

BAB II

TINJAUAN PUSTAKA

2.1 Instalasi Gawat Darurat (IGD)

Gawat Darurat adalah keadaan klinis yang membutuhkan tindakan medis segera untuk penyelamatan nyawa dan pencegahan kecacatan (Permenkes RI No. 47 tahun 2018). Pelayanan kegawat daruratan adalah tindakan medis yang dibutuhkan oleh pasien gawat darurat dalam waktu segera untuk menyelamatkan nyawa dan pencegahan kecacatan (Permenkes RI No. 47 tahun 2018). Fasilitas Pelayanan Kesehatan adalah suatu alat dan/atau tempat yang digunakan untuk menyelenggarakan upaya pelayanan kesehatan, baik promotif, preventif, kuratif maupun rehabilitatif yang dilakukan oleh pemerintah pusat, pemerintah daerah, dan/atau masyarakat (Permenkes RI No. 47 tahun 2018).

IGD adalah salah satu unit pelayanan di Rumah Sakit yang menyediakan penanganan awal (bagi pasien yang datang langsung ke rumah sakit)/lanjutan (bagi pasien rujukan dari fasilitas pelayanan kesehatan lain), menderita sakit ataupun cedera yang dapat mengancam kelangsungan hidupnya (Permenkes RI No. 47 tahun 2018). IGD berfungsi menerima, menstabilkan dan mengatur Pasien yang membutuhkan penanganan kegawat daruratan segera, baik dalam kondisi sehari-hari maupun bencana (Permenkes RI No. 47 tahun 2018). IGD berfungsi menerima, menstabilkan dan mengatur pasien yang membutuhkan penanganan kegawatdaruratansegera, baik dalam kondisi sehari - hari maupun bencana (Permenkes RI No. 47 tahun 2018).

2.2 Web Mobile

Mobile web adalah halaman HTML berbasis browser yang diakses menggunakan perangkat portable (smartphone atau tablet) melalui jaringan seluler seperti 3G, 4G, 5G maupun Wifi. Mobile Web dirancang untuk menampilkan data seperti teks, gambar, dan video dari sebuah website kedalam sebuah tampilan yang lebih kecil yakni perangkat mobile

Berikut contoh gambar dari web mobile:



Gambar 2. 1 Web mobile

2.3 PHP (Hypertext preprocessor)

Hypertext preprocessor (PHP) yaitu Bahasa pemrograman web serverside yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server. PHP adalah script yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Semua script PHP dieksekusi pada server di mana script tersebut dijalankan. Kode PHP diawali dengan `<?php`. Pasangan kedua kode inilah yang berfungsi sebagai tag kode PHP. Berdasarkan tag inilah, pihak server dapat memahami kode PHP dan kemudian memproses. Hasilnya dikirim ke browser (Anhar,2010)

Contoh halaman yang menggunakan PHP adalah Facebook. Buka *home page* mereka, dan Anda bisa melihat bahwa URL-nya diakhiri dengan `.php` (`facebook.com/home.php`). Dengan kata lain, halaman ini dibuat dengan file PHP (`home.php`) yang memuat gabungan kode *hypertext preprocessor* dan tag HTML.

Berikut script PHP sederhana:

```
<html>
<head>
<title>Percobaan PHP</title>
</head>
```

```
<body>
<?php echo '<p>Salam dari Hostinger</p>'; ?>
</body>
</html>
```

Skrip ini adalah contoh paling sederhana karena Anda tidak perlu menjalankannya dengan cara apa pun. Kami hanya menggunakan pernyataan **echo** untuk menginstruksikan halaman agar menampilkan teks **Salam dari Hostinger**. Namun, contoh ini bisa menunjukkan bagaimana situs Anda akan menafsirkan kode PHP ke dalam HTML.

2.4 Mysql

MySQL adalah sebuah database management system (manajemen basis data) menggunakan perintah dasar SQL (Structured Query Language) yang cukup terkenal. Database management system (DBMS) MySQL multi pengguna dan bersifat gratis. Mysql digunakan sebagai wadah dalam mengelola data yang dapat disimpan digunakan Kembali dengan cara yang lebih efisien (Setyawan and Pratiwi, 2019). Ada tiga bentuk SQL yang perlu Anda ketahui, yaitu *Data Definition Language (DDL)*, *Data Manipulation Language (DML)*, dan *Data Control Language (DCL)*.

2.5 Algoritma Haversine

Haversine formula merupakan sebuah algoritma yang diterapkan secara matematis untuk digunakan dalam navigasi dengan memberikan perhitungan jarak lingkaran besar antar dua titik di permukaan bumi yang didasarkan pada garis bujur dan garis lintang bumi. Titik tersebut berada pada permukaan bola bumi dimana garis longitude dan latitude. Rumus ini di temukan oleh *James Andrew* pada tahun 1805. Kemudian dipergunakan oleh *Josef de Mandoza y rios* pada tahun 1801. *Prof James Inman* memperkenalkan istilah *haversine* untuk pertama kali pada tahun 1835. Rumus *Haversine* juga di gunakan jarak mengukur Bintang.

Berikut adalah penjelasan rumus *Haversine*:

Rumus *Haversine* formula

$$\theta = \frac{d}{r}$$

$$\text{haversine}(\theta) = \sin^2\left(\frac{\theta}{2}\right)$$

$$\frac{d}{r} \text{haversine}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{haversine}(\lambda_2 - \lambda_1)$$

$$d = r \cdot \text{haversine}^{-1}(h)$$

$$= 2r \sin^{-1}(\sqrt{h})$$

$$d = 2r \sin^{-1}\left(\sqrt{\sin^2 \frac{\phi_2 - \phi_1}{2} \cos(\phi_1) \cos(\phi_2) \sin^2 \frac{\lambda_2 - \lambda_1}{2}}\right)$$

dimana :

d = Jarak antara dua titik pada garis permukaan bola

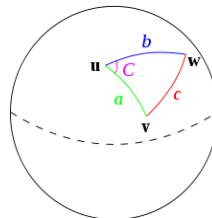
r = Jari – jari bola

p = latitude

λ = longitude

h = $\text{haversine}(\theta)$

Rumus *haversine* memiliki banyak bentuk rumus-rumus sesuai kegunaan yang berbeda seperti perhitungan luas, menghitung jarak penerbangan. Algoritma pada rumus ini digunakan untuk mengukur jarak antara titik koordinat GPS Titik koordinat posisi pengguna dengan titik tujuan. Algoritma adalah kunci untuk menentukan jarak posisi pengguna dengan titik tujuannya.



Gambar 2. 2 Model Bola bumi

Gambar 1 menjelaskan tentang segitiga pada bola bumi dihitung menggunakan hukum rumus haversine. jika Panjang dari ketiga sisi adalah a (u ke v), b (dari u untuk w), dari c (dari v ke w), dari sudut sebaliknya adalah C, maka hukum haversine sebagai berikut

$$\text{Haversine}(c) = \text{haversine}(a-b) + \sin(a)\sin(b)\text{haversine}(c)$$

Rumus diterapkan pada penelitian ini seperti berikut:

$$R = \text{Radian Bumi} = 6371 \text{ km.}$$

$$\Delta \text{lat} = \text{lat2} - \text{lat1}$$

$$\Delta \text{long} = \text{long2} - \text{long1}$$

$$a = \sin^2(\Delta \text{lat}/2) + \cos(\text{lat1}) * \cos(\text{lat2}) * \sin^2(\Delta \text{long}/2) \quad c = 2 * \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R * c$$

2.6 Penelitian Terkait

Pada pembuatan penelitian ini, terdapat beberapa inspirasi yang berasal dari penelitian yang terkait dengan permasalahan dan dijadikan sebagai referensi pada penelitian ini :

Tabel 2. 1 Penelitian Terkait

No	Penulis	Judul	Metode	Hasil
1	Zelvia Ayu Puspita1) , Fauziah2) , Ira Diana Sholihati3)	METODE HAVERSINE FORMULA PADA PENCARIAN RUMAH SAKIT DI WILAYAH JAKARTA SELATAN BERBASIS ANDROID	Algoritma Haversine	Dengan adanya aplikasi pencarian ini membantu masyarakat untuk menerima informasi tentang lokasi, rute dan jarak rumah sakit terutama dalam keadaan darurat tertentu dengan mencari rumah sakit terdekat.
2	Eko Budi Susanto1 , Paminto Agung Christianto1,* Mohammad Reza Maulana2 Bambang Ismanto3	Penerapan Google Maps API dan Formula Haversine Pada Aplikasi Ambulance Rakyat	Algoritma Haversine	Aplikasi ambulance rakyat dibangun untuk memberikan salah satu solusi bagi Pemerintah Kota Pekalongan pada permasalahan tindakan kegawatdaruratan. Aplikasi ambulance rakyat dapat digunakan oleh masyarakat Kota Pekalongan ketika membutuhkan transportasi

				(ambulance) menuju layanan medis terdekat
3	UUNG UNGKAWA1 ,DWIKI FAIZAL M2	APLIKASI PENENTUAN RUTE TERPENDEK DALAM MENENTUKAN UNIT GAWAT DARURAT	Algoritma Haversine	Pada penelitian ini penggunaan algoritma a star dalam studi kasus diatas dapat diimplementasikan lalu memberikan hasil rute dari lokasi pengguna menuju lokasi tujuan rumah sakit berdasarkan jarak.
4	Rahmi Hidayati1,*, Nurul Mutiah2	Penerapan Metode Haversine Formula Pada Pencarian Lokasi Fasilitas Kesehatan Terdekat	Algoritma Haversine	yang diperoleh pada penelitian ini adalah aplikasi pencarian lokasi fasilitas kesehatan terdekat di kota Pontianak mendapatkan urutan lokasi fasilitas kesehatan terdekat dengan membandingkan data lokasi pengguna berada dengan lokasi- lokasi fasilitas kesehatan yang ada disekitar pengguna,

				menghitung selisih jarak yang paling terkecil sebagai lokasi terdekat.
--	--	--	--	--