BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Penelitian terkait bertujuan untuk mengambil beberapa refrensi jurnal terkait yang digunakan dalam mendukung penelitian publikasi ilmiah dalam junal lokal yaitu:

1. Peneliti yang dilakukan oleh Yulianto, Dkk (2018) dengan judul Penerapan Formula Haversine Pada Sistem Informasi Geografis Pencarian Jarak Terdekat Lokasi Lapangan Futsal. Berdasarkan penelitian yang dilakukan, dijelaskan bahwa Formula Haversine dapat dimanfaatkan sebagai pencarian jarak terdekat dengan cara mencari hasil yang paling kecil nilainya sebagai lokasi dengan jarak terdekat.
2. Penelitian yang dilakukan oleh Muhammad Arizal Fahribi Cahyo Firdous (2017) dengan judul Sistem Pencarian Jarak Terdekat Menggunakan Metode Haversine Berbasis *Web*. Berdasarkan penelitian yang dilakukan, dijelaskan dengan adanya sistem pencarian jarak terdekat ini mampu membantu pengguna kendaraan bermotor pada jalur utama Kediri-Blitar, Kediri-Tulungagung dan Kediri-Nganjuk untuk menemukan bengkel terdekat dari lokasi *user* berada memanfaatkan GPS dan aplikasi *Google Maps* yang ada pada perangkat *web user*.
3. Penelitian yang dilakukan oleh Fitri Maharany (2015) dengan judul Pengembangan Sistem Pencarian Lokasi Kuliner Di Kota Malang Dengan

Location Based Service. Berdasarkan hasil pengujian aplikasi yang dilakukan maka dapat disimpulkan bahwa Sistem Pencarian Lokasi di Kota Malang yang dibangun sudah berjalan sesuai dengan yang diharapkan dan didapatkan hasil bahwa semua menu utama serta fitur-fitur yang dibuat Sistem Pencarian Lokasi Kuliner dapat berjalan sesuai fungsinya.

Berdasarkan penelitian di atas penulis menemukan perbedaan dari penelitian yang akan dilakukan oleh penulis yaitu: Bahwa penelitian ini menerapkan metode *haversine* dalam perhitungan jarak lokasi sama seperti penelitian lainnya namun berbeda dalam proses penerapannya dan objek penelitian yang dilakukan. Dalam menyelesaikan penelitian menerapkan peta *Maps Api* sehingga sistem sangat mudah digunakan, sangat efektif dan efisien dalam pencarian lokasi survey angkatan Kerja Nasional yang dilakukan oleh Pegawai Badan Pusat Statistik Kota Medan.

II.2. Uraian Teoritis

Penelitian yang dilakukan dengan cara langsung dan dengan beberapa sistem pendukung untuk menyelesaikan judul pencarian lokasi Survey angkatan kerja nasional dengan menggunakan metode *haversine*, aplikasi dan bahasa program yaitu :

II.2.1. Sistem Informasi Geografis

Sistem Informasi Geografis (GIS) adalah suatu sistem informasi menyangkut keberadaan obyek di permukaan bumi berikut informasi yang terkandung di dalamnya yang menpunyai keterkaitan secara

geografis dengan obyek lainnya. Dengan sistem ini data dapat dikelola, dilakukan manipulasi untuk keperluan analisis secara komprehensif. (Fitri Maharany : 2015).

II.2.2. *Dreamweaver*

Dreamweaver adalah aplikasi desain dan pengembangan web yang menyediakan editor *Wysiwyg* visual (bahasa sehari-hari yang disebut sebagai *Design view*) dan kode editor dengan fitur standar seperti *syntax highlighting, code completion* dan *code collapsing* serta fitur lebih canggih seperti *real-time syntax checking* dan *code introspection* untuk menghasilkan petunjuk kode untuk membantu pengguna dalam menulis kode. (Mara Destiningrum : 2015).

II.2.3. *Web*

Web adalah sekumpulan halaman yang menampilkan informasi data teks, data gambar, data animasi, suara, video dan gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*) melalui jalur koneksi *internet*. Berdasarkan uraian diatas, penulis menyimpulkan bahwa *web* adalah Sebuah *software* yang berfungsi untuk menampilkan dokumen -dokumen pada suatu *web* yang membuat pengguna dapat mengakses internet melalui *software* yang terkoneksi dengan internet. (Mara Destiningrum : 2017).

II.2.4. *Google Maps API*

Google Maps API adalah suatu library yang berbentuk JavaScript, dan dapat ditampilkan melalui web dengan HTML serta koneksi internet yang stabil. Dengan menggunakan Google Maps API, maka akan menghemat waktu dan biaya untuk membangun aplikasi peta digital, sehingga hanya fokus pada data yang ditampilkan. Google Maps API sangat mendukung developer dalam pengerjaan aplikasi yang membutuhkan peta digital, seperti foursquare atau path. Beberapa fitur dari Google Maps API ini seperti marker sebuah tempat, garis penghubung antar tempat satu dengan tempat lain, polygon, dan lain-lain. (Fitri Maharani,Dkk : 2015*)*.

II.2.5. Sistem Koordinat

Lintang dan bujur yang ditampilkan pada *Google Maps* merupakan dasar dari sistem koordinat geografis yang digunakan untuk menunjukan suatu titik di bumi. Garis lintang yaitu garis vertikal yang mengukur sudut antara suatu titik dengan garis katulistiwa. Titik di utara katulistiwa dinamakan lintang utara, sedangkan titik di selatan khatulistiwa dinamakan lintang selatan. Garis bujur yaitu garis horizontal yang mengukur sudut antara suatu titik dengan titik nol bumi, yaitu Greenwich di London Britania Raya yang mrupakan titik bujur 0^o atau 360^o yang diterima secara internasional. (Yulianto : 2018).

II.2.6. Angkatan kerja

Angkatan kerja adalah penduduk yang sudah memasuki usia kerja. Baik yang sudah bekerja maupun belum bekerja atau sedang mencari pekerjaan.

Menurut ketentuan pemerintah indonesia, penduduk yang sudah memasuki usia kerja adalah berusia minimal 15 tahun sampai 65 tahun. Akan tetapi tidak semua penduduk yang memasuki usia kerja termasuk angkatan kerja. Sebab penduduk yang tidak akif dalam kegiatan ekonomi tidak termasuk dalam kelompok angkatan kerja. Misalnya ibu rumah tangga, pelajar, mahasiswa dsb. (Asri Nur Hidayah Dkk : 2017)

II.2.7. PHP *(Hypertext Prepocessor)* dan HTML *(Hypertext Murkup Langunge)*

PHP merupakan bahasa pemograman yang di fungsikan untuk membuat suatu *website* yang dinamis. PHP dan HTML merupakan kode yang menjadi satu kesatuan, HTML digunakan sebagai pembangun atau pondasi dari kerangka *layout web,* sedangkan PHP di fungsikan sebagai proses pembangun *web*. (Satriaji Cahyo Nugroho : 2015).

II.2.8. Pengertian *Xampp*

XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, Perl. *XAMPP* adalah tool yang menyediakan paket perangkat lunak dalam satu buah paket. *XAMPP* adalah paket PHP yang berbasis Open Source yang dikembangkan oleh sebuah komunitas Open Source. Dengan menggunakan *XAMPP* tidak perlu lagi bingung untuk melakukan penginstalan program-program yang lain, karena semua kebutuhan telah disediakan oleh *XAMPP*. (Sylvia Tri Yuliani : 2016*)*.

(Sumber : Sylvia Tri Yuliani : 2016)

II.2.9. Metode *Haversine*

Formula Haversine adalah persamaan penting dalam sistem navigasi, nantinya Formula Haversine akan menghasilkan jarak terpendek antara dua titik, misalnya pada bola yang diambil dari garis bujur (longtitude) dan garis lintang (latitude). Formula ini pertama kali ditemukan oleh Jamez Andrew di tahun 1805, dan digunakan pertama kali oleh Josef de Mendoza y Ríos di tahun 1801. Istilah haversine ini sendiri diciptakan pada tahun 1835 oleh Prof. James Inman. Josef de Mendoza y Ríos menggunakan haversine pertama kali dalam penelitiannya tentang "Masalah Utama Astronomi Nautical", Proc. Royal Soc, Dec 22. 1796. Haversine digunakan untuk menemukan jarak antar bintang.

Formula Haversine adalah persamaan yang digunakan dalam navigasi, yang memberikan jarak lingkaran besar antara dua titik pada permukaan bola (bumi) berdasarkan bujur dan lintang. Formula Haversine merupakan suatu metode untuk

mengetahui jarak antar dua titik dengan memperhitungkan bahwa bumi bukanlah sebuah bidang datar namun adalah sebuah bidang yang memilki derajat kelengkungan. (Yulianto, dkk : 2018).

Penggunaan rumus ini mengasumsikan pengabaian efek ellipsoidal, cukup akurat untuk sebagian besar perhitungan, juga pengabaian ketinggian bukit dan kedalaman lembah di permukaan bumi. Berikut adalah rumus haversine:

$\Delta lat = lat2 - lat1$ .....(1)

$\Delta long = long2 - long1$ .....(2)

$a = \sin2 (\Delta lat/2) + \cos(lat1).\cos(lat2).\sin2 (\Delta long/2)$ .....(3)

$c = 2\text{atan2}(\sqrt{a},\sqrt{1-a})$ .....(4)

$d = R.c$ .....(5)

Di mana :

R = jari-jari bumi sebesar 6371(km)

Δlat = besaran perubahan latitude(Bujur)

$\Delta long$ = besaran perubahan longitude(Lintang)

C = kalkulasi perpotongan sumbu

d = jarak (km)

1 derajat = 0.0174532925 radian

II.2.10. *MySQL*

MySQL (My Structured Query Language) adalah Suatu sistem basis data relation atau Relational Database managemnt System (RDBMS) yang mampu bekerja secara cepat dan mudah digunakan MySQL juga merupakan program pengakses database yang bersifat jaringan, sehingga sapat digunakan untuk aplikasi multi user (banyak pengguna). *MySQL* didistribusikan gratis dibawah lisensi *GPL (General Public License).* Dimana setiap program bebas menggunakan *MySQL* namun tidak bisa dijadikan produk turunan yang dijadikan *closed source* atau komersial. (Mara Destiningrum : 2015).

II.2.11. *UML* (*Unified Modelling Language*)

Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

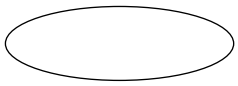
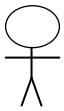
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. (Windu Gatta : 2013).

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasiskan UML adalah sebagai berikut:

1. *Use case* Diagram

Use case diagram merupakan pemodelan untuk kelakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram, yaitu:

Tabel II.1. Simbol *Use Case*

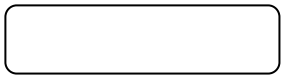
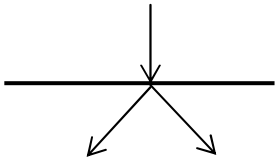
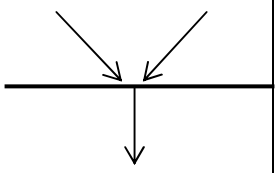
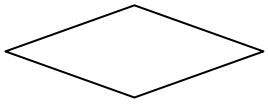
Gambar	Keterangan
	*Use case* menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama *use case*.
	Aktor adalah *abstraction* dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasikan aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan *use case*, tetapi tidak memiliki control terhadap *use case*.
	Asosiasi antara aktor dan *use case*, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan *use case* yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	Include, merupakan di dalam *use case* lain (*required*) atau pemanggilan *use case* oleh *use case* lain, contohnya adalah pemanggilan sebuah fungsi program.
	Extend, merupakan perluasan dari *use case* lain jika kondisi atau syarat terpenuhi.

(Sumber : Windu Gatta : 4-6)

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram,* yaitu:

Tabel II.2. Simbol *Activity Diagram*

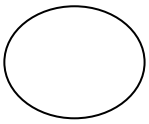
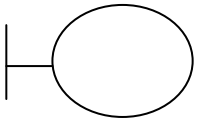
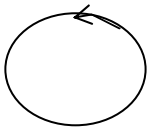
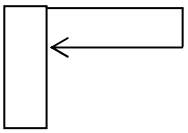
Gambar	Keterangan
	Start point, diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	End point, akhir aktifitas.
	*Activites*, menggambarkan suatu proses/kegiatan bisnis
	*Fork* (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	*Join* (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	*Decision Points,* menggambarkan pilihan untuk pengambilan keputusan, *true, false.*
	Swimlane, pembagian *activity* diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Windu Gatta : 6)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu:

Tabel II.3. Simbol *Sequence Diagram*

Gambar	Keterangan
	*EntityClass,* merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	*Boundary Class,* berisi kumpulan kelas yang menjadi *interface* atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan *form* cetak.
	*Control class,* suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	Message, simbol mengirim pesan antar *class*.
	*Recursive,* menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri
	Activation, activation mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	Lifeline, garis titik-titik yang terhubung dengan objek, sepanjang *lifeline* terdapat *activation*.

(Sumber : Windu Gatta : 7)

4. *Class Diagram* (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggng jawab entitas yang menentukan perilaku sistem.

Class diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), *Visibility,* tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

Tabel II.4. *Multiplicity Class Diagram*

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Windu Gatta : 8-9)