



Aplikasi Pencarian Ruangan Menggunakan *Augmented Reality* di Gedung Cc Universitas Muhammadiyah Jember

Della Hartika Putri¹, Daryanto^{2*}, Agung Nilogiri³

Program Studi Teknik Informatika, Universitas Muhammadiyah Jember^{1,2,3}

Email: dellahartikaputri@gmail.com¹, daryanto@unmuhjember.ac.id^{2*}, agungnilogiri@unmuhjember.ac.id³

ABSTRAK

Gedung Cc Universitas Muhammadiyah Jember mempunyai banyak ruangan yang berada dalam daerah yang cukup luas. Selain itu, rute jalan dari gedung Cc menyulitkan seseorang untuk mencapai suatu lokasi atau ruangan, dan *Google Maps* tidak bekerja dengan baik untuk navigasi dalam ruangan. *Augmented Reality* merupakan sebuah konsep pelapisan konten visual di atas visualisasi kondisi sebenarnya yang terlihat melalui sebuah kamera. *Augmented reality* memanipulasi perangkat *mobile* dengan menampilkan lapisan informasi tambahan secara visual diatas lapisan tampilan layar dalam kondisi sebenarnya dari perangkat kamera sehingga dapat berinteraksi. Dari masalah tersebut, timbul ide untuk membangun aplikasi dengan metode *augmented reality* menggunakan telepon genggam untuk pencarian lokasi atau ruangan di Gedung Cc Universitas Muhammadiyah Jember. Pengujian fungsionalitas system pada aplikasi yang dibuat ini dengan metode *blackbox*. Dalam pengujian akurasi digunakan untuk mengetahui apakah metode A* dapat menemukan jalur terpendek melalui *Augmented Reality*. Sehingga menghasilkan sebuah penelitian bahwa Algoritma A* berhasil diterapkan di aplikasi pencarian jalur terpendek menggunakan sistem operasi IOS atau perangkat Iphone. Dari 3 parameter pengukuran yaitu jarak real dan jarak aplikasi yang diujikan menggunakan kasus 1 dan kasus 2 didapatkan masing-masing hasil 100% dan 90% sedangkan untuk ketepatan penunjuk arah sebesar 100%.

Kata Kunci: Ruangan, Gedung, *Mobile*, *Augmented Reality*, *Black Box*, Algoritma A*

ABSTRACT

The Cc Building of the Muhammadiyah University of Jember has many rooms located in a fairly large area. In addition, the walking route from the Cc building makes it difficult for a person to reach a location or room, and *Google Maps* does not work well for indoor navigation. *Augmented Reality* is a concept of layering visual content on top of the visualization of actual conditions seen through a camera. *Augmented reality* manipulates mobile devices by displaying an additional layer of information visually on top of the screen display layer in the actual state of the camera device so that it can interact. From this problem, the idea arose to build an application with the augmented reality method using a mobile phone to search for a location or room in the Cc Building of the Muhammadiyah University of Jember. Testing the system functionality in this application using the black box method. In the accuracy test, it is used to determine whether the A* method can find the shortest path through *Augmented Reality*. This resulted in a study that the A* Algorithm was successfully applied in the shortest path search application using the IOS operating system or Iphone device. Of the 3 measurement parameters, namely the real distance and the application distance, which were tested using case 1 and case 2, the results were 100% and 90% respectively, while the accuracy of the directions was 100%.

Keywords: Room, Building, *Mobile*, *Augmented Reality*, *Black Box*, A* Algorithm

1. PENDAHULUAN

Google Maps merupakan sebuah aplikasi yang memiliki fungsi sebagai penyedia layanan peta dinamis yang digunakan untuk menampilkan informasi lokasi dan rute jalan berdasarkan data yang tersimpan (Mahdia & Noviyanto, 2013). Akan tetapi, *Google Maps* tidak berfungsi maksimal ketika berada di dalam ruangan.

Gedung Cc Universitas Muhammadiyah Jember mempunyai banyak ruangan yang berada dalam daerah yang cukup luas. Selain itu, rute jalan dari gedung Cc menyulitkan seseorang untuk mencapai suatu lokasi atau ruangan, dan *Google Maps* tidak bekerja dengan baik untuk navigasi dalam ruangan.

Augmented Reality (AR) merupakan sebuah konsep pelapisan konten visual di atas visualisasi kondisi sebenarnya yang terlihat melalui sebuah kamera. *Augmented reality* memanipulasi perangkat *mobile* dengan menampilkan lapisan informasi tambahan secara visual diatas lapisan tampilan layar dalam kondisi sebenarnya dari perangkat kamera sehingga dapat berinteraksi.

Dari masalah tersebut, timbul ide untuk membangun aplikasi *augmented reality* dengan menggunakan telepon genggam untuk pencarian lokasi atau ruangan di Gedung Cc Universitas

Muhammadiyah Jember. Penggunaan AR merupakan alternatif yang inovatif untuk dijadikan pengalaman dan pembelajaran berbasis lokasi kepada pengguna. Secara tradisional, orientasi lingkungan diberikan melalui rambu-rambu, pemandu wisata, atau melalui *Web*. Sebagai tujuan perjalanan, penambahan informasi terhadap suatu lingkungan sangat diperlukan. Berdasarkan permasalahan tersebut, maka rumusan pada penelitian ini adalah bagaimana aplikasi *augmented reality* penunjuk ruangan di Gedung Cc Universitas Muhammadiyah Jember diterapkan pada penggunaan sehari-hari dan berapa nilai akurasi A* yang diterapkan pada pengujian pada aplikasi.

2. KAJIAN PUSTAKA

A. *Augmented Reality* (AR)

Augmented Reality (AR) merupakan teknologi yang mengkolaborasikan benda maya 2 dimensi maupun 3 dimensi ke dalam sebuah lingkungan nyata 3 dimensi kemudian memproyeksikan benda-benda maya tersebut dalam waktu nyata (*real time*). (Maramis dkk., 2016).

Augmented Reality memiliki dua metode yang sedang dikembangkan dan dapat digunakan oleh pengembang aplikasi, meliputi:

1. *Marker Based Tracking* : Merupakan metode *augmented reality* dimana menggunakan benda dua dimensi yang disebut *marker* sebagai tempat untuk memproyeksi objek-objek virtual kedalam lingkungan sebenarnya.
2. *Markerless Tracking* : Merupakan metode *augmented reality* dimana pengguna tidak memerlukan lagi sebuah *marker* sebagai tempat memproyeksikan benda-benda virtual dua dimensi.

B. *Unity*

Unity dibangun tahun 2004 oleh David Helgason, Nicholas Francis, dan Joachim Ante. *Unity* merupakan *Game Engine* yang dibangun berdasarkan kepedulian terhadap *indie developer* yang tidak dapat membeli *game engine* karena terlalu mahal. Fokus perusahaan ini adalah membuat sebuah perangkat lunak yang bisa digunakan oleh semua orang, khususnya untuk membangun sebuah *game*. Di tahun 2009, *Unity* diluncurkan secara gratis dan pada bulan April 2012, *Unity* mencapai popularitas tertinggi dengan lebih dari sejuta *developer* terdaftar di seluruh dunia (Roedavan, 2018).

C. *Vuforia*

Vuforia merupakan alat pengembangan *software* atau biasa disebut *Software Development Kit* (SDK) yang memungkinkan pengembang melakukan pengembangan atau pembangunan aplikasi berbasis *Augmented Reality*. *Vuforia* memanfaatkan *Computer Vision* atau biasa disebut Kamera untuk mengenali dan melacak gambar target (*marker*) atau objek lain untuk menampilkan objek 3D sederhana, secara *realtime* pada perangkat *mobile*. Kemampuan untuk mengidentifikasi citra memungkinkan pengembang untuk memosisikan dan mengorientasi objek virtual dari *marker* sehingga perspektif penggunaan pada objek sesuai, seakan objek virtual menjadi bagian dari dunia nyata (Jati, 2017).

D. *Placenote*

Placenote merupakan *Software Development Kit* (SDK) yang memungkinkan membuat aplikasi AR berbasis *cloud* dengan cepat yang secara permanen menyimpan konten *Augmented Reality* pada suatu lokasi di dalam dan di luar ruangan. *Placenote* tidak memerlukan GPS.

Placenote terintegrasi dengan komponen *Augmented Reality* pada perangkat iOS dengan membawa fungsi pelacakan ARKit berbasis *cloud* dengan bahasa mesin yang mempermudah membuat aplikasi *Augmented Reality*.

E. Algoritma A*

Algoritma A* atau disebut juga Algoritma A-*Star*, adalah algoritma yang dikembangkan oleh Hart, Nilson, dan Raphael pada tahun 1968. Algoritma A* merupakan salah satu algoritma paling populer hingga saat ini, karena sangat optimal untuk melakukan pencarian jalur dan penentuan jalur

terpendek. Agar dapat lebih memahami prinsip kerja algoritma A*, penjelasan tahapan secara ringkas algoritma A* dalam *pseudocode* ditunjukkan pada gambar 1.

```
1) Menambahkan starting point ke dalam openlist.
2) Lakukan proses looping tahapan berikut:
  a. Hitung cost dalam openlist dengan rumus:  $f(n)=g(n)+h(n)$ ,
     dimana:
        $f(n)$  = cost untuk node n
        $g(n)$  = cost mencapai node n dari akar
        $h(n)$  = cost mencapai node tujuan dari node n
  b. Setelah mendapatkan nilai  $f(n)$  terendah kemudian jadikan node
     tersebut menjadi currentnode.
  c. Masukkan currentnode ke closedlist.
  d. Untuk tiap neighbor tetangga yang berdekatan dengan current
     node:
     - Jika tidak walkable atau jika termasuk closedset, maka
       abaikan node.
     - Jika walkable dan belum ada pada openlist, maka buat current
       node parent dari node tetangga ini, dan simpan nilai  $f$ ,  $g$  dan
        $h$  dari node ini.
     - Jika sudah ada pada openlist, periksa apakah ini jalur dari
       node ini ke current node yang lebih baik dengan menggunakan
       cost  $G$  sebagai tolak ukurannya. Jika simpul dengan cost  $G$ 
       lebih rendah menandakan bahwa ini adalah jalan yang lebih
       baik, lalu buatlah neighbor node ini sebagai camefrom dari
       current node, dan menghitung ulang nilai  $G$  dan  $F$  dari simpul
       ini.
  e. Berhenti perulangan ketika node tujuan telah ditambahkan ke
     dalam openlist, dalam hal ini, jalur telah ditemukan. Atau
     hentikan perulangan jika openlist telah kosong. Dalam kasus ini,
     tidak ada jalan yang di temukan/gagal.
3) Simpan rute
4) Telusuri rute secara backward, mulai dari goal node ke parent sampai
   mencapai node awal.
```

Gambar 1. *Pseudocode* algoritma A*

3. METODE PENELITIAN

A. Kerangka Penelitian

1) Studi Literatur

Tahap pertama yang dilakukan adalah studi literatur yaitu mempelajari artikel dan hasil penelitian terdahulu. Studi literatur yang dilakukan dengan membaca buku, jurnal, dan *paper* yang terkait dengan masalah yang akan dibahas.

2) Menyusun Proposal Penelitian

Sebelum menyusun proposal penelitian terlebih dahulu melakukan identifikasi masalah, kemudian merumuskannya dalam bentuk kalimat tanya, setelah rumusan masalah dibuat langkah selanjutnya yaitu menentukan pemecahan dari masalah yang akan diselesaikan. Dalam penelitian ini, mendapatkan hasil akhir berupa pembangunan aplikasi navigasi *indoor* dengan *augmented reality* di Gedung Cc Universitas Muhammadiyah Jember.

3) Analisis Kebutuhan

Pada tahap ini akan diidentifikasi beberapa kebutuhan dari aplikasi yang akan dirancang, mulai dari berupa *software*, *hardware*, dan data kebutuhan fungsional dari perangkat lunak.

4) Pembangunan

Proses pembangunan dilakukan merupakan sebuah aplikasi *augmented reality* berbasis *mobile*, yang akan dibangun menggunakan *tools IDE Unity 3D*, dengan *SDK Placernote*, dan bahasa *C#*.

5) Implementasi dan Pengujian

Setelah proses pembangunan aplikasi selesai, aplikasi akan melalui tahap implementasi dan pengujian yang merupakan tahap akhir, dan aplikasi akan diinstal pada *smartphone* iOS, dan akan dilakukan implementasi untuk mengukur kelayakan aplikasi.

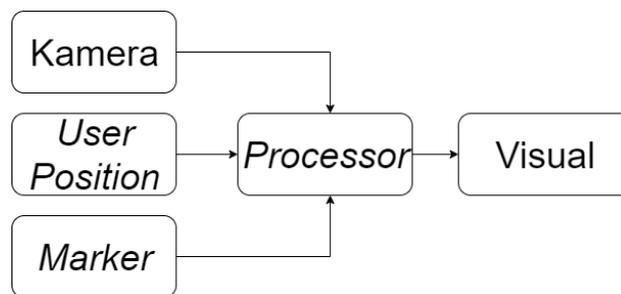
B. Gambaran Umum Sistem

Sistem yang akan dibangun berupa sistem navigasi indoor untuk Gedung Cc Universitas Muhammadiyah Jember yang akan dibangun dengan *platform* iOS yang akan memiliki sistem *augmented reality*, yang nantinya aplikasi ini memiliki tujuan untuk memberi informasi peta ruangan pada Gedung Cc agar memudahkan pengguna dalam pencarian sebuah lokasi di Gedung Cc, penggunayang menjadi target adalah mahasiswa maupun warga luar yang mengunjungi Gedung Cc Universitas Muhammadiyah Jember.

Aplikasi ini diperuntukkan membantu pengguna dalam pencarian letak ruangan ataupun kursi dosen dengan cara melakukan pemindaian terhadap titik awal lokasi *user*, kemudian *user* diharapkan untuk mengikuti konten visual yang merupakan petunjuk pada sistem tersebut.

C. Teknik Visualisasi

Pada sebuah layar kamera untuk menampilkan objek, dilakukan pembacaan *marker* oleh kamera. *Marker* yang saling berkaitan dalam aplikasi akan menampilkan objek dan informasi pada layar *smartphone* iOS. Teknik visualisasi menggunakan teknik *MarkerBase AR*, seperti pada gambar 2.



Gambar 2. Teknik Visualisasi

1) Kamera

Pada *augmented reality*, kamera berfungsi sebagai alat pembaca *marker*. Lingkungan dan objek yang terekam dikamera akan ditampilkan pada layar *smartphone* iOS.

2) *User Position*

User Position merupakan sistem yang akan membaca dimana posisi *user* berada, dan sebagai acuan peneliti untuk meletakkan sebuah *marker*.

3) *Marker*

Marker pada sistem ini merupakan penanda sebagai acuan penempatan elemen virtual yang akan mempermudah *user* dalam pencarian sebuah lokasi.

4) *Processor*

Processor adalah sebagai pengolah dan pemroses data yang dapat menghasilkan suatu informasi yang diperlukan.

5) Tampilan Visual

Tampilan visual aplikasi ini merupakan sebuah output yang berupa tampilan objek nyata dan objek virtual.

D. Kebutuhan Sistem

- 1) *Hardware*
 - a. Untuk dapat menginstal IDE *Unity*, diperlukan laptop dengan minimal RAM 4 GB.
 - b. Untuk dapat menjalankan aplikasi, diperlukan *smartphone* iOS dengan minimal iOS 11.
 - c. Dalam menjalankan aplikasi guna menguji dan menemukan masalah dalam aplikasi menggunakan SDK, diperlukan kabel USB sebagai media pendukung USB *debugging*.
- 2) *Software*
 - a. *Unity 3D*, merupakan sebuah IDE yang akan digunakan dalam membangun aplikasi.
 - b. *Placenote*, sebagai *Software Development Kit* untuk membantu pengembang membuat aplikasi yang dapat berinteraksi dengan dunia nyata.

4. HASIL DAN PEMBAHASAN

A. Implementasi Sistem

Pada poin ini, menjelaskan tentang hasil implementasi sistem yang telah dibangun. Hasil implementasi sistem yang dimaksud meliputi implementasi *hardware*, implementasi *software*, implementasi *user interface* dan implementasi algoritma.

- 1) Implementasi algoritma merupakan potongan kode class hasil dari implementasi algoritma A* yang berguna untuk menentukan jalur terpendek.

```

using UnityEngine;
using System;
using System.Collections.Generic;

public class AStar : MonoBehaviour
{
    public List<Node> FindPath(Node startNode, Node targetNode, Node[] allNodes)
    {
        List<Node> openList = new List<Node>();
        openList.Add(startNode);
        List<Node> closedList = new List<Node>();
        while (openList.Count > 0)
        {
            Node currentNode = openList[0];
            for (int i = 1; i < openList.Count; i++)
            {
                if (openList[i].GCost < currentNode.GCost)
                {
                    openList[i].GCost = currentNode.GCost;
                    openList[i].HCost = currentNode.HCost;
                    openList[i].FCost = currentNode.FCost;
                }
            }
            openList.Remove(currentNode);
            closedList.Add(currentNode);
            if (currentNode == targetNode)
            {
                return new List<Node> { startNode, targetNode };
            }
            foreach (Node connection in currentNode.connections)
            {
                if (!closedList.Contains(connection))
                {
                    float costToConnection = currentNode.GCost + Mathf.Abs(currentNode.x - connection.x) + Mathf.Abs(currentNode.z - connection.z);
                    if (costToConnection < connection.GCost || !openList.Contains(connection))
                    {
                        connection.GCost = costToConnection;
                        connection.HCost = Mathf.Abs(connection.x - targetNode.x) + Mathf.Abs(connection.z - targetNode.z);
                        connection.FCost = connection.GCost + connection.HCost;
                        openList.Add(connection);
                    }
                }
            }
        }
        return null;
    }
}
    
```

Gambar 3. Potongan kode program implementasi algoritma A*

- 2) Implementasi *hardware* merupakan spesifikasi perangkat komputer dan smartphone yang digunakan untuk implementasi system.

Tabel 1. Spesifikasi *hardware*

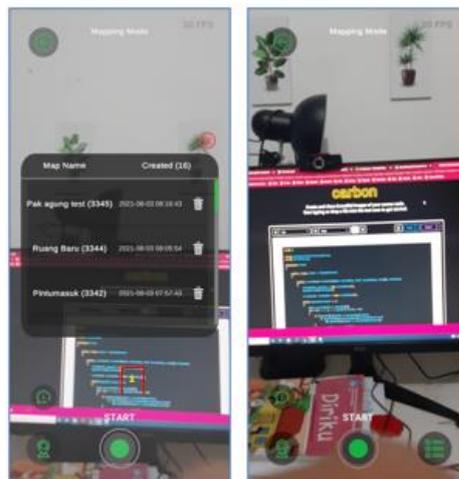
No	Perangkat Keras	Spesifikasi
1	MacBook Air 2020	Memori 8GB, SSD 256GB, Intel Core i5 quadcore 1,1 Ghz, Mac OS BigSur, Wi-Fi
2	Iphone 11	RAM 4GB, ROM 64GB, IOS 14, CPU Apple A11 Bionic Hexa Core 2.39 Ghz, GPU Apple GPU, Layar 6.1 Inch Super Amoled, Dual Camera Depan z MP, 12+12 MP Belakang

- 3) Implementasi *software* merupakan kebutuhan *software* pendukung yang digunakan untuk mengimplementasikan sistem.

Tabel 2. *Software* pendukung

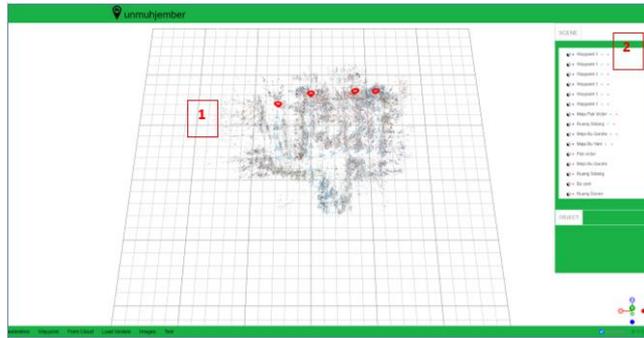
No	Perangkat Keras	Spesifikasi
1	Unity 2019.4.3fl	<i>Integrated Development Environment</i> , sebuah <i>editor</i> yang digunakan untuk membangun aplikasi berbasis grafis
2	Azure Spatial Anchor	Layanan pengembang lintas <i>platform</i> yang memungkinkan untuk membuat pengalaman realitas campuran
3	XCode 12	<i>Integrated Development Environment</i> , sebuah <i>editor</i> yang digunakan untuk membangun aplikasi berbasis IOS, IOSX, WatchOS, dan TVOS
4	ARKit	SDK IOS yang memberikan kapabilitas <i>augmented reality</i> tanpa memerlukan <i>hardware</i> tambahan
5	ARwayKit	Layanan SDK untuk pengembangan AR <i>Interactive</i> lintas <i>platform</i>

- 4) Implementasi *user interface* merupakan hasil dari implementasi sistem pada bagian User Interface. Adapun *user interface* yang telah dibangun terdiri dari 3 aplikasi yaitu untuk *user interface* mapping, web studio dan location. Masing-masing aplikasi mempunyai fungsi tersendiri yaitu *user interface* mapping digunakan untuk melakukan perekaman lokasi atau pemetaan. Web studio digunakan sebagai penentuan jalur dari hasil pemetaan. Dan yang terakhir yaitu location digunakan sebagai penunjuk mencari lokasi terdekat. Antarmuka untuk aplikasi *Mapping* yang digunakan sebagai perekam atau pemetaan mempunyai implementasi sistem ditunjukkan pada gambar 4.



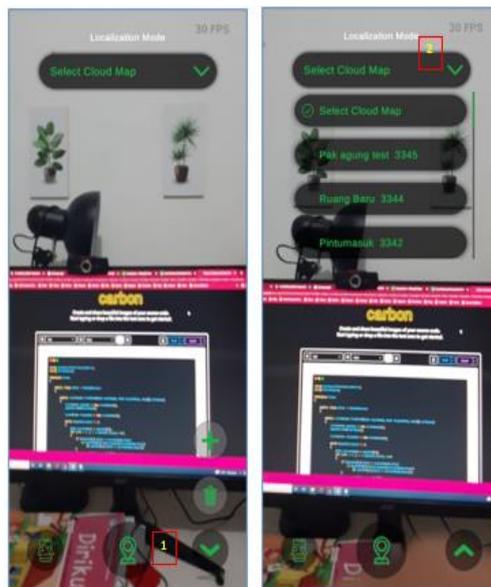
Gambar 4. Tampilan User Interface untuk Aplikasi Mapping

Pada gambar 5 menunjukkan tampilan antarmuka web studio, dimana antarmuka ini digunakan sebagai pembuatan jalur atau peta. Disini juga segala informasi terkait titik dan jalur semua disimpan. Lembar kerja yang digunakan untuk meletakkan hasil mapping dan pembuatan jalur. Scene digunakan sebagai kontrol untuk pengaturan tiap objek yang tersedia di lembar kerja.



Gambar 5. Web Studio untuk menentukan Path

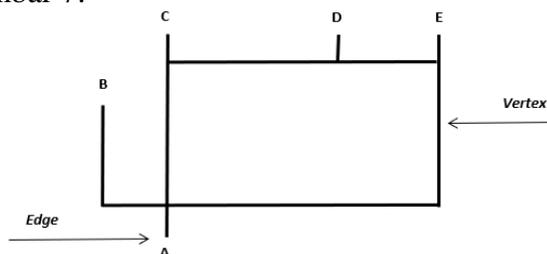
Pada antarmuka *localization* digunakan sebagai penunjuk arah ketika ingin menuju ke sebuah lokasi. Berikut antarmuka *localization*. Antarmuka ini digunakan untuk mengaktifkan pencarian lokasi dan digunakan sebagai daftar *map* yang akan dituju.



Gambar 6. User Interface Localization

B. Pengujian Sistem

Dalam penelitian ini untuk mendapatkan hasil yang diharapkan maka berdasarkan analisis masalah dan desain pada metode penelitian maka terdapat beberapa pengujian yaitu pengujian fungsionalitas sistem dimana pengujian menggunakan mekanisme *blackbox*. Sedangkan untuk pengujian akurasi berfungsi untuk mendapatkan apakah metode A* dapat menentukan jalur terpendek. Studi kasus yang peneliti gunakan adalah Ruang Dosen dimana terdapat 5 *vertex* dan 4 *edge* berikut sketsanya ditunjukkan pada gambar 7.



Gambar 7. User Interface Localization

Keterangan Gambar 7:

- A adalah pintu masuk ruang dosen
- B adalah Ruang Rapat / Sidang
- C adalah Meja Bu Qurrota Ayun
- D adalah Meja Bu Yeni
- E adalah Meja Pak Victor

Tahap pengujian menggunakan metode *black box testing* yang digunakan untuk mengetahui apakah fitur aplikasi bekerja sesuai yang diharapkan. Pengujian *black box* digunakan untuk menguji sebuah *software* tanpa harus memperhatikan detail *software* (Priyaungga dkk., 2020). Hasil uji coba ditampilkan pada Tabel 3, 4, dan 5.

Tabel 3. Hasil uji coba fitur mapping

Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tombol Start	Dapat melakukan perekaman sesuai lokasi area	Tombol aktif dan berfungsi untuk proses merekam	Berfungsi
Tombol Stop	Dapat melakukan pemberhentian perekaman	Tombol aktif dan berfungsi untuk memberhentikan perekaman	Berfungsi
Tombol List Map	Dapat menampilkan daftar hasil rekaman yang telah disimpan	Tombol berhasil menampilkan daftar hasil rekaman yang telah disimpan	Berfungsi
Inisialisasi Camera	Dapat aktif dan memunculkan camera	Camera berhasil aktif dan dapat digunakan	Berfungsi
Spatial Anchor	Dapat mendeteksi objek melalui anchor disaat merekam area	Anchor dapat mendeteksi objek di sekitar area	Berfungsi
Dialog Name Map	Dapat memberi nama map dan menyimpan hasil rekaman	Pemberian nama dan penyimpanan hasil rekaman berhasil disimpan	Berfungsi
Uploading	Dapat melakukan proses uploading atau penyimpanan ke server cloud	Pross uploading berhasil tersimpan di cloud	Berfungsi

Tabel 4. Hasil uji coba fitur localization

Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Button localize	Dapat menscaning dan mendeteksi titik mulai pencarian lokasi	Scanning dan pencarian lokasi berhasil ditemukan dan berfungsi	Berfungsi
Indikator Arah	Dapat menampilkan indicator arah	Indikator arah dapat dimunculkan dalam bentuk AR	Berfungsi
Indikator Jarak	Dapat menampilkan indicator jarak	Indikator jarak berhasil muncul di aplikasi	Berfungsi
Indikator Simbol Penunjuk	Dapat memunculkan pemandu arah dalam bentuk simbol penunjuk	Indikator simbol penunjuk arah berhasil dan berfungsi	Berfungsi
Daftar list cloud Map	Dapat menampilkan daftar hasil rekaman map	Point cloud berhasil menampilkan daftar data rekaman map	Berfungsi

Dalam pengujian akurasi digunakan untuk mengetahui apakah metode A* dapat menemukan jalur terpendek melalui *Augmented Reality*. Skenario pengujian akurasi adalah tahapan alur cerita yang akan dilakukan untuk melakukan pengujian. Adapun *parameter* yang digunakan untuk mengukur

akurasi adalah jarak sebenarnya, jarak real dan ketepatan penunjuk arah yang direkomendasikan. Terdapat beberapa skenario ditunjukkan pada tabel 5.

Tabel 5. Skenario pengujian akurasi

Skenario	Keterangan
Kasus 1 tanpa pembanding	Pengujian yang dilakukan dimana pencarian vertex hanya melalui 1 edge
Kasus 2 dengan pembanding	Pengujian yang dilakukan dimana pencarian vertex melalui 2 edge

Hasil uji coba akurasi, berdasarkan skenario pengujian yang telah dilakukan yaitu uji coba pada kasus 1 dengan tanpa pembanding dan kasus 2 dengan pembanding, didapatkan hasil yang ditunjukkan pada tabel 6.

Tabel 6. Hasil pengujian kasus 1 tanpa pembanding

No	Pengujian	Jarak aplikasi (meter)	Jarak real (meter)	Selisih jarak (meter)	Persentase kesesuaian (%)
1	Dari A ke B (Ruang Rapat)	5	3	4	60
2	Dari A ke C (Meja Bu Qurrota)	9	5	1	55.60
3	Dari A ke E (Meja Pak Victor)	12	11	1	91.67
4	Dari C ke D (meja Bu Yeni)	5	4	1	80
5	Dari C ke E (Meja Pak Victor)	6	5	1	83.34
6	Dari D ke C (Meja Bu Qurrota)	5	4	1	80
7	Dari D ke E (Meja Pak Victor)	2	1	1	50
8	Dari E ke D (Meja Bu Yeni)	2	1	1	50
9	Dari E ke C (Meja Bu Qurrota)	6	5	1	83.34

Tabel 7. Hasil pengujian kasus 1 dengan pembanding

No	Pengujian	Jarak aplikasi (meter)	Jarak real (meter)	Selisih jarak (meter)	Persentase kesesuaian (%)
1	Dari A ke D (Meja Bu Yeni)	13	9	4	69.2
2	Dari A ke E (Meja Pak Victor)	12	10	2	83.34
3	Dari B ke C (Meja Bu Qurrota)	10	8	2	80
4	Dari B ke D (Meja Bu Yeni)	12	12	-	100
5	Dari B ke E (Meja Pak Victor)	15	13	2	86.67
6	Dari C ke B (Ruang Rapat)	10	8	2	80
7	Dari C ke D (Meja Bu Yeni)	5	4	1	80
8	Dari D ke B (Ruang Rapat)	12	12	-	100
9	Dari D ke C (Meja Bu Qurrota)	5	4	1	80
10	Dari D ke E (Meja Pak Victor)	1	1	1	100
11	Dari E ke B (Ruang Rapat)	15	13	2	86.67
12	Dari E ke C (Meja Bu Qurrota)	5	5	-	100

Dari semua pengujian akurasi didapatkan hasil skenario kasus 1 akurasinya adalah 100% sedangkan untuk kasus 2 akurasinya mencapai 90% antara jarak aplikasi dan jarak *real* dimana ketepatan penunjuk arah sesuai sesuai jarak terpendek dengan nilai akurasi adalah 100%. Ketepatan jarak aplikasi dan jarak *real* dipengaruhi pada saat proses *mapping*. Oleh sebab itu, ketepatan yang dihasilkan masih belum dapat mencapai ke angka 100%.

5. KESIMPULAN

Berdasarkan hasil analisis dan pembahasan yang sudah dilakukan, dapat disimpulkan bahwa algoritma A* berhasil diterapkan di aplikasi pencarian jalur terpendek menggunakan sistem operasi

IOS atau perangkat *Iphone*. Dari 3 parameter pengukuran yaitu jarak *real* dan jarak aplikasi yang diujikan menggunakan kasus 1 dan kasus 2 didapatkan masing-masing hasil 100% dan 90% sedangkan untuk ketepatan penunjuk arah sebesar 100%.

Saran pada penelitian selanjutnya diharapkan untuk mengkaji lebih banyak studi kasus terkait penerapan di area outdoor karena penelitian ini masih dalam lingkup indoor yaitu ruang dosen. Untuk parameter uji kecepatan proses algoritma A* dapat ditambahkan sebagai bahan pengkajian selanjutnya karena keterbatasan dalam penelitian ini hanya menggunakan 3 parameter yaitu akurasi jarak real, jarak rekomendasi aplikasi dan ketepatan penunjuk arah.

6. DAFTAR PUSTAKA

- Jati, M, I. (2017). *Membuat Aplikasi Augmented Reality bagian 1 (Membuat License Key, Database dan Upload Marker)*. Retrieved August 6, 2019, from <https://www.codepolitan.com/membuat-aplikasi-augmented-reality-bagian-1-membuat-license-key-database-dan-upload-marker-5852d4228d473-21874>
- Mahdia, F., & Noviyanto, F., (2013). Pemanfaatan Google Maps API Untuk Pembangunan Sistem Informasi Manajemen Bantuan Logistik Pasca Bencana Alam Berbasis Mobile Web. *Jurnal Sarjana Teknik Informatika*. 1(1): 162-171.
- Maramis, M. I., Lumenta A.S., & Sugiarto B. A., (2016). Augmented Reality Pada Aplikasi Untuk Memperlihatkan Gedung Fatck. *E-burnal Teknik Elektro dan Komputer*. 5(1): 40-48.
- Priyaangga, Bayu Aji, Dwi Bayu Aji, Mukron Syahroni, Nurul Tri Sukma Aji, & Aries Saifudin. (2020). Pengujian Black Box Pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi* 3(3):150.
- Roedavan, R. (2018). *Unity Tutorial Game Engine*. Bandung: Informatika.