

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu / Kajian Pustaka

Penelitian mengenai Sistem Informasi Geografis sebelumnya sudah banyak dilakukan dengan berbagai macam studi kasus dan metode pengembangan sistem. Pada penelitian yang akan dilakukan, telah dilakukan studi literatur terhadap beberapa jurnal terkait. Tabel 2.1 menunjukkan penelitian terkait SIG beserta metode pengembangan sistem dan metode pengujian sistem.

Tabel 2.1 Tabel Penelitian Terdahulu

Judul, Penulis, dan Tahun Penelitian	Metode Pengembangan Sistem	Objek Penelitian	Tools Untuk Menampilkan Peta
Sistem Informasi Geografis Pemetaan Lokasi Praktek Kerja Lapangan Mahasiswa Teknik Informatika Universitas Mulawarman, Nashiruddin Kafi, Nataniel Dengen, Edy Budiman (2019)	<i>Waterfall</i>	<i>WebGIS</i> pemetaan lokasi PKL	<i>Google Maps</i>
Implementasi Sistem Informasi Geografis (SIG) Pada Penyebaran lokasi Kuliah Kerja Nyata (KKN), Alfi Andryan Suryadi, Husni Mubarak, Rohmat Gunawan, (2018)	<i>Prototype</i>	<i>WebGIS</i> pemetaan lokasi KKN	<i>Google Maps</i>
Aplikasi Sistem Informasi Geografis Pencarian Puskesmas Di Kabupaten Lampung Timur, Dedi Darwis, A. Ferico Octaviansyah, Heni Sulistiani, Yeron Roosyan Putra, (2020)	<i>Prototype</i>	SIG berbasis <i>mobile</i> pencarian puskesmas	<i>Google Maps</i>
Sistem Informasi Geografis Tingkat Kerusakan Ruas Jalan Berbasis Web, Maya Selvia Lauryn, Muhammad Ibrohim, (2019)	<i>Waterfall</i>	<i>WebGIS</i> pemetaan tingkat kerusakan ruas jalan	ArcView
Sistem Informasi Geografis Potensi Sumber Daya Kelautan Berbasis Android, Pratyaksa Ocsa Nugraha Saian, Raka Arya Pratama, Yerymia Alfa Susetyo, (2021)	<i>Agile</i>	SIG berbasis <i>mobile</i> pemetaan potensi sumber daya kelautan	<i>Google Maps</i>

Penelitian [5] bertujuan untuk membuat sebuah SIG sebagai media untuk mendapatkan detail informasi lokasi baik itu perusahaan, organisasi, maupun industri. Informasi tersebut digunakan untuk memetakan lokasi mana saja yang diambil oleh mahasiswa Teknik Informatika Universitas Mulawarman sebagai lokasi Praktek Kerja Lapangan. Metodologi yang digunakan yaitu model

Waterfall yang terdiri dari tahap analisis, perancangan, pemrograman, pengujian, operasi dan perawatan. Pada tahap perancangan peneliti membuat UML (*Unified Modelling Language*), ERD (*Entity Relationship Diagram*) dan desain *interface*. Pada tahap implementasi, peneliti menggunakan Bahasa pemrograman PHP dan *framework Codeigniter*. Untuk menampilkan peta lokasi, peneliti memanfaatkan *Google Maps*. Hasil dari rancangan ini sistem mampu mengelola data PKL terutama titik lokasi praktek yang terintegrasi dengan *Google Maps* dan dapat memberikan informasi mengenai lokasi PKL yang telah dilaksanakan maupun yang sedang berlangsung [5]. Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu pada metode pengembangan sistem dan *tools* yang digunakan untuk menampilkan peta, pada penelitian yang dilakukan menggunakan metode *agile* dan menggunakan *LeafletJS* untuk menampilkan peta sedangkan pada penelitian tersebut menggunakan metode *waterfall* dan menggunakan *Google Maps* untuk menampilkan peta.

Penelitian [6] bertujuan untuk merancang sistem informasi geografis yang dapat memetakan lokasi KKN. Metode pengembangan sistem dilakukan dengan model *prototype* yang terdiri 5 tahapan yaitu *communication*, *quick plan*, *modelling quick design*, *contruction of prototype*, dan *deployment delivery and feedback*. Pemodelan datanya menggunakan *Entity Relationship Diagram* (ERD). Untuk menampilkan peta memanfaatkan *google map*. Hasilnya sistem dapat menyediakan informasi pemetaan lokasi KKN sebanyak 91 desa yang dapat diakses oleh 281 DPL dan 2178 mahasiswa peserta KKN, serta dapat menampilkan lokasi rute menuju desa lokasi KKN beserta koordinat lokasi setiap desa [6]. Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu pada metode pengembangan sistem dan *tools* yang digunakan untuk menampilkan peta, pada penelitian yang dilakukan menggunakan metode *agile* dan menggunakan *LeafletJS* untuk menampilkan peta sedangkan pada penelitian tersebut menggunakan metode *prototype* dan menggunakan *Google Maps* untuk menampilkan peta.

Penelitian [7] bertujuan untuk membangun sebuah sistem informasi geografis lokasi puskesmas beserta fasilitas kesehatan di Kabupaten Lampung Timur. Metode pengembangan sistem yang digunakan pada penelitian ini yaitu menggunakan model dengan tahapan 5 tahapan antara lain, *communication, quick plan, modelling quick design, contruction of prototype*, dan *deployment delivery and feedback*. Perancangan sistem terdiri dari *use case diagram* dan rancangan basis data. Subjek uji coba dari hasil penelitian ini adalah tim ahli dengan pengujian fungsional menggunakan metode *black box testing*. Hasil penelitian menunjukkan bahwa aplikasi atau sistem yang dibangun dapat menampilkan lokasi Puskesmas dan direction menuju lokasi Puskesmas yang dituju oleh user yang dapat membantu user (masyarakat menemukan lokasi Puskesmas yang dituju) [7]. Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu pada metode pengembangan sistem dan *tools* yang digunakan untuk menampilkan peta, pada penelitian yang dilakukan menggunakan metode *agile* dan menggunakan *LeafletJS* untuk menampilkan peta sedangkan pada penelitian tersebut menggunakan metode *prototype* dan menggunakan *Google Maps* untuk menampilkan peta.

Penelitian [8] bertujuan untuk membuat sistem informasi geografis berbasis web yang dapat menyajikan informasi data spasial dalam bentuk peta digital mengenai tingkat kerusakan ruas jalan di Kabupaten Serang agar masyarakat mudah untuk memperoleh informasi tentang kondisi jalan tersebut. Sistem ini dirancang menggunakan model UML (*Unified Modelling Language*) dengan bahasa pemrograman PHP serta menggunakan *MySQL* sebagai *database*. Aplikasi SIG yang digunakan yaitu *ArcView*, *MapServer* dan *Pmapper*. Hasil penelitian menyimpulkan bahwa Sistem Informasi Geografis Tingkat Kerusakan Jalan Kabupaten Serang ini, diharapkan dapat bermanfaat bagi pemerintah khususnya Dinas Pekerjaan Umum Kabupaten Serang dalam penyajian Informasi kerusakan jalan [8]. Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu pada metode pengembangan sistem dan *tools* yang digunakan untuk menampilkan peta, pada penelitian yang dilakukan menggunakan metode *agile* dan menggunakan *LeafletJS* untuk menampilkan

peta, sedangkan penelitian sebelumnya menggunakan metode *waterfall* dan menggunakan *ArcView*, *MapServer* serta *Pmapper* untuk menampilkan peta.

Penelitian [9] bertujuan untuk membuat sistem informasi geografis untuk mengetahui potensi sumber daya kelautan di Yogyakarta. Penelitian ini menggunakan metode *agile* sebagai metode pengembangan sistem yang terdiri dari 4 tahapan yaitu *planning*, *design*, *implementation*, dan *analysis*. Pada tahap perancangan sistem terdiri dari *use case diagram* dan *flowchart system*. Pengujian sistem dilakukan yaitu menggunakan metode *black box testing* dan pengujian beta. Hasil pengujian menunjukkan 82,93% pengguna sudah puas dengan sistem ini dan menunjukkan hasil kategori “Sangat Bagus” [9]. Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu pada *tools* yang digunakan untuk menampilkan peta, pada penelitian yang dilakukan menggunakan *LeafletJS*, sedangkan pada penelitian tersebut menggunakan *Google Maps*.

Tabel 2.1 menunjukkan bahwa, dalam membangun sistem informasi terdapat berbagai metode pengembangan sistem. Masing-masing metode memiliki kelebihan dan kekurangan. Adapun perbedaan dari setiap metode pengembangan sistem berdasarkan kelebihan dan kekurangan yang dijelaskan pada Tabel 2.2.

Tabel 2.2 Kelebihan dan Kekurangan Metode Pengembangan Sistem [10].

Metode	Kelebihan	Kekurangan
<i>Waterfall</i>	<ol style="list-style-type: none"> 1. Mudah dalam pengelolaan karena hampir seluruh requirements telah diidentifikasi dan didokumentasikan 2. Tahapan yang berurutan secara linier, identifikasi dan dokumentasi yang lengkap, menyebabkan proses mudah dipahami. 	<ol style="list-style-type: none"> 1. Tahapan yang berurutan secara linier tidak memungkinkan untuk kembali pada tahapan selanjutnya 2. Tidak fleksibel terhadap perubahan kebutuhan yang terjadi dalam tahap pengembangan sistem 3. Hampir tidak ada toleransi kesalahan, terutama pada tahapan <i>planning</i> dan <i>design</i>
<i>Prototype</i>	<ol style="list-style-type: none"> 1. <i>User experience</i> yang meningkat, karena secara terus menerus melakukan uji coba dan evaluasi 2. Kesalahan dan redundansi dapat diminimalkan karena proses identifikasi yang baik 	<ol style="list-style-type: none"> 1. Setiap evaluasi membutuhkan penyesuaian, setiap penyesuaian akan meningkatkan kompleksitas sistem yang dikembangkan 2. Memberikan beban tambahan kepada <i>programmer</i>

Metode	Kelebihan	Kekurangan
<i>Agile</i>	<ol style="list-style-type: none"> 1. Metode ringan sesuai proyek ukuran kecil menengah 2. Menekankan produk akhir 3. Berulang 4. Pendekatan berbasis tes untuk persyaratan dan jaminan kualitas 	<ol style="list-style-type: none"> 1. Tidak cocok untuk menangani dependensi yang kompleks 2. Lebih risiko keberlanjutan, rawatan dan diperpanjang

Setelah membandingkan beberapa metode pengembangan sistem berdasarkan kelebihan dan kekurangan. Kemudian melakukan pemilihan metodologi pengembangan perangkat lunak yang tepat berdasarkan kriteria, menurut Dennis terdiri dari beberapa kriteria meliputi: kejelasan kebutuhan pengguna (*clarity user requirement*), penguasaan teknologi (*familiarity with technology*), tingkat kerumitan sistem (*system complexity*), tingkat kehandalan sistem (*system realibility*), waktu pelaksanaan (*short time schedules*), dan visibilitas jadwal pelaksanaan (*schedule visibility*) [10]. Hal ini dijelaskan pada Tabel 2.3.

Tabel 2.3 Kriteria Pemilihan Metode Pengembangan Sistem [10].

Kriteria Pengembangan Sistem	<i>Waterfall</i>	<i>Prototype</i>	<i>Agile</i>
Kejelasan Kebutuhan Pengguna	Buruk	Baik Sekali	Baik Sekali
Penguasaan Teknologi	Buruk	Buruk	Buruk
Tingkat Kerumitan Sistem	Baik	Buruk	Buruk
Tingkat Kehandalan Sistem	Baik	Buruk	Baik
Waktu Pelaksanaan	Buruk	Baik Sekali	Baik Sekali
Visibilitas Jadwal Pelaksanaan	Buruk	Baik Sekali	Baik

Berdasarkan Tabel 2.3 dari seluruh kriteria, metode *waterfall* memiliki 4 nilai buruk dari 6 kriteria, metode *prototype* memiliki 3 nilai buruk dari 6 kriteria, sedangkan metode *agile* hanya memiliki 2 nilai buruk dari 6 kriteria. Dengan demikian, jika dibandingkan dengan metode *waterfall* dan *prototype*, secara keseluruhan metode *agile* memiliki kriteria yang baik dalam melakukan pengembangan sistem. Sehingga dengan membandingkan kelebihan dan kekurangan dari setiap metode serta memperhatikan kriteria dalam pemilihan metode, penelitian ini dilakukan dengan memilih metode *agile*.

2.2. Dasar Teori

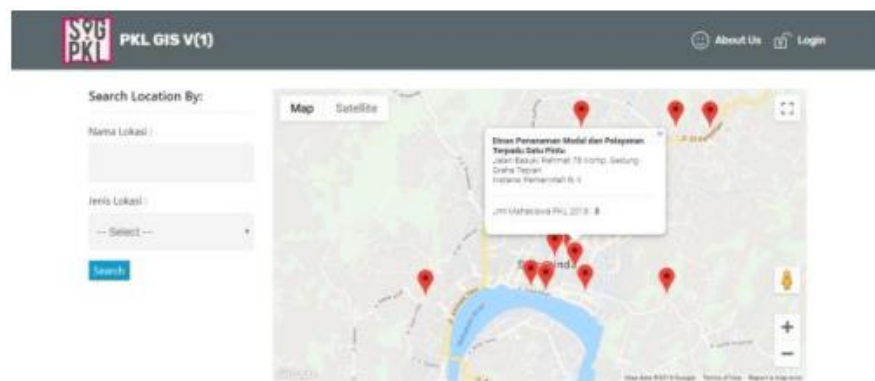
2.2.1. Praktik Kerja Lapangan (PKL)

Praktik Kerja Lapangan (PKL) merupakan implementasi dari pengetahuan dan keterampilan yang telah didapatkan selama masa

kuliah untuk berkontribusi membantu pemecahan masalah di berbagai perusahaan atau instansi yang terkait dengan bidang yang telah dipelajari pada masing-masing program studinya [1]. PKL pada umumnya wajib diikuti oleh para mahasiswa pada semester tertentu, bertujuan untuk melatih mahasiswa agar mengenal dan menghayati ruang lingkup pekerjaan di lapangan, guna mengadaptasi diri dengan lingkungan untuk melengkapi proses belajar yang didapat di bangku kuliah [1].

2.2.2. Sistem Informasi Geografis

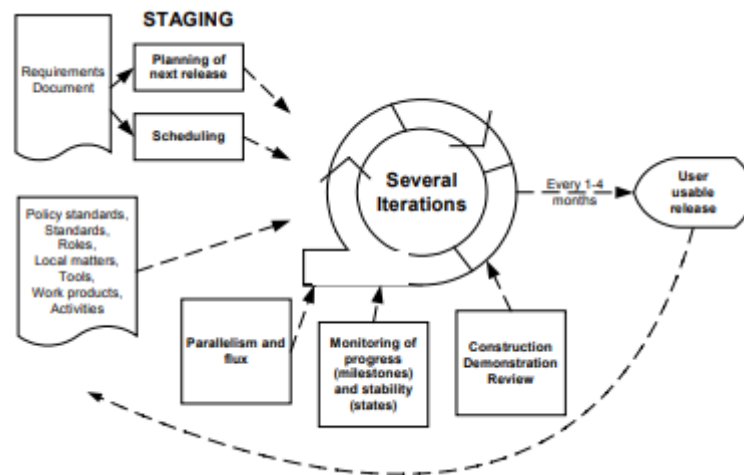
Sistem Informasi Geografis (SIG) dapat diartikan sebagai suatu sistem berbasis komputer yang berfungsi untuk mengumpulkan, menyimpan, mengolah, dan menganalisa, serta menyajikan data dan informasi dari suatu objek atau fenomena yang berkaitan dengan letak atau keberadaannya di permukaan bumi dalam bentuk peta digital atau informasi geografis [5]. Gambar 2.1 merupakan salah satu contoh penerapan SIG yaitu aplikasi untuk pemetaan lokasi PKL mahasiswa Teknik Informatika Universitas Mulawarman. Aplikasi ini berbasis *website* yang bertujuan untuk menyajikan informasi mengenai lokasi PKL, sehingga memberikan kemudahan bagi mahasiswa untuk mengetahui lokasi PKL dengan *view* tertentu. Pada sistem ini, setiap titik koordinat berisi informasi nama lokasi, alamat, jenis lokasi dan jumlah mahasiswa PKL [5].



Gambar 2.1 Contoh SIG [5].

2.2.3. Agile Development

Agile Software Development merupakan metodologi pengembangan perangkat lunak untuk jangka pendek dan memerlukan adaptasi cepat dari pengembang terhadap perubahan dalam bentuk apapun. Metode *agile* menerapkan konsep iterasi (perulangan). Sehingga sistem dapat menyesuaikan terhadap perubahan yang ada [4].



Gambar 2.2 Tahapan Metode Agile [4].

Gambar 2.2 menunjukkan tahapan-tahapan yang terdapat pada metode *agile*. Ada beberapa langkah dalam *Agile Development Methods*, yaitu :

1. Perencanaan, pada langkah ini pengembang membuat rencana tentang kebutuhan dari perangkat lunak yang akan dibuat. Pada perencanaan ini, pengembang melakukan rancangan sebagai perencanaan dalam pembuatan aplikasi, mulai dari rancangan basisdata sampai rancangan desain sistem.
2. Implementasi, bagian dari proses dimana *programmer* melakukan pengkodean perangkat lunak. Setelah perencanaan selesai, penulis melakukan implementasi atau pengkodean perangkat yang nantinya akan digunakan untuk membuat sistem.

3. Tes perangkat lunak, disini perangkat lunak yang telah dibuat dites oleh bagian control kualitas agar *bug* yang ditemukan bisa segera diperbaiki dan kualitas perangkat lunak terjaga.
4. Dokumentasi, setelah dilakukan tes perangkat lunak langkah selanjutnya yaitu proses dokumentasi perangkat lunak untuk mempermudah proses *maintenance* kedepannya.
5. *Deployment*, yaitu proses yang dilakukan oleh penjamin kualitas untuk menguji kualitas sistem. Setelah sistem memenuhi syarat maka perangkat lunak siap di *deployment*.



2.2.4. UML (*Unified Modelling Language*)


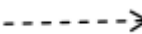
UML merupakan salah satu standar bahasa yang banyak digunakan di dunia industry. Digunakan untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [11]. Penelitian ini menggunakan 4 diagram UML diantaranya yaitu:

a. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* bekerja dengan mendeskripsikan tipikal interaksi antara user sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem itu dipakai [11]. Simbol-simbol yang digunakan dalam *use case diagram* dijelaskan pada Tabel 2.4.

Tabel 2.4 Simbol *Use Case* [12].





Nama	Simbol	Keterangan
<i>Use Case</i>		Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i> .
Aktor		Untuk mengidentifikasi 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.

Nama	Simbol	Keterangan
Asosiasi		Lintasan komunikasi antara actor dengan <i>use case</i> .
<i>Include</i>		Penambahan perilaku ke suatu <i>use case</i> dasar yang secara eksplisit mendeskripsikan penambahan tersebut.

b. Activity Diagram

Activity diagram merupakan diagram yang menggambarkan *workflow* atau aktivitas dari sebuah sistem yang ada pada perangkat lunak [11]. *Activity diagram* dibuat berdasarkan jumlah fungsionalitas yang terdapat pada *use case diagram*. Simbol-simbol yang digunakan dalam *activity diagram* dijelaskan pada Tabel 2.5.


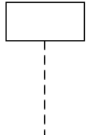
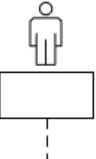



Tabel 2.5 Simbol-Simbol *Activity* [12].

Nama	Simbol	Keterangan
<i>Start Point</i>		Diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
<i>End Point</i>		Untuk mengakhiri aktifitas.
<i>Activities</i>		Menggambarkan suatu proses/kegiatan bisnis.
<i>Decision Point</i>		Menggambarkan pilihan untuk pengambilan keputusan <i>true</i> atau <i>false</i> .

c. Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Gambaran *sequence diagram* dibuat minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksinya pesan sudah dicakup pada *sequence diagram* sehingga semakin banyak *use case* yang didefinisikan, maka *sequence diagram* yang harus dibuat juga semakin banyak [11]. Simbol-simbol yang digunakan dalam *sequence diagram* dijelaskan pada Tabel 2.6.

Tabel 2.6 Simbol-Simbol *Sequence* [12].

Nama	Simbol	Keterangan
<i>Message</i>		Untuk mengirim pesan antar <i>class</i> .
<i>Object Lifene</i>		Merepresentasikan entitas tunggal dalam <i>sequence diagram</i> .
<i>Actor Lifeline</i>		Merepresentasikan entitas tunggal dalam <i>sequence diagram</i> di luar sistem dan berinteraksi dengan sistem.
<i>Recursive</i>		Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
<i>Activation</i>		Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi
<i>Lifeline</i>		Menghubungkan objek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifasinya).

d. *Class Diagram*

Class diagram merupakan gambaran struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* terdiri dari atribut dan operasi. Tujuannya untuk membuat hubungan antara dokumentasi perancangan dan perangkat lunak sesuai [11].

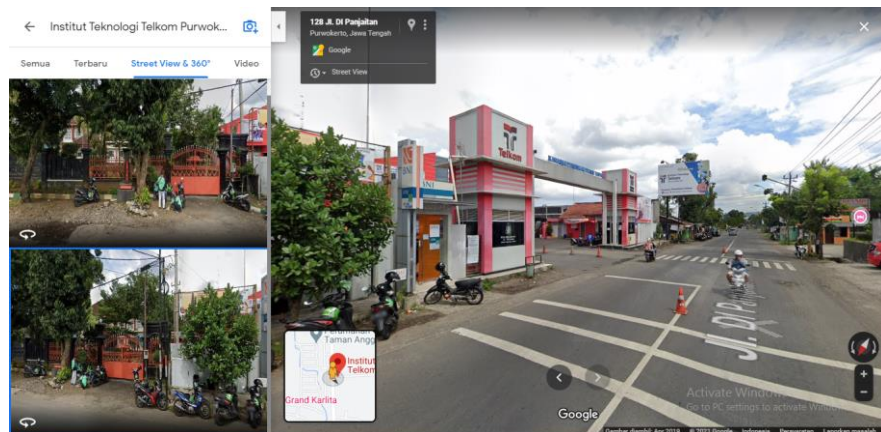
2.2.5. ERD (*Entity Relationship Diagram*)

Diagram Hubungan Entitas atau *Entity Relationship Diagram* (ERD) merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antara penyimpan. ERD digunakan dalam membangun basis data untuk menggambarkan relasi atau hubungan dari dua *file* atau dua tabel. Komponen utama ERD yaitu Entitas dan Relasi. Entitas merupakan sekumpulan objek yang dapat diidentifikasi secara unik atau saling berbeda, biasanya digambarkan dengan simbol persegi panjang. Relasi merupakan

hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda [13].

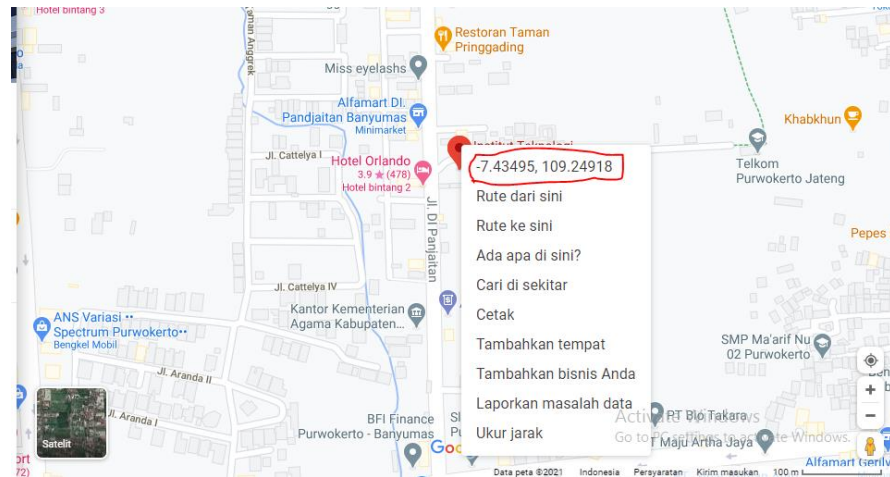
2.2.6. Google Maps

Google Maps adalah layanan gratis yang diberikan oleh *google* dan sangat *popular*. *Google maps* adalah suatu peta dunia yang digunakan untuk melihat suatu daerah. Dengan kata lain *google maps* merupakan suatu peta yang dapat dilihat dengan menggunakan suatu browser. Sebagai aplikasi navigasi, *Google Maps* memiliki banyak fitur beberapa diantaranya yaitu menyimpan alamat rumah dan kantor, membagikan lokasi, melihat *street view*, eksplorasi tempat baru serta fitur *download* untuk menyimpan peta suatu lokasi. Gambar 2.3 menunjukkan fitur *street view* yang terdapat pada *Google Maps* [14].



Gambar 2.3 *Google Maps Street View*

Google Maps dapat menampilkan titik koordinat suatu lokasi dengan melakukan klik kanan pada marker, lalu muncul titik koordinat yang berupa data *latitude* dan *longitude* seperti pada Gambar 2.4. Titik koordinat adalah titik pertemuan antara kedua garis lintang dan garis bujur. Garis lintang tersebut dengan *latitude* dan garis bujur disebut *longitude*. Garis lintang adalah garis dari atas ke bawah (*vertical*) yang menghubungkan kutub utara dengan kutub selatan bumi, sedangkan garis bujur adalah garis mendatar (*horizontal*) yang sejajar dengan garis khatulistiwa [15].



Gambar 2.4 Titik Koordinat Pada *Google Maps*

2.2.7. LeafletJS

LeafletJS adalah *library JavaScript open-source* yang memiliki banyak fitur pemetaan yang akan dibutuhkan oleh *developer*. Leaflet adalah alternatif baru pembuatan peta *online* yang ringan, *open source* dan kompatibel baik pada *desktop*, *web*, maupun *mobile*. Leaflet sangat mudah digunakan dan APInya didokumentasikan dengan sangat baik. Library leaflet JavaScript memiliki fitur yang lengkap baik pada library intinya maupun pada komunitasnya yang menyediakan plugin-plugin yang dapat menunjang Leaflet [16].

2.2.8. Codeigniter

Codeigniter adalah *framework* dengan ukuran kecil, waktu eksekusi yang minimal dan dapat diandalkan. Menurut survei, *Codeigniter* adalah *framework* yang paling mudah untuk dipelajari. Strukturnya sederhana, dokumentasinya lengkap dan jelas. Hal ini membuat *Programmer* pemula (*fresh graduate*) yang tidak terbiasa menggunakan *framework*, mampu dengan mudah menguasainya. *Codeigniter* menggunakan konsep MVC (*Model-View-Controller*) sehingga program menjadi lebih rapi, tidak bercampur antara *logic* dan *presentation* (*spaghetti code*). Sehingga program dapat di-*maintain* dengan mudah [17].

Pada penelitian ini, menggunakan *framework codeigniter 3*. CI3 memiliki *query builder* yang disediakan untuk memudahkan pemrogram untuk melakukan manipulasi *database*, membantu pemrogram untuk tidak perlu lagi menyusun perintah SQL untuk melakukan manipulasi *database*. Pembentukan perintah SQL untuk *query* dan manipulasi *database* telah ditangani oleh CI3. Pemrogram bisa dengan mudah mendefinisikan parameter nama tabel, nama kolom dan kriteria *query* yang diinginkan. [18].

2.2.9. MySQL

MySQL adalah sebuah *database management system* (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. MySQL dikembangkan sekitar tahun 1994 oleh sebuah perusahaan pengembang perangkat lunak dan konsultan basis data bernama MySQL AB yang berada di Swedia. MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus - kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. Ada 3 macam bentuk SQL antara lain *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Data Control Language* (DCL) [19].

1. *Data Definition Language* (DDL)

Struktur basis data yang menggambarkan skema basis data secara keseluruhan dan didesain dengan bahasa khusus yang disebut *Data Definition Language* (DDL). Bahasa tersebut dapat digunakan untuk membuat table baru, membuat indeks, mengubah table, menentukan struktur penyimpanan table, dan sebagainya. *Syntax* yang termasuk dalam DDL yaitu: *create*, *alter*, dan *drop*. Ketiga perintah tersebut dijelaskan pada tabel 2.7 [19].

Tabel 2.7 Tabel Perintah DDL [19].

<i>Create</i>	Digunakan untuk membuat tabel dan <i>database</i> .
<i>Drop</i>	Digunakan untuk menghapus <i>database</i> dan tabel.
<i>Alter</i>	Digunakan untuk mengubah struktur tabel yang sudah ada.

2. *Data Manipulation Language (DML)*

DML (*Data Manipulation Language*) adalah kelompok perintah yang berfungsi untuk memanipulasi data dalam basis data. Misalnya untuk pengambilan, penyisipan, perubahan dan penghapusan data. Perintah yang termasuk dalam kategori DML adalah *insert*, *delete*, *update*. Perintah tersebut dijelaskan pada tabel 2.8 [19].

Tabel 2.8 Tabel Perintah DML [19]

<i>Insert</i>	Digunakan untuk memasukkan data ke dalam tabel pada <i>database</i> .
<i>Update</i>	Digunakan untuk mengubah data yang ada di dalam tabel pada <i>database</i> ..
<i>Delete</i>	Digunakan untuk menghapus data di dalam tabel pada <i>database</i> ..

3. *Data Control Language (DCL)*

DCL digunakan untuk mengatur hak akses masing-masing *user*. DCL dapat memberikan hak akses *database*, mendefinisikan *space*, mengalokasikan *space*, dan melakukan audit penggunaan *database*. Ada beberapa perintah DCL yang dijelaskan pada tabel 2.9 [19].

Tabel 2.9 Tabel Perintah DCL [19]

<i>Grant</i>	Digunakan untuk memberikan izin kepada <i>user</i> untuk mengakses <i>database</i> .
<i>Revoke</i>	Digunakan untuk membatalkan izin <i>user</i> untuk mengakses <i>database</i> .
<i>Commit</i>	Digunakan untuk menetapkan penyimpanan pada <i>database</i> .
<i>Roolback</i>	Digunakan untuk membatalkan penyimpanan pada <i>database</i> .

2.2.10.PHP

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu Bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web*. PHP merupakan Bahasa *scripting server-side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, *server*lah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan. Adapun pengertian lain PHP adalah akronim dari Hypertext Preprocessor, yaitu suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data mengirimkannya kembali ke *web browser* menjadi kode HTML [20].