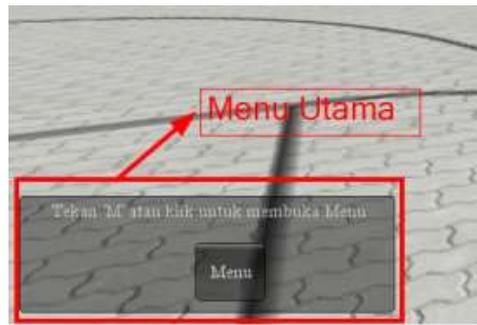
Gambar 5. Tampilan program setelah melakukan masukan navigasi maju



Gambar 6. Tampilan program setelah melakukan masukan mouse ke arah kanan bawah

3.1.2. Pengujian Tampilan Menu

Setelah dilakukan pengujian terhadap semua standar navigasi yang dibutuhkan, selanjutnya dilakukanlah pengujian terhadap tampilan menu. Gambar 7 menampilkan pengujian pada tampilan menu.



Gambar 7. Tampilan menu utama pada layar

Senarai MenuScript berikut menjelaskan tentang proses penampilan menu utama ke layar saat program berjalan.

```

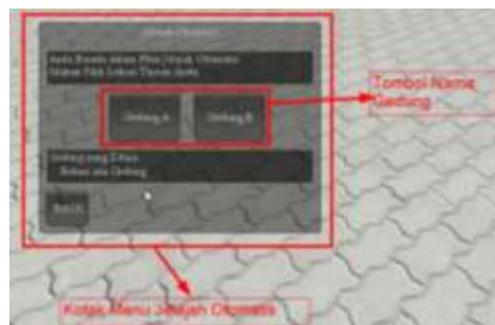
void OnGUI(){
GUI.skin = MenuSkin;
GUI.Box(new Rect (10, 490,320,100), "Tekan 'M' atau klik untuk membuka Menu");

if (GUI.Button (new Rect (160, 530, 60, 50), "Menu") || Input.GetKey(KeyCode.M))
{
Subyek.GetComponent<GUIMenu>().enabled = true;
Subyek.GetComponent<GUIScript>().enabled = false;
Subyek.GetComponent<MenuScript>().enabled = false;
}
}
    
```

3.1.3. Pengujian Jelajah Otomatis

Salah satu fitur yang terdapat dalam perangkat lunak realitas maya ini adalah fitur jelajah otomatis. Gambar 8 menampilkan menu **Jelajah Otomatis**.

Di dalam fitur jelajah otomatis, pengguna dapat menentukan lokasi mana di kawasan gedung Jurusan Teknik Elektro Universitas Diponegoro yang ingin dituju. Dalam fitur ini, pengguna diminta untuk melakukan masukan lokasi gedung, lalu lokasi lantai, dan yang terakhir nama ruangan yang dituju. Setelah pengguna menentukan lokasi yang dipilih, sistem akan mengarahkan pengguna menuju lokasi yang dipilih sebelumnya.



Gambar 8. Tampilan menu jelajah otomatis

Mode jelajah otomatis ini dapat dijelaskan dalam senarai sebagai berikut.

```
GetComponent<NavMeshAgent>().enabled = true;  
if (Subyek.GetComponent<GUINamaRuang>().kodePenje  
lajah=="B203")  
{  
GetComponent<NavMeshAgent>().destination=  
B203.position;  
}
```

Gambar 9 menampilkan tampilan mode Jelajah Automatis.



Gambar 9. Tampilan mode jelajah otomatis

4. Kesimpulan

Dari proses studi dapat disimpulkan bahwa sebuah perangkat lunak realitas maya kampus Teknik Elektro Universitas Diponegoro dapat diimplementasikan menggunakan mesin penerjemah grafis Unity3D berbasis bahasa pemrograman C#. Sistem yang dibangun merupakan aplikasi berbasis web yang dapat dijalankan dan diakses melalui *web browser* dengan prasyarat instalasi *Unity Web Player*. Perangkat lunak tersebut dapat dilengkapi dengan fitur-fitur tambahan sesuai dengan kebutuhan pengguna. Perangkat lunak realitas maya ini disarankan dapat diterapkan secara *online* pada jaringan Internet di kampus Jurusan Teknik Elektro Universitas Diponegoro sehingga bisa diakses dan dijalankan pengguna dari jarak jauh. Selanjutnya, perangkat lunak ini disarankan bisa dikembangkan ke dalam aplikasi realitas maya dengan tampilan yang lebih baik, dengan kawasan yang lebih luas untuk diakses dan dukungan objek-objek 3D yang semakin lengkap.

Referensi

- [1] Chonoles, Michael Jesse and James A. Schardt. *UML 2 for Dummies*. U.S.A., Hungry Minds Publishing: 2003
- [2] Dariush, Derakhshani and Randi Munn. *Introducing 3ds Max 9: 3D for Beginners*. Wiley Publishing, 2007.
- [3] Davis, Harold. *Visual C# .NET Programming*. Richard Mills, 2002.
- [4] Ericson, Christer. *Real-Time Collision Detection*. Morgan Kaufman Publisher, 2004.
- [5] Harris, Andy. *Microsoft C# Programming for the Absolute Beginner*. Premiere Press, 2002

- [6] Luke, Ahearn. *3D Game Textures: Create Professional Game Art Using Photoshop*. U.S.A: Focal Press, 2006
- [7] Rasheed, Faraz. *C# School: 14 Lessons to Get You Started With C# and .NET*. Synchron Data, 2005.
- [8] van der Bergen, Gino. *Collision Detection in Interactive 3D Environment*. Morgan Kaufman Publisher, 2004
- [9] _____. *Belajar Pemrograman C#*. Penerbit Andi, 2008.
- [10] _____. *3D Platformer Tutorial: Building a 3D Platform Game in Unity 2.0*. Unity Team Documentation.