**PROPOSAL TUGAS AKHIR**

**PEMANFAATAN *AUGMENTED REALITY* UNTUK PENCARIAN DAN *PROFILING* PESANTREN SE-JAWA BARAT MENGGUNAKAN ALGORITMA *HAVERSINE FORMULA* DAN *DIJKSTRA***

**Oleh:**

**Nikko Eka Saputra**

**1157050115**

****

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SUNAN GUNUNG DJATI**

**BANDUNG**

**2019**

# **LEMBAR PERSETUJUAN PROPOSAL**

**Nama : Nikko Eka Saputra**

**NIM : 1157050115**

**Jurusan : Teknik Informatika**

**Judul Tugas Akhir : Pemanfaatan *Augmented Reality* untuk Pencarian dan *Profiling* Pesantren se-Jawa Barat menggunakan Algoritma *Haversine Formula* dan *Dijkstra***

***(Utilization of Augmented Reality for Search and Profile of Islamic Boarding Schools in West Java’s using Haversine Formula and Dijkstra Algorithms)***

**Proposal ini telah diperiksa dan disetujui**

**pada tanggal ........................**

|  |  |  |
| --- | --- | --- |
| **Ketua Jurusan,** |  | **Calon Pembimbing,** |
| **Mohamad Irfan, ST., M.Kom** |  | **Mohamad Irfan, ST., M.Kom.** |
| **NIP. 198310232009121005** |  | **NIP. 198310232009121005** |

# **LEMBAR PENGESAHAN PROPOSAL**

**Nama : Nikko Eka Saputra**

**NIM : 1157050115**

**Jurusan : Teknik Informatika**

**Judul Tugas Akhir : Pemanfaatan *Augmented Reality* untuk Pencarian dan *Profiling* Pesantren se-Jawa Barat menggunakan Algoritma *Haversine Formula* dan *Dijkstra***

***(Utilization of Augmented Reality for Search and Profile of Islamic Boarding Schools in West Java’s using Haversine Formula and Dijkstra Algorithms)***

**Proposal ini telah diseminarkan dan disahkan**

**pada tanggal .............................**

|  |  |  |
| --- | --- | --- |
| **Ketua Jurusan,** |  | **Penelaah,** |
| **Mohamad Irfan, ST., M.Kom.** |  | **Ichsan Taufik, ST., MT.** |
| **NIP. 198310232009121005** |  | **NIP. 198009132006041002** |

# **PROPOSAL TUGAS AKHIR**

**PEMANFAATAN *AUGMENTED REALITY* UNTUK PENCARIAN DAN *PROFILING* PESANTREN SE-JAWA BARAT MENGGUNAKAN ALGORITMA *HAVERSINE FORMULA* DAN *DIJKSTRA***

* 1. **Latar Belakang**

Manusia adalah makhluk berakal yang ciptakan oleh Allah untuk senantiasa mencari ilmu pengetahuan. Salah satu cara untuk meraih ilmu pengetahuan tersebut adalah melalui pendidikan. Dalam hal pendidikan, kebutuhan manusia tidak cukup hanya dengan pendidikan umum saja tetapi diperlukan juga pendidikan agama untuk membentuk karakter dan kepribadian manusia menjadi lebih baik. Salah satu tempat untuk mempelajari dan memperdalam pendidikan keagamaan adalah di Pondok Pesantren. Dhofier mendefinisikan pesantren sebagai lembaga pendidikan tradisional Islam untuk mempelajari, memahami, menghayati dan mengamalkan ajaran Islam dengan menekankan pentingnya moral keagamaan dalam perilaku sehari-hari [1].

Dalam menjalankan aktivitasnya, pesantren memiliki peranan yang sangat penting dalam perkembangan suatu daerah khususnya pada nilai sosial dan keislaman. Karena di zaman sekarang ini tidak dapat dipungkiri sudah mulai terjadi yang disebut dengan krisis akhlak, pondok pesantren dapat dijadikan sebuah benteng agar tidak terbawa arus tersebut. Pesantren juga dapat membentuk generasi muda yang tangguh dalam bersaing dengan tantangan zaman yang tentunya tetap berpedoman pada nilai-nilai keislaman. Oleh karena itu, banyak orang tua lebih memilih untuk menempatkan anak-anak mereka ke pondok pesantren dengan harapan dapat membentuk kepribadian, akhlak, dan moral keagamaan yang baik.

Berdasarkan data yang bersumber dari Kementrian Agama Provinsi Jawa Barat pada tahun 2019 menyatakan bahwa ada 7.910 pondok pesantren yang terletak di Provinsi Jawa Barat. Sehingga berdasarkan jumlah pesantren yang banyak dan kebutuhan masyarakat terhadap pesantren, secara tidak langsung hal itu membuat masyarakat menginginkan kemudahan untuk mendapatkan informasi tentang pesantren pula. Maka dalam hal ini perlu didukung oleh teknologi yang berkembang saat ini untuk membantu baik bagi pihak pesantren dalam memberikan informasi mengenai pesantren maupun bagi masyarakat dalam mencari informasi tentang pesantren. Dengan melihat urgensi dari peranan pondok pesantren di Jawa Barat menjadi motivasi untuk membuat dan merancang sebuah sistem untuk memetakan lokasi pesantren yang tidak semua masyarakat mengetahuinya dan juga memberikan informasi mengenai pondok pesantren tersebut. Media yang digunakan yaitu *Augmented Reality* yang merupakan teknologi yang dapat menggabungkan benda digital ke dalam dunia nyata. Dalam hal ini, *Augmented Reality* berguna untuk menampilkan marker lokasi pesantren di dalam kamera dengan menggunakan metode *Haversine Formula* untuk menghitung jarak dan *Dijkstra* untuk mencari rute terdekat antara pengguna dengan lokasi pesantren tersebut.

Berdasarkan penjelasan tersebut, maka topik tugas akhir yang akan diangkat adalah tentang implementasi *Augmented Reality* untuk pemetaan lokasi pesantren se-Jawa Barat dengan judul **“Pemanfaatan *Augmented Reality* untuk Pencarian dan *Profiling* Pesantren se-Jawa Barat menggunakan Algoritma *Haversine Formula* dan *Dijkstra*”**.

* 1. **Perumusan Masalah**

Berdasarkan dari latar belakang di atas maka dapat dirumuskan menjadi beberapa masalah diantaranya:

1. Bagaimana merancang dan mengimplementasikan teknologi *Augmented Reality* dengan algoritma *Haversine Formula* dan *Dijkstra* untuk pencarian dan *profiling* pesantren se-Jawa Barat?
2. Bagaimana kinerja *Haversine Formula* dan *Dijkstra* untuk mencari rute terpendek dari satu titik ke titik yang lain?
   1. **Tujuan dan Manfaat**

Berdasarkan rumusan masalah tersebut, maka tujuan perancangan aplikasi ini adalah sebagai berikut :

1. Merancang dan mengimplementasikan teknologi *Augmented Reality* dengan algoritma *Haversine Formula* dan *Dijkstra* untuk pencarian dan *profiling* pesantren se-Jawa Barat.
2. Mengetahui kinerja *Haversine Formula* dan *Dijkstra* untuk mencari rute terpendek dari satu titik ke titik yang lain.
   1. **Batasan Masalah**

Adapun batasan masalah dari penelitian ini adalah:

1. Teknologi yang digunakan adalah *Augmented Reality.*
2. Objek yang dapat dideteksi adalah lokasi pondok pesantren se-Jawa Barat.
3. Sumber data pondok pesantren bersumber dari Kementrian Agama Provinsi Jawa Barat.
4. Profil pesantren memiliki atribut nama pesantren, pimpinan, nomor telepon, ormas, kurikulum, fasilitas, alamat lengkap, dan foto pesantren.
5. Teknik yang digunakan adalah *Location Based Services (LBS)*.
6. Informasi yang ditampilkan berupa marker 2 dimensi yang berisi gambaran singkat dari pesantren dan jarak tempuhnya.
7. Aplikasi bisa menampilkan navigasi menuju ke lokasi pesantren.
8. Perhitungan jarak menggunakan algoritma *Haversine Formula*.
9. Penentuan rute terdekat menuju lokasi pesantren menggunakan algoritma *Dijkstra.*
10. Produk yang dihasilkan adalah aplikasi Android untuk pengguna, web admin untuk pengelolaan data pesantren, dan web service berupa JSON.
    1. ***State of The Art***

Banyak penelitian yang telah dilakukan sebelumnya untuk melakukan pencarian sebuah lokasi atau tempat dengan menggunakan Sistem Informasi Geografis dan Augmented Reality. Dalam upaya untuk mengembangkan sistem yang serupa dengan hal tersebut maka dirasa perlu dilakukan studi literatur sebagai salah satu tahap pada metode penelitian yang akan dilakukan. Berikut merupakan penelitian yang dilakukan sebelumnya dan memiliki kecocokan dengan topik yang peneliti bahas, antara lain:

1. Penelitian yang dilakukan oleh Fachri Febriana pada tahun 2014 dengan judul Aplikasi Panduan Tur Kampus dengan Teknologi Augmented Reality Berbasis Android (Studi Kasus Universitas Islam Negeri Sunan Gunung Djati Bandung) bertujuan untuk mengetahui lokasi fasilitas atau gedung dan navigasi di kampus UIN Sunan Gunung Djati Bandung menggunakan teknik *Markerless Augmented Reality* dengan library *Mixare AR* [2].
2. Penelitian yang dilakukan oleh Mohammad Ilyas pada tahun 2015 dengan judul Aplikasi Pencarian Lokasi dan Jarak Dinas-Dinas di Kabupaten Subang Dengan Augmented Reality Menggunakan Metode Location Based Services Berbasis Android bertujuan untuk pencarian lokasi dan jarak dinas-dinas yang ada di kabupaten Subang menggunakan *Augmented Reality* dengan library *AR Browser* dan juga *web service* untuk mengambil data lokasi berformat *JSON* [3].
3. Penelitian yang dilakukan oleh Sokthay Chanphearith dkk pada tahun 2016 dengan judul Analisis dan Implementasi Berbasis Lokasi Aplikasi Mobile Augmented Reality untuk Pencarian Tempat Wisata dan Tempat-tempat Kuliner di Kota Phnom Penh, Kamboja bertujuan untuk mencari tempat wisata dan kuliner di kota Phnom Penh menggunakan *Augmented Reality* dengan menggunakan *web service* untuk mendapatkan data lokasinya dan *GPS* untuk mendapatkan lokasi pengguna saat ini [4].
4. Penelitian yang dilakukan oleh Alim Hardiansyah pada tahun 2017 dengan judul Merancang Augmented Reality Berbasis Aplikasi Android dengan Location Based Services bertujuan untuk menampilkan lokasi SMA di kota Tangerang menggunakan *Augmented Reality* dengan metode *Location Based Service* dan *Global Positioning System* [5].
5. Penelitian yang dilakukan oleh Sigin Khoerul Anwar pada tahun 2017 dengan judul Implementasi Augmented Reality Menggunakan Location Based Service Dan Haversine Formula Untuk Penunjang Fasilitas Umum Di Kabupaten Pangandaran bertujuan untuk menunjang fasilitas pariwisata di kabupaten Pangandaran menggunakan *Augmented Reality* dengan metode *Location Based Service* dan *Harvesine Formula* [6].
6. Penelitian yang dilakukan oleh Indra Permana dkk pada tahun 2015 dengan judul Location Based Service Sebagai Penunjuk Lokasi Hotel Di Kota Semarang Berbasis Augmented Reality bertujuan untuk menampilkan lokasi hotel di kota Semarang menggunakan *channel* dari Junaio yang mana informasi hotelnya didapatkan dari *database server* dengan format XML [7].
7. Penelitian yang dilakukan oleh Yohanes Priyo Atmojo dkk pada tahun 2014 dengan judul Pemanfaatan Augmented Reality Pada Sistem Informasi Geografis Kampus Di Bali bertujuan untuk memberikan informasi lokasi PTS yang ada di Bali menggunakan *Augmented Reality* dan mengambil data melalui *web service* berformat *JSON* [8].
8. Penelitian yang dilakukan oleh Galih Laksono dkk pada tahun 2016 dengan judul Pemanfaatan Teknologi Augmented Reality Markerless Sebagai Media Pengenalan Gedung Universitas Kanjuruhan Malang Berbasis Android bertujuan untuk menampilkan gedung-gedung yang ada di Universitas Kanjuruhan Malang dengan menggunakan metode *Markerless Augmented Reality* dalam menampilkan objek-objek digital [9].
9. Penelitian yang dilakukan oleh Andriyan Budiyanto dkk pada tahun 2016 dengan judul Perancangan Aplikasi Islamic Boarding School Finder berbasis Android menggunakan Algoritma Dijkstra yang bertujuan untuk pencarian Islamic Boarding School dengan menggunakan algoritma dijkstra untuk mencari rute terdekatnya [10].
10. Penelitian yang dilakukan oleh Albes Fajri dkk pada tahun 2018 dengan judul Implementasi Sistem Informasi Geografis untuk Pencarian Indekos dengan Jarak Terpendek menggunakan Metode Algoritma Dijkstra yang bertujuan untuk indekos di kelurahan Kandang Limun dan Beringin Raya [11].

Berikut adalah Tabel 1.5.1 yang menjelaskan secara singkat tentang *state of the art* yang digunakan pada penelitian ini.

**Tabel 1.5.1 State of the Art**

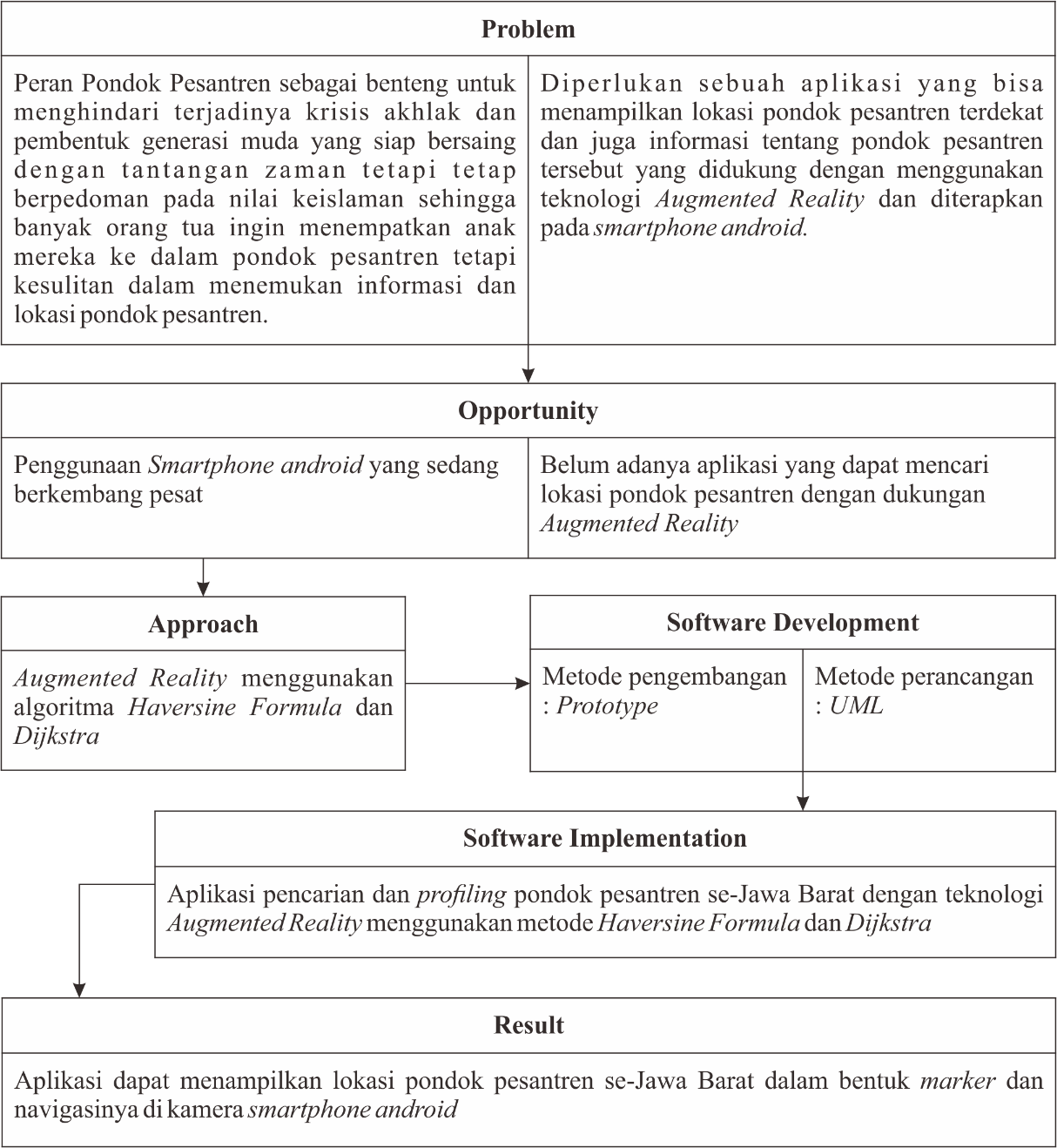
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Peneliti** | **Judul Penelitian** | **Kelebihan dan Kekurangan** |
| 1 | Fachri Febriana  (2014) | Aplikasi Panduan Tur Kampus dengan Teknologi Augmented Reality Berbasis Android (Studi Kasus Universitas Islam Negeri Sunan Gunung Djati Bandung) [13] | * Menggunakan metode *Markerless Augmented Reality* * Menggunakan library Mixare AR * Tampilan *marker* kurang menarik |
| 2 | Mohammad Ilyas  (2015) | Aplikasi Pencarian Lokasi dan Jarak Dinas-Dinas di Kabupaten Subang Dengan Augmented Reality Menggunakan Metode Location Based Services Berbasis Android [14] | * Menggunakan metode *Location Based Service* * Tampilan *marker* sudah menarik * Ada fitur map/navigasi pada kamera *Augmented Reality* * Ada fitur *offline maps* |
| 3 | Sokthay Chanphearith, dkk  (2016) | Analisis dan Implementasi Berbasis Lokasi Aplikasi Mobile Augmented Reality untuk Pencarian Tempat Wisata dan Tempat-tempat Kuliner di Kota Phnom Penh, Kamboja [15] | * Tampilan aplikasi keseluruhan dan *marker* sudah menarik * Ada fitur map/navigasi pada kamera *Augmented Reality* * Bisa mengatur *range radius* * Ada fitur navigasi dari pengguna dengan lokasi |
| 4 | Alim Hardiansyah  (2017) | Merancang Augmented Reality Berbasis Aplikasi Android dengan Location Based Services [16] | * Menggunakan metode *Location Based Service* dengan GPS * Koneksi internet harus stabil ketika menjalankan aplikasi * Tampilan aplikasi secara keseluruhan kurang menarik |

**Tabel 1.5.1 State of the Art (Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Peneliti** | **Judul Penelitian** | **Kelebihan dan Kekurangan** |
| 5 | Sigit Khoerul Anwar  (2017) | Implementasi Augmented Reality Menggunakan Location Based Service Dan Haversine Formula Untuk Penunjang Fasilitas Umum Di Kabupaten Pangandaran [17] | * Menggunakan *Harvesine Formula* untuk menghitung jarak antara pengguna dengan lokasi * Tampilan *marker* kurang menarik |
| 6 | Indra Permana, dkk  (2015) | Location Based Service Sebagai Penunjuk Lokasi Hotel Di Kota Semarang Berbasis Augmented Reality [18] | * Menggunakan metode *Location Based Service* * Menggunakan *channel* dari aplikasi Junaio * Ada fitur jarak antara pengguna ke lokasi hotel * Aplikasi belum terpublikasi |
| 7 | Yohanes Priyo Atmojo, dkk  (2014) | Pemanfaatan Augmented Reality Pada Sistem Informasi Geografis Kampus Di Bali [19] | * Data lokasi kampus diambil dari webservice berformat JSON * Tampilan *marker* kurang menarik * Belum ada fitur navigasi ke lokasi kampusnya |
| 8 | Galih Laksono, dkk  (2016) | Pemanfaatan Teknologi Augmented Reality Markerless Sebagai Media Pengenalan Gedung Universitas Kanjuruhan Malang Berbasis Android [20] | * Mudah diaplikasikan oleh pengguna * Aplikasi masih belum mampu menampilkan lokasi gedung secara akurat |

* 1. **Kerangka Pemikiran**

Adapun kerangka pemikiran dari penelitian tugas akhir ini direpresentasikan pada Gambar 1.6.1.



**Gambar 1.6.1 Kerangka Pemikiran**

* 1. **Studi Pustaka**

### **Pondok Pesantren**

Definisi pesantren menurut asal katanya yaitu berasal dari kata “santri” kemudian mendapatkan awalan “pe” dan akhiran “an” yang mana menunjukan tempat, maka pesantren artinya adalah tempat untuk para santri. Terkadang pesantren juga dianggap sebagai kombinasi dari kata “santri” yang berarti manusia baik dengan suku kata “tra” yang berarti suka menolong sehingga pesantren dapat diartikan sebagai tempat pendidikan manusia yang baik-baik [1]. Pesantren juga dikenal dengan tambahan istilah pondok yang dalam arti kata bahasa Indonesia memiliki arti gubug, kamar, rumah kecil yang terbuat dari bambu dengan menekankan kesederhanaan dari bangunannya.

Pada hakikatnya dari berbagai definisi pesantren yang dikemukakan, maka dapat dipahami bahwa pondok pesantren adalah sebuah lembaga pendidikan Islam tradisional yang di dalamnya mempelajari agama dengan menekankan pembentukan moral dan akhlak santri yang dibimbing oleh kyai, kitab kuning sebagai sumber utama pembelajaran dan masjid sebagai pusat kegiatan santri.

Jawa Barat merupakan provinsi yang mempunyai beberapa wilayah didalamnya. Penduduk jawa barat mayoritasnya beragama islam, dengan mayoritas agama islam terdapat pula pondok pesantren untuk memperdalamnya. Keberadaan pesantren di Jawa Barat memiliki peran penting dalam perjalanan sejarah bangsa Indonesia. Oleh karena itu, dibutuhkan adanya keseimbangan keilmuan yang diberikan oleh para kyai untuk mencerdaskan santri/santriwati pada pondok pesantren yang berada di jawa Barat.

Pondok pesantren di Jawa Barat terbagi menjadi beberapa wilayah diantaranya dapat dilihat pada tabel 2.2 sebagai berikut:

Tabel 2.2 Daftar Wilayah Pondok Pesantren di Jawa Barat

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kabupaten / Kota** | **Jumlah Pondok Pesantren** |
| 1 | Kota Bandung | 104 |
| 2 | Kab Bandung | 616 |
| 3 | Kab Sumedang | 50 |
| 4 | Kab Garut | 948 |
| 5 | Kab Tasikmalaya | 149 |

Tabel 2.2 Daftar Wilayah Pondok Pesantren di Jawa Barat (Lanjutan)

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Kabupaten / Kota** | **Jumlah Pondok Pesantren** |
| 6 | Kab Ciamis | 491 |
| 7 | Kab Bekasi | 201 |
| 8 | Kab Karawang | 456 |
| 9 | Kab Purwakarta | 190 |
| 10 | Kab Subang | 377 |
| 11 | Kab Bogor | 1277 |
| 12 | Kab Cirebon | 584 |
| 13 | Kota Cirebon | 46 |
| 14 | Kab Indramayu | 198 |
| 15 | Kab Kuningan | 175 |
| 16 | Kab Majalengka | 200 |
| 17 | Kota Bekasi | 45 |
| 18 | Kota Depok | 39 |
| 19 | Kota Tasikmalaya | 229 |
| 20 | Kota Cimahi | 39 |
| 21 | Kota Banjar | 46 |
| 22 | Kab Bandung Barat | 488 |
| 23 | Kab Pangandaran | 195 |
| **TOTAL** | | **7.910** |

### ***Haversine Formula***

Haversine Formula merupakan metode untuk mengetahui jarak antar dua titik dengan memperhitungkan bahwa bumi bukanlah sebuah bidang datar namun adalah sebuah bidang yang memilki derajat kelengkungan. Metode Haversine Formula menghitung jarak antara 2 titik dengan berdasarkan panjang garis lurus antara 2 titik pada garis bujur dan lintang [12]. Dalam hal ini adalah titik GPS pengguna dengan titik GPS tujuan yang masing-masing berisi koordinat longitude dan latitude sebagai data utama dalam perbandingan jarak sehingga dapat ditentukannya lokasi terdekat.

Berikut adalah bentuk persamaan dari algoritma *Haversine Formula*:

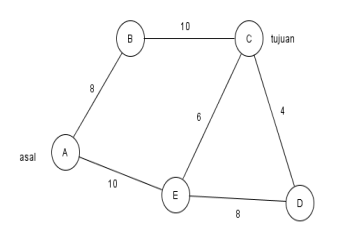
;

*R = Radius Bumi = 6371 km (1 derajat = 0.0174532925 radian);*

### **Algoritma *Dijkstra***

Algoritma Dijkstra adalah algoritma untuk mencari jarak terpendek dari sejumlah langkah. Algoritma ini ditemukan oleh Edsger Wybe Dijkstra. Dengan menggunakan prinsin Greedy yang menyatakan bahwa pada setiap langkah kita memilih sisi yang berbobot minimum dan memasukkannya ke dalam himpunan solusi [13].

Dijsktra dapat menentukan jarak terpendek dari suatu vertex ke vertex yang lainnya pada suatu graph yang berbobot, dimana jarak antara vertex adalah bobot dari tiap edge pada graph tersebut. Cara kerjanya adalah dengan mencari jarak terpendek dari vertex asal ke vertex terdekatnya, kemudian ke vertex yang kedua, dan seterusnya. Graph berarah digambarkan pada gambar 2.1 berikut.



Gambar 2.1 Graph Berbobot

Misalnya diberikan graph berbobot dan berarah seperti gambar di atas. Akan dicari lintasan terpendek dari simpul A ke simpul C. Maka untuk menyelesaikan permasalahan tersebut dapat diselesaikan sebagai berikut.

A ke B dengan jarak = 8

A ke E dengan jarak = 10

B ke C dengan jarak = 10

E ke C dengan jarak = 6

E ke D dengan jarak = 8

D ke C dengan jarak = 4

Kemudian untuk menentukan jarak terdekat dari semua kemungkinan yang ada yaitu :

* 1. A ke B dengan jarak = 8

A ke E dengan jarak = 10

Maka dari data diatas dipilih lah A ke B = 8

* 1. A ke B ke C dengan jarak = 18

A ke E dengan jarak = 10

Maka dipilih A ke E dengan jarak = 10, karena lebih kecil / dekat.

### ***Augmented Reality***

*Augmented Reality* merupakan gabungan dari objek dunia nyata dan maya yang ditampikan pada lingkungan nyata, berjalan secara *realtime* dalam waktu yang nyata dan dapat terintegrasi antar benda 3 dimensi. Penggabungan benda nyata dan maya dapat memungkinkan dengan teknologi yang sesuai, interaktivitas dapat memungkinkan melalui perangkat-perangkat input tertentu dan integrasi yang baik memerlukan *marker* yang efektif [14].

Tujuan dari *augmented reality* adalah untuk menampilkan objek dunia maya ke dalam dunia nyata. Pada saat ini, penggunaan dari *augmented reality* sudah sangat populer digunakan pada beberapa bidang. Diantaranya bidang kedokteran untuk pencitraan medis, bidang penerbangan untuk membantu pilot menunjukkan dataran, dan di museum untuk menampilkan artifak sejarah. *Augmented Reality* juga telah diaplikasikan dalam perangkat yang sudah banyak digunakan seperti *smartphone.*

Adapun karakteristik secara lebih spesifik dari *Augmented Reality* dapat didefinisikan sebagai berikut:

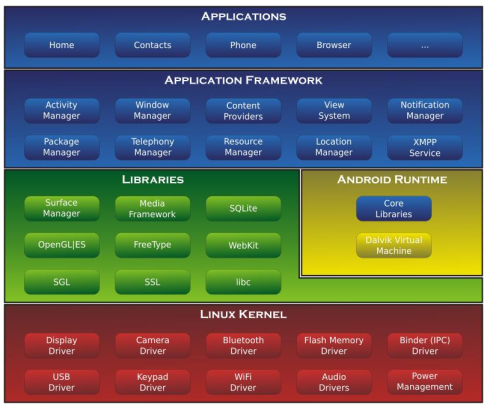
1. Menggabungkan dunia nyata dengan grafis komputer.
2. Mempunyai interaksi dengan objek secara langsung dalam waktu tertentu.
3. Melakukan pendeteksian objek secara *realtime.*
4. Menyediakan pengenalan objek.
5. Menyediakan data *realtime*

### **Android**

Android adalah sebuah sistem operasi untuk mobile yang berbasis Linux. Android menyediakan platform open source bagi para developer untuk membuat aplikasi mereka sendiri. Pada awalnya, perusahaan Google membeli perusahaan Android sebagai pendatang baru yang membuat perangkat lunak untuk mobile. Kemudian Android pada perkembangannya dibentuklah Open Handset Alliance, konkorsium dari 34 perusahaan perangkat keras, perangkat lunak dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcom, T-Mobile dan Nvidia [15].

Terdapat beberepa versi pada sitem operasi pada Android, mulai dari Android versi 1.1, Android versi 1.5 (Cupcake), Android versi 1.6 (Donut), Android versi 2.0/2.1 (Eclair), Android versi 2.2 (Froyo: Frozen Yoghurt), Android versi 2.3 (Gingerbread), Android versi 3.0 (Honeycomb) dan Android versi 4.0 (ice Cream Sanwich).

Pada dasarnya arsitektur Android tersurun atas beberapa layer yang saling berkaitan antara satu dengan yang lainnya. Berikut adalah Gambar 2.2 yang menampilkan gambaran dari arsitektur yang digunakan pada sistem operasi Android.



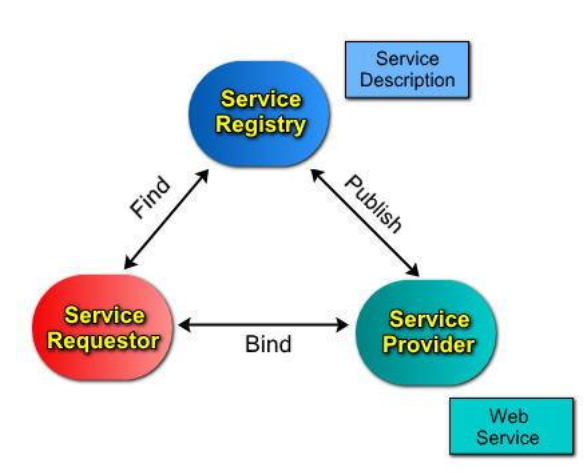
Gambar 2.2 Arsitektur Sistem Android

Saat ini terdapat dua jenis distributon dari sistem operasi Android. Pertama yaitu dari Google yang mana sistem ini mendapatkan dukungan penuh dari layanan yang disediakan oleh Google. Kedua yang distribusinya tanpa dukungan langsung dari Google atau dikenal dengan sebutan Open Handset Distribution (OHD).

### ***Web Service***

*Web service* adalah suatu perangkat lunak yang dibuat untuk mendukung integrasi dan interaksi antar sistem pada suatu jaringan. Biasanya digunakan sebagai suatu fasilitas yang disediakan oleh sebuah website untuk menjadi sebuah jembatan antara *database* dengan sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan yang disediakan. Implementasi data pada *webservice* adalah berupa format *JSON* atau *XML* sehingga data bisa diakses oleh sistem lain yang berbeda *platform,* sistem operasi, maupun bahasa pemrograman [16].

Secara umum, arsitektur dari *web service* dapat dilihat pada Gambar 2.3 berikut.



Gambar 2.3 Arsitektur *web service*

Pada gambar di atas, ada 3 komponen berjalannya suatu *web service.* Berikut adalah penjelasannya:

1. *Service Provider,* berfungsi untuk menyediakan operasi pada *web service.*
2. *Service Requestor,* bertindak sebagai *client* yang berinteraksi terharap layanan yang disediakan.
3. *Service Registry,* berfungsi sebagai tempat *service provider* mempublikasikan layannya.

### ***Location Based Service***

*Location Based Service* adalah sebuah layanan yang didasarkan pada posisi pengguna berada saat ini. Terkadang, pengguna tidak mengetahui posisi di mana dia berada. Oleh karena itu, sistem akan bekerja untuk menemukan posisinya secara *realtime*. Selanjutnya setelah posisi diketahui, maka data tersebut akan digunakan untuk memenuhi kebutuhan pengguna dan memungkinkan juga untuk mengakses aplikasi yang memang membutuhkan posisi pengguna sebagai syarat melalui GPS.

LBS mensyaratkan adanya interaksi berupa komunikasi dua arah. Misalnya seperti pengguna harus berinteraksi dengan *service provider* untuk mendapatkan data posisi, arah dan jenis informasi yang dibutuhkan. Setelah melakukan request maka akan dikirimkan informasi sesuai dengan yang diminta oleh pengguna [5].

### ***Global Positioning System***

GPS *(Global Positioning System)* adalah sebuah sistem navigasi radio dan penentuan posisi pengguna dengan menggunakan satelit. GPS bernama asli NAVSTAR GPS yaitu *Navigation Satellite Timing and Ranging Global Positioning System.* Sistem GPS memungkinkan dapat digunakan dalam segala kondisi, karena didesain sedemikian rupa untuk memberikan informasi posisi dan informasi mengenai waktu secara *realtime* di seluruh dunia.

Dibandingkan dengan metode untuk menentukan posisi lainnya, GPS mempunyai banyak kelebihan dan keuntungan, baik dalam segi operasional maupun kualitas posisi yang diberikan. Berikut ini adalah beberapa kelebihan penggunaan GPS [17]:

1. GPS memungkinkan digunakan pada kondisi cuaca dan waktu kapanpun.
2. GPS mempunyai ketinggian yang cukup tinggi, sekitar 20.000 km di atas permukaan bumi dan jumlahnya relatif lebih banyak.
3. GPS tidak terpengaruh dari topografis daerah.
4. GPS memungkinkan memberikan posisi dengan ketelitian sprektumnya cukup luas.
5. GPS tidak dikenakan biaya oleh pemilik satelit.

### **Java**

1. Sejarah Singkat Java

James Gosling, Patrick Naughton, Chris Warth, Ed Frank, dan Mike Sheridan adalah para pelopor Java dari Sun Microsystems, Inc di tahun 1991. Mereka membutuhkan kurang lebih 1.5 tahun untuk membuat versi pertamanya. Java pada awalnya disebut “Oak” namun kemudian berubah menjadi “Java” pada tahun 1995 dikarenakan Oak sudah digunakan oleh pihak lain. Pada proses pengembangannya, banyak orang yang terlibat baik dalam rancangan dan evolusi bahasa ini. Bill Joy, Arthur van Hoff, Jonathan Payne, Frank Yellin, dan Tim Lindholm merupakan orang-orang yang menjadi kunci keberhasilan java [18].

1. Keunggulan Java

Java merupakan bahasa yang unik dan inovatif dengan motto “write once run anywhere” sehingga program Java dapat dijalankan di berbagai platform sistem operasi seperti Linux, Windows maupun Unix[19]. Berikut ini adalah beberapa kelebihan pemrograman Java dibandingkan dengan bahasa pemrograman lainnya.

1. Bersifat portable dan platform independent

Java dapat dijalankan di berbagai platform dengan syarat tersedianya JVM pada platform tersebut.

1. Pembuang Sampah (Garbage Collection)

Java melakukan pembuangan sendiri “sampah-sampah” yang tidak berguna, artinya alokasi memori pada objek java dilakukan secara otomatis. Ini membuat program Java bebas dari permasalahan pengelolaan memori.

1. Bebas Arsitektur (Architecture Neutral)

Pada dasarnya, Java dirancang untuk bekerja pada berbagai arsitektur prosesor dan berbagai sistem operasi. Oleh karena itu, program java bekerja sebagai interpreter untuk mengubah program Java menjadi byte-code sebelum dijalankan pada suatu mesin tertentu.

1. Tangguh (Robust)

Berdasarkan hasil intepreter Java dapat mengecek seluruh akses sistem yang dilakukan oleh program, maka Java tidak akan membuat sistem menjadi crash. Bila terjadi permasalahan serius, program Java membuat pengecualian (exception) yang dapat ditangani dan dikelola oleh program tanpa beresiko memacetkan sistem.

1. Aman (Secure)

Sistem Java sangat handal dalam mengelola memori, tidak hanya memverifikasi seluruh akses ke memori, tetapi juga menjamin tidak ada virus yang dapat menyerang program yang sedang berjalan. Penyebabnya dikarenakan Java tidak mendukung pointer, sehingga program tidak mendapat akses ke area sistem dimana ia tidak mendapat otorisasi.

1. Dapat Diperluas

Program Java mendukung metode native code, yaitu memungkinkan pemrogram menulis fungsi dalam bahasa lain, biasanya C++ yang dapat dieksekusi secara lebih cepat karena langsung berjalan di atas perangkat keras yang bersangkutan, dari pada yang ditulis dalam Java yang berjalan di atas JVM. Metode native code dikaitkan secara dinamis (dynamically linked) ke program Java, yakni dikaitkan dengan program saat runtime.

### **PHP**

PHP adalah singkatan dari “Hypertext Preprocessor”, yang berarti ada sebuah bahasa scripting tingkat tinggi yang dipasang pada dokumen HTML. Sebagian besar style dan sintaks dalam PHP mirip dengan bahasa C, Java dan Perl, namun pada PHP ada beberapa fungsi yang lebih spesifik. Adapun tujuan utama dari penggunaan bahasa ini adalah untuk membuat website menjadi dinamis dan dapat dilakukannya pengelolaan data secara otomatis[20].

Contoh terkenal dari aplikasi PHP adalah forum (phpBB) dan MediaWiki (software di belakang Wikipedia). PHP merupakan salah satu dari sekian banyak pilihan seperti ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Wordpress, Mambo, Joomla! Postnuke, Xaraya, dan lain-lain.

### ***Database***

1. Definisi Database

Database atau yang biasa disebut basis data adalah sebuah tempat penyimpanan dimana terdapat kumpulan data besar yang tidak hanya berisi data operasional tetapi juga deskripsi data. Database adalah kumpulan data yang saling terhubung secara logis dan deskripsi dari data tersebut, dirancang untuk menemukan informasi yang dibutuhkan oleh sebuah organisasi [21]. Dalam merancang database, salah satu hal yang perlu diperhatikan adalah efisiensi. Redudansi data yang terlalu banyak dapat mengurangai efisiensi pada database sehingga perlu adanya normalisasi.

1. Pengertian *Database Management System* (DBMS)

Secara umum dapat diartikan bahwa *Database Management System* (DBMS) adalah suatu sistem software yang memungkinkan user dapat mendefinisikan, membuat, dan memelihara serta menyediakan akses kontrol terhadap data. Seperti dijelaskan sebelumnya database sendiri adalah sekumpulan data yang berhubungan dengan secara logika dan memiliki beberapa arti yang saling berpautan[22]. DBMS menyediakan fasilitas sebagai berikut:

1. *Data Definition Language* *(DDL)* mengizinkan pengguna untuk memberikan deskripsi pada database, misalnya seperti menentukan tipe dan batasan data yang akan disimpan dalam database.
2. *Data Manipulation Language (DML)* mengizinkan pengguna untuk melakukan manipulasi data, seperti memasukan data, menghapus data, dan mendapatkan data dari database.
3. *Data Control Language (DCL)* mengizinkan pengguna untuk melakukan akses terkontrol ke database, misalnya *security system integrity system, concurrency control system, recovery control system, user-accesible catalog.*
4. MySQL

MySQL adalah sebuah program database server yang mampu mengirim dan menerima data dengan sangat cepat, kelebihan lain adalah bisa multiuser serta menggunakan perintah dasar SQL (Structured Query Language). MySQL mempunyai dua bentuk lisensi, yaitu FreeSoftware dan Shareware. Untuk yang biasa kita gunakan adalah MySQL FreeSoftware yang berada dibawah Lisensi GNU/GPL (General Public License).

Selain database server, MySQL juga merupakan program yang dapat mengakses suatu database MySQL yang berposisi sebagai Server, yang berarti program kita berposisi sebagai Client. Jadi intinya adalah MySQL dapat digunakan sebagai client mupun server. Database MySQL juga berbentuk database relasional atau disebut Relational Database Management System (RDBMS) yang menggunakan suatu bahasa permintaan yang bernama SQL (Structured Query Language)[23].

1. Javascript Object Notation (JSON)

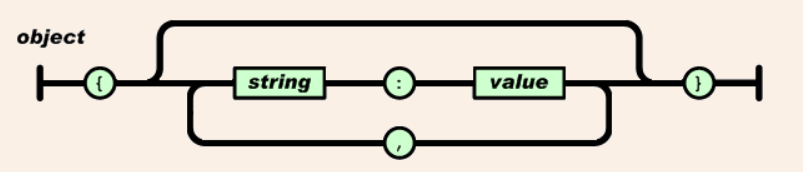
JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999 (http://www.json.org).[24]

JSON adalah salah satu bahasa markup yang dapat melakukan pertukaran data. Dengan membuat sebuah JSON sama halnya dengan kita membuat sebuah object pada javascript itu sendiri sehingga memudahkan bagi para developer[24].

Struktur dari JSON adalah sebagai berikut :

1. *JSON Object*

JSON Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan attribut dipisahkan oleh , (koma). Struktur dari objek dapat dilihat dari gambar 2.4 berikut :



Gambar 2.4 JSON Object

1. *JSON Array*

JSON Array adalah kumpulan nilai yang terurutkan. Array dimulai dengan [ (kurung kotak buka) dan diakhiri dengan ] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma). Struktur dari array dapat dilihat dari gambar 2.5 berikut :



Gambar 2.5 JSON Array

1. *Value*

Value dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah Array. Struktur-struktur tersebut dapat disusun bertingkat. Struktur dari value dapat dilihat dari gambar 2.6 berikut :



Gambar 2.6 Value

### **Metodologi Pengembangan Perangkat Lunak**

Hampir setiap *user* dalam pengembangan perangkat lunak tidak mengetahui kebutuhan *input*, proses, dan *output*. Pengembang mungkin kurang yakin dengan pengambilan algoritma, dari adaptasi penggunaan sistem, atau dari iteraksi antara manusia dan mesin. Oleh karena itu metode *prototyping* cocok dengan penelitian ini.

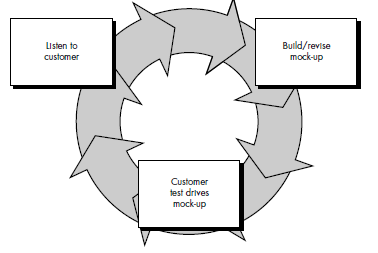
Fase pengembangan yang dimulai dari mengumpulkan kebutuhan. Pengembang dan pelanggan akan bertemu lalu menyusun tujuan dari pengembangan perangkat lunak ini, mengidentifkasi kebutuhan yang mereka inginkan, dan garis besar yang sangat penting dalam pengembangan ini. Lalu kemudian dibuatlah desain sementara. Desain tersebut berfokus pada representasi dari kebutuhan perangkat lunak yang dibutuhkan oleh *user* seperti *input* dan *output*nya. Desain sementara tersebut mengarah ke pengembangan dari sebuah *prototype*. *Prototype* tersebut kemudian dievaluasi oleh *user* dan digunakan untuk menyaring kebututuhan perangkat lunak yang akan dibangun. [25]

Adapun cakupan aktivitas dari *prototyping model* terdiri dari:

1. Mendefinisikan objektif secara keseluruhan dan mengidentifikasi kebutuhan yang sudah diketahui.
2. Melakukan perancangan secara cepat sebagai dasar untuk membuat *prototype*.
3. Menguji coba dan mengevaluasi *prototype* dan kemudian melakukan penambahan dan perbaikan-perbaikan terhadap *prototype* yang sudah dibuat.

Metode pengembangan perangkat lunak model *prototype* dirancang agar dapat menerima perubahan-perubahan dalam rangka menyempurnakan *prototype* yang sudah ada sehingga pada akhirnya dapat menghasilkan perangkat lunak yang dapat diterima dan perubahan-perubahan yang terjadi dapat dianggap merupakan bagian dari proses pengembangan itu sendiri.

Secara ideal *prototype* berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. *Prototype* bisa menjadi paradigma yang efektif bagi rekayasa perangkat lunak. Kuncinya adalah mendefinisikan aturan-aturan main pada saat awal, yaitu pelanggan dan pengembang keduanya harus setuju bahwa *prototype* dibangun berfungsi sebagai mekanisme pendefinisian kebutuhan seperti pada Gambar 2.7 berikut [25].



Gambar 2.7Metode *Prototype*

Namun, pada dasarnya semua metode pengembangan Perangkat Lunak secara umum sama yaitu terdiri dari fase-fase berikut:

1. Fase Pendefinisian
2. *Planning*.
3. Pendefinisian perangkat lunak yang dikembangkan.
4. Alokasi sumber daya yang diperlukan.
5. *Scheduling*.
6. Estimasi biaya.
7. Mendefinisikan alokasi biaya yang akan dikeluarkan untuk pengembangan sistem yang akan dibangun.
8. Fase Pengembangan
9. Perancangan.
10. Pendeskripsian arsitektur perangkat lunak yang akan di bangun.
11. Fase Verifikasi
12. Verifikasi untuk semua unit fungsional.
13. Pengetesan fungsi-fungsi dan *interface*.
14. Perawatan (*maintenance*).
15. Mengecek ulang kesalahan-kesalahan perangkat lunak.
16. Adaptasi terhadap perubahan lingkungan.
17. Menambahkan dan mengembangkan kebutuhan yang akan datang.

### **Unified Modeling Language (UML)**

Pada perkembangan teknik pemrograman berorientasi objek, munculah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modelling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. 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 [26].

Notasi notasi dasar UML dapat dilihat pada Tabel 2.3 berikut ini.

Tabel 2.3 Notasi UML

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *Actor* | Mendefinisikan kumpulan entitas atau peran yang berada dan berinteraksi dengan *use case* |
| 2 |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*dependent)* akan mempengaruhi elemen yang bergantung padanya dalam hal ini yaitu elemen yang tidak mandiri |
| 3 |  | *Generalization* | Hubungan dimana objek anak (*descendent)* berbagi perilaku dan struktur data dari objek yang berada di induk (*ancestor)* |
| 4 |  | *Include* | Menspesifikasikan *use case* sumber secara eksplisit |
| 5 |  | *Extend* | Menspesifikasikan bahwa *use case* target memperluas prilaku dari *use case* sumber pada suatu titik yang diberikan |
| 6 |  | *Association* | Hubungan antara objek yang satu dengan objek yang lainnya |
| 7 |  | *System* | Menspesifikasikan paket yang ditampilkan sistem secara terbatas |
| 8 |  | *Use Case* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem dimana menghasikan suatu hasil yang terukur bagi aktor |

Adapun tujuan pemodelan dengan menggunakan UML adalah sebagai berikut:

* + - 1. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemograman dan proses rekayasa.
      2. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
      3. UML juga berfungsi sebagai sebuah cetak biru *(blueprint)* karena sangat lengkap dan detail. Dengan *blueprint* maka akan bisa diketahui informasi secara detail tentang coding program atau bahkan membaca program dan mengaplikasikannya kembali ke dalam bentuk diagram (*reserve engineering*).

Memberikan model yang siap pakai, karena bahasa pemodelan visual yang ekpresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.

### **Pengujian Perangkat Lunak**

Pengujian perangkat lunak adalah suatu elemen yang dapat menguji kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain dan pengkodean. Pengujian yang dilakukan mencari kesalahan yang terjadi pada pengembangan perangkat lunak. Selama definisi awal dari fase pembangunan, pengembangan berusaha untuk membangun perangkat lunak dari konsep yang abstrak sampai dengan implementasi [27].

Pentingnya pengujian perangkat lunak dan implikasi yang dapat mengacu pada kualitas perangkat lunak tidak dapat terlalu ditekan karena melibatkan sederetan aktivitas produksi dimana peluang terjadinya kesalahan manusia sangat besar dan karena ketidakmampuan manusia untuk melakukan dan berkomunikasi dengan sempurna maka pengembangan perangkat lunak diiringi dengan aktivitas jaminan kualitas [27].

* + - * 1. Pengujian *Blackbox*

Pengujian black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian black-box memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian black-box bukanlah alternatif dari pengujian white-box, tetapi mungkin pendekatan komplementer yang kemungkinan besar mampu mengungkap jelas kesalahan dari pada metode white-box. [28]

Pengujian black-box berusaha menemukan kesalahan dalam kategori sebagai berikut:

1. Fungsi-fungsi yang tidak benar/hilang
2. Kesalahan interface.
3. Kesalahan dalam struktur data atau akses basis data eksternal.
4. Kesalahan kinerja
5. Inisialisasi dari kesalahan terminasi.

Tidak seperti white box yang melakukan pengujian diawal, pengujian black-box cenderung diaplikasikan selama tahap akhir pengujian. Karena pengujian black-box memperhatikan struktur kontrol, maka perhatian akan berfokus pada domain informasi [28].

* 1. **Metodologi Penelitian**

### **Metodologi Penelitian**

Adapun metode penelitian yang digunakan dalam penelitian ini dilakukan dalam beberapa tahap, di antaranya yaitu :

1. Pengumpulan Data

Pada tahap pengumpulan data ada 2 tahapan yang dilakukan di antaranya yaitu :

1. Observasi

Langkah pertama dalam pengumpulan data adalah dengan mengadakan penelitian dan peninjauan langsung terhadap objek masalah yang diangkat.

1. Studi Literatur

Setelah itu mengumpulkan beberapa bahan acuan seperti literatur, jurnal, *paper,* dan bahan lain yang berhubungan dengan topik penelitian yang diambil.

1. Analisis

Analisis dilakukan terhadap data yang telah dikumpulkan, untuk mendefinisikan perangkat lunak dan perangkat keras yang akan dibuat. Analisis merupakan tahap yang sangat penting sebelum dilakukannya perancangan.

1. Desain

Tahapan desain dilakukan setelah analisis. Dalam tahap ini kebutuhan-kebutuhan yang didapati pada tahap analisis dibuat menjadi bentuk yang mudah dipahami dan gambaran teknis dari sistem.

1. Implementasi Program

Setelah aplikasi didesain sedemikian rupa dengan mengikuti metode yang baik dan benar maka selanjutnya masuk ke tahap implementasi program dengan menulis kode program sesuai dengan bahasa pemrograman yang digunakan.

1. Pengujian

Tahap selanjutnya setelah aplikasi diimplementasi maka program diuji dengan *black box testing* untuk memastikan bahwa aplikasi berjalan dengan semestinya.

### **Metodologi Pengembangan**

Metode pengembangan yang dipakai dalam mengembangkan aplikasi ini yaitu dengan menggunakan metodologi *Prototype.* Adapun tahapan yang dilakukan dalam metode *Prototype* ini dapat dijelaskan sebagai berikut.

1. *Listen to customer*

Pada tahap ini dilakukan pengumpulan ide dari kebutuhan sistem yang akan dibangun. Untuk membuat suatu sistem yang sesuai kebutuhan maka dapat dengan cara melakukan wawancara kepada *customer* atau bisa dengan melihat sistem yang sedang berjalan untuk mengetahui ide pokok masalah.

1. *Build mockup / Revise mockup*

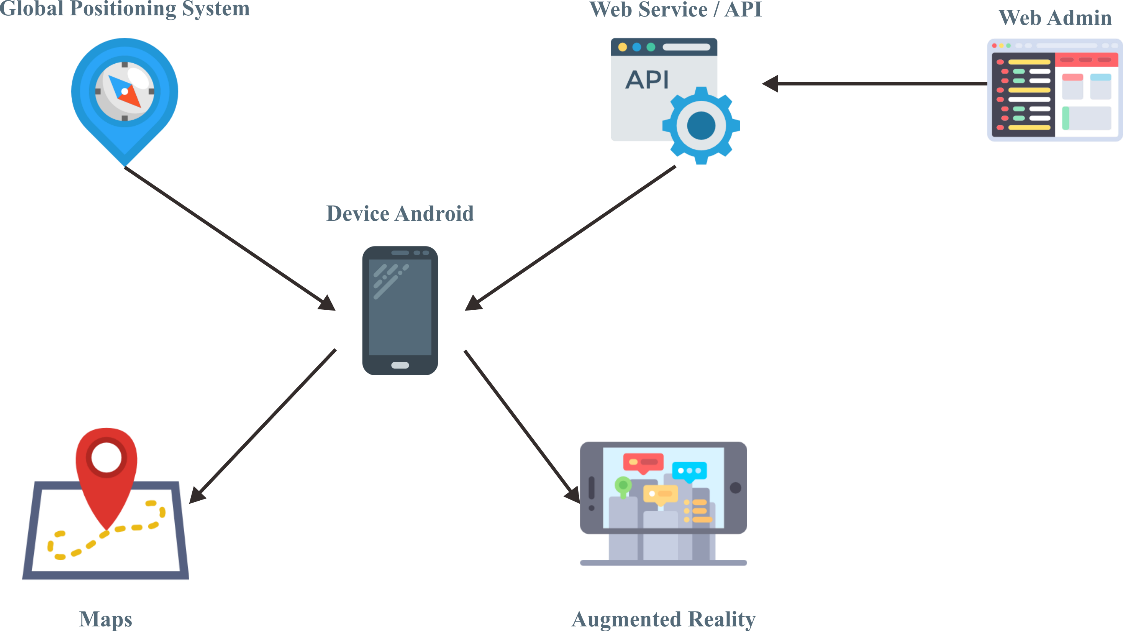
Setelah mengetahui kebutuhan, dilakukan perancangan dan pembuatan *blue print / prototype* sistem. *Prototype* yang dibuat harus sesuai dengan kebutuhan yang telah didefinisikan pada tahap sebelumnya.

1. *Customer test drives mockup*

Kemudian, *prototype* dari sistem akan diuji coba oleh user sekaligus melakukan evaluasi terhadap kekurangannya. Pengembang kemudian melakukan kembali tahap pertama untuk mendengarkan keluhan *user* dan melakukan revisi terhadap *prototype* yang tidak sesuai dengan kebutuhan *user,* apabila sudah tidak ada revisi bisa dilanjutkan ke tahap berikutnya.

* 1. **Arsitektur Sistem**

Secara umum rancangan arsitektur sistem secara keseluruhan dapat dilihat pada Gambar 1.9.1.



**Gambar 1.9.1 Arsitektur Sistem Keseluruhan**

Adapun penjelasan rancangan sistem secara keseluruhan adalah sebagai berikut :

1. *Global Positioning System (GPS)*.

Dalam hal ini GPS digunakan untuk mendapatkan lokasi pengguna. Jadi dengan menggunakan GPS pengguna dapat mengetahui posisinya berada ketika menjalankan aplikasi. Data yang diambil dari GPS tersebut berupa koordinat longitude dan latitude dari *smartphone* yang digunakan pengguna berada.

1. Web Admin

Web admin digunakan untuk mengelola data-data pondok pesantren seperti pengelolaan kategori/aliran pondok pesantren ataupun data pondok pesantren itu sendiri.

1. *Web Service / API*

Pada dasarnya *web service* adalah sebuah jembatan yang mengubungkan antara database dan aplikasi. Dalam hal ini *web service* digunakan untuk menampilkan data-data pondok pesantren se-Jawa Barat dengan format JSON.

1. *Android Device*

Berfungsi untuk mengakses informasi yang didapatkan dari GPS dan *Web Service.* Selain itu juga, untuk mengakses fitur-fitur utama yang ada pada aplikasi.

1. *Maps*

Berfungsi untuk menampilkan informasi berupa peta mengenai posisi dari pengguna dan pondok pesantren berada.

1. *Augmented Reality*

Dalam hal ini, *augmented reality* berguna untuk menampilkan visualisasi berupa marker yang berisi informasi pondok pesantren dan jarak tempuhnya secara *realtime*.

* 1. **Jadwal Kegiatan**

Adapun jadwal kegiatan selama penelitian berlangsung digambarkan dalam Tabel 1.10.1 berikut.

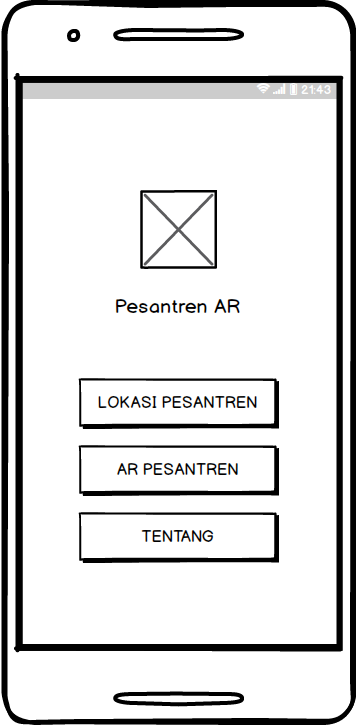
**Tabel 1.10.1 Tabel Jadwal Kegiatan**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Jenis Kegiatan** | **Januari** | | | | **Februari** | | | | **Maret** | | | | **April** | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1 | Listen to Customer |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Builds Mockup |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Costumer Test Drives Mockup |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* 1. ***Mock Up* Aplikasi**

Berikut adalah rancangan antarmuka sistem yang akan nantinya akan dibuat.

1. Tampilan menu utama ketika masuk ke aplikasi.



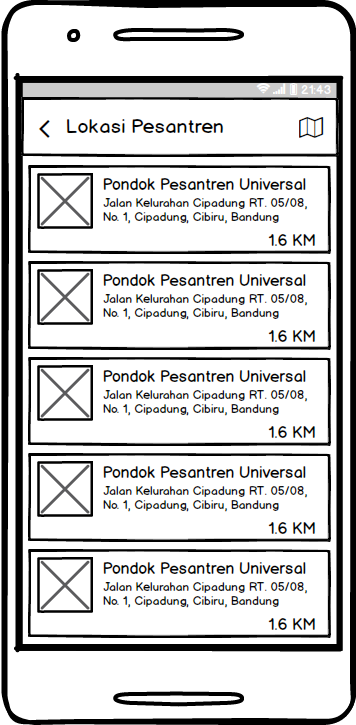
**Gambar 1.11.1 Tampilan Menu Utama**

1. Tampilan daftar lokasi pesantren dalam bentuk map.



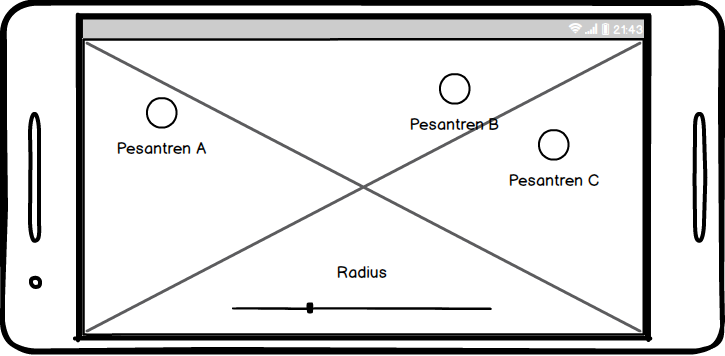
**Gambar 1.11.2 Tampilan Lokasi Pesantren (Map)**

1. Tampilan daftar lokasi pesantren dalam bentuk list.



**Gambar 1.11.3 Tampilan Lokasi Pesantren (List)**

1. Tampilan kamera *Augmented Reality* untuk menampilkan *marker* lokasi pesantren.



**Gambar 1.11.4 Tampilan Kamera *Augmented Reality***

1. Tampilan detail pesantren



**Gambar 1.11.5 Tampilan Detail Pesantren**

# **DAFTAR PUSTAKA**

[1] Z. Dhofier, *Tradisi Pesantren*. Jakarta: LP3ES, 1994.

[2] F. Febriana, “Aplikasi Panduan Tur Kampus dengan Teknologi Augmented Reality Berbasis Android (Studi Kasus Universitas Islam Negeri Sunan Gunung Djati Bandung),” 2014.

[3] M. Ilyas, “Aplikasi Pencarian Lokasi dan Jarak Dinas-Dinas di Kabupaten Subang Dengan Augmented Reality Menggunakan Metode Location Based Services Berbasis Android,” 2015.

[4] S. Chanphearith and A. J. Santoso, “Analysis and Implementation of Location-Based Augmented Reality Mobile Application for Searching Tourist Attractions and Culinary Places in Phnom Penh,” vol. 4, no. 6, pp. 126–136, 2016.

[5] A. Hardiansyah, “Designing Android Based Augmented Reality Location-Based Service Application,” vol. 2, no. 2, pp. 110–115, 2017.

[6] S. K. Anwar, “Implementasi Augmented Reality Menggunakan Location Based Service dan Haversine Formula Untuk Penunjang Fasilitas Umum Di Kabupaten Pangandaran,” 2017.

[7] I. Permana, O. D. Nurhayati, and K. T. Martono, “Location Based Service sebagai Penunjuk Lokasi Hotel di Kota Semarang berbasis Augmented Reality,” vol. 3, no. 4, pp. 461–470, 2015.

[8] Y. P. Atmojo, I. G. Ngurah, and D. Paramartha, “Pemanfaatan Augmented Reality Pada Sistem Informasi Geografis Kampus Di Bali,” pp. 63–74, 2014.

[9] G. Laksono, “Pemanfaatan Teknologi Augmented Reality Markerless Sebagai Media Pengenalan Gedung Universitas Kanjuruhan Malang Berbasis Android,” 2016.

[10] A. Budianto and E. R. Nainggolan, “Perancangan Aplikasi Islamic Boarding School Finder berbasis Android menggunakan Algoritma Dijkstra,” pp. 153–160, 2016.

[11] A. Fajri, E. P. Purwandari, and F. F. Coastera, “Implementasi Sistem Informasi Geografis Untuk Pemetaan Indekos Dengan Jarak Terpendek Menggunakan Metode Algoritma Dijkstra ( Studi Kasus : Kelurahan Kandang Limun dan Beringin Raya ),” vol. 6, no. 1, pp. 69–79, 2018.

[12] R. Hermawan, D. Putra, H. Sujiani, and N. Safriadi, *Penerapan Metode Haversine Formula Pada Sistem Informasi Geografis Pengukuran Luas Tanah*, 1st ed. 2016.

[13] S. Arfianto, “Sistem Aplikasi Penentuan Rute Terpendek pada Jaringan Multi Moda Transportasi Umum Menggunakan Algoritma Dijkstra,” 2002.

[14] R. T. Azuma, “A Survey of Augmented Reality. Presence: Teleoperators and Virtual Environments 6,” 1997.

[15] N. Safaat, *Android (Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android)*. Bandung: Informatika, 2011.

[16] “http://jeniferhartoyo.blogspot.com/2013/04/tugas-pi-teori-web-service-soap-and-rest.html.” .

[17] H. . Abidin, *Penentuan Posisi dengan GPS dan Aplikasinya*. Jakarta: Pradnya Paramita, 2000.

[18] P. Naughton *et al.*, “Sejarah Java Apa itu Teknologi JAVA ? Mengapa Mempelajari JAVA ?,” pp. 1–4, 1995.

[19] C. Warth, E. Frank, and M. Sheridan, “A . Kilas Balik Sejarah Java,” pp. 1–7, 1990.

[20] A. Lena and K. Ratna, “Pengertian PHP dan MySQL,” *Ilmu Teknol. Inf.*, pp. 2–3, 2008.

[21] J. A. Hall, “Database Management System,” in *Accounting Information Systems*, 4th ed., South Western, 2004, pp. 9–6.

[22] Rahmayanti, “Database,” in *Pengantar Basis Data*, 2009, pp. 2–4.

[23] H. Saputro, “MODUL PEMBELAJARAN PRAKTEK BASIS DATA ( MySQL ),” p. 34, 2012.

[24] 2012:38) (Tata Sutabri, “BAB II Landasan Teori,” *Google.com*, p. 20, 2011.

[25] “http://anyes48.blogstudent.mb.ipb.ac.id/2012/09/25/siklus-pengembangan-sistem-informasi-dan-prototyping-yang-digunakan-dalam-perusahaan-bagi-end-user-dan-is-specialist/.” .

[26] R. A.S and M. Salahudin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasasi Objek*. Bandung: Informatika, 2013.

[27] L. Linda, “Pengujian Perangkat Lunak (Software Testing),” *1991, Proceedings. Int. Test Conf.*, p. 1110, 2015.

[28] R. S. Pressman, *Rekayasa Perangkat Lunak Pendekatan Praktisi Buku 1*, 2nd ed. Yogyakarta: Andi, 2002.

# **LAMPIRAN**