

LAPORAN PROYEK PERANGKAT LUNAK (PPL)

**Perancangan Sistem Informasi Dan Visualisai Data Penyebaran Covid 19
Berbasis Web menggunakan PHP Framework(CODEIGNETER)**

INFORMATIKA UNIVERSITAS BENGKULU



Disusun oleh

- | | | |
|----|--------------------------------|--------------------|
| 1. | UTAMA KHALID LUBIS | (G1A016017) |
| 2. | M. PRATAMA HIDAYATULLAH | (G1A018074) |
| 3. | MEYZAN AL YUTRA | (G1A018086) |

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS BENGKULU
2020**

KATA PENGANTAR

Puji syukur kami haturkan kepada Allah swt, yang telah memberikan rahmat serta hidayah-Nya, sehingga penulis dapat menyelesaikan laporan Proyek Perangkat Lunak yang berjudul **“Sistem Informasi Penyebaran Covid 19 Berbasis Web** “.

Dalam menyusun laporan ini, tentu tak lepas dari peran dan bantuan beberapa pihak. Oleh karena itu, kami ucapkan terima kasih yang tulus dan sebesar

besarnya kepada:

1. Ibu Ruvita. selaku dosen pembimbing
2. Ibu Arie Vatesia, S.T., M.T.I., Ph.D. selaku Ketua Program Studi Informatika Universitas Bengkulu.

Laporan ini dibuat untuk memenuhi mata kuliah Proyek Perangkat Lunak (PPL), penulis berharap agar laporan ini dapat bermanfaat dan menjadi refrensi dalam menambah wawasan dan ilmu pengetahuan bagi semua orang yang membaca.

Penulis sangat mengharapkan kritik dan saran yang sifat nya membangun agar sistem ini bisa dikembangkan lebih baik lagi. Kami juga berharap agar laporan ini dapat berguna bagi yang membaca, dan sistem yang kami bangun dapat benar-benar membantu dalam mengawasi perkembangan covid dan bagaimana Langkah penanganan yang dapat diambil. Atas perhatian pembaca, penulis mengucapkan terima kasih.

Bengkulu, 15 Oktober 2020

Penulis

DAFTAR ISI

KATA PENGANTAR	I
DAFTAR ISI.....	II
DAFTAR GAMBAR	IV
DAFTAR TABEL.....	V
BAB I <u>P</u> ENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	2
1.5. Manfaat.....	2
1.6. Metodologi Pengumpulan Data.....	3
1.7. Sistematika Penulisan.....	4
BAB II <u>L</u> ANDASAN TEORI	6
2.1 Sistem Informasi.....	6
2.2 Pemetaan	6
2.3 <i>API</i>	6
2.4 Mysql.....	6
2.5 <i>Web Server</i>	7
2.6 <i>Flowchart</i>	7
2.7 <i>Unified Modeling Language (UML)</i>	8
2.8 <i>Database</i>	14
2.9 Konsep Basis Data.....	15
2.10 <i>PHP</i>	16
2.11 <i>HTML</i>	17
2.12 <i>Bootstrap</i>	17
2.13 <i>Java</i>	18
BAB III <u>A</u> NALISIS DAN PERANCANGAN	19
3.1 Analisa Permasalahan.....	19

3.2	Analisa Kebutuhan	19
3.2.1	Kebutuhan Fungsional	19
3.2.2	Kebutuhan Non-Fungsional.....	20
3.3	Metode Pengembangan Sistem	21
3.4	Deskripsi Sistem.....	22
3.5	Perancangan Sistem.....	22
3.5.1	Use Case Diagram	23
3.5.2	<i>Activity Diagram</i>	24
3.5.3	<i>Sequence Diagram</i>	26
3.5.4	<i>Class Diagram</i>	27
3.6	Struktur <i>Database</i>	28
3.7	Desain <i>Interface</i>	29
3.7.1	Dashboard Masyarakat	29
BAB IV HASIL DAN PEMBAHASAN		32
4.1.	Implementasi Sistem Dan Uji Coba Sistem	32
4.1.1.	Dashboard Masyarakat	32
BAB V KESIMPULAN DAN SARAN.....		37
5.1.	Kesimpulan.....	37
5.2.	Saran	37
DAFTAR PUSTAKA		38

DAFTAR GAMBAR

Gambar 1.1 API DataSet Covid 19	3
Gambar 1.2 API Pemetaan	4
Gambar 2.1 Diagram UML	8
Gambar 3.1 <i>Flowchart tahap pembuatan Sistem Informasi Covid 19</i>	22
Gambar 3.2 <i>Use case Diagram Pada</i>	23
Gambar 3.3 <i>Activity Diagram Masyarakat Pada Manajemen Data Covid</i>	24
Gambar 3.4 <i>Activity Diagram Masyarakat Manajemen Data Perkembangan</i> ...	25
Gambar 3.5 <i>Sequence Diagram Masyarakat</i>	26
Gambar 3.6 <i>Class Diagram Data Covid</i>	27
Gambar 3.7 Struktur Database	28
Gambar 3.8 <i>Desain Interface Dashboard Masyarakat</i>	29
Gambar 3.9 <i>Desain Interface Data Provinsi</i>	29
Gambar 3.10 <i>Desain Interface Pemetaan</i>	30
Gambar 3.11 <i>Desain Interface Data Kecamatan</i>	30
Gambar 3.12 <i>Desain Interface Pemetaan Kecamatan</i>	31
Gambar 3.13 <i>Desain Interface Perkembangan H/Bln</i>	31
Gambar 4.1 <i>Dashboard Masyarakat</i>	32
Gambar 4.2 <i>Data Provinsi pada Manajemen Data Covid Masyarakat</i>	32
Gambar 4.3 <i>Pemetaan pada Manajemen Data Covid Masyarakat</i>	33
Gambar 4.4 <i>Data Kecamatan pada Manajemen Data Covid Masyarakat</i>	33
Gambar 4.5 <i>Grafik Data Kecamatan Manajemen Data Covid Masyarakat</i>	34
Gambar 4.6 <i>Pemetaan Kecamatan pada Manajemen Data Covid Masyarakat</i> ..	34
Gambar 4.7 <i>Perbandingan Hari Manajemen Data Perbandingan Masyarakat</i>	35
Gambar 4.8 <i>Perbandingan Bln Manajemen Data Perbandingan Masyarakat</i>	35
Gambar 4.9 <i>Perbandingan Data Manajemen Data Perbandingan Masyarakat</i> ...	36

DAFTAR TABEL

Tabel 2.1 Keterangan Simbol <i>Use case Diagram</i>	9
Tabel 2.2 Keterangan Simbol <i>Class Diagram</i>	11
Tabel 2.3 Keterangan Simbol <i>Activity Diagram</i>	11
Tabel 2.4 Keterangan Simbol <i>Sequence Diagram</i>	12
Tabel 3.1 <i>Database</i> provinsi	28
Tabel 3.2 <i>Database</i> data_kecamatan	28

BAB I

PENDAHULUAN

1.1. Latar Belakang

Di Indonesia ada Lembaga Badan Nasional Penanggulangan Bencana (BNPB). Dimana Lembaga ini mengatasi bencana yang terjadi di Indonesia baik bencana dengan kondisi geografis, geologis, hidrologis, maupun Kesehatan. Dan menyediakan informasi kepada masyarakat agar memberikan informasi yang di ingin di ketahui. Terkhusus Di era globalisasi saat ini, ilmu teknologi sangatlah berkembang dengan pesat dan maju sehingga dapat memberikan kemudahan dalam berbagai aspek kehidupan seperti dalam penyebaran bencana. Sehingga pada saat pandemi seperti saat ini dimana orang-orang bertanya-tanya bagaimana perkempangan dari covid ini. Sedangkan mereka tidak memiliki akses sendiri untuk melihat data-data yang ada dan hanya mendengarkan apa yang diberitakan di tv atau media lain. Ini menyebabkan kurang puasnya masyarakat dalam menerima informasi yang ada. Sehingga peran BNPB di sini sungguh diperlukan.

Disebabkan Informasi yang kurang memuaskan, BNPB menjabarkan data covid yang ada supaya masyarakat bisa mendapatkan informasi yang dibutuhkan, oleh karena data yang di perikan berupa data yang banyak dan bahkan terlalu umum menimbulkan keliruan informasi meyebabkan masyarakat menjadi beranggapan bahwa covid ini adalah sesuatu yang tidak berbahaya sehingga mereka melanggar aturan yang telah di tetapkan pemerintah seperti PSBB.

Oleh karena itu, kami berupaya untuk memecahkan permasalahan tersebut dengan melakukan perancangan sistem informasi berbasis web pada *penyebaran covid*. Dengan adanya perancangan sistem ini data-data *penyebaran covid* dapat diolah dengan tepat dan dapat menghasilkan informasi yang bermanfaat sehingga masyarakat, dapat melihat dan memastikan peyebaran covid yang ada, beserta lokasi-lokasi yang terkena covid.

1.2. Rumusan Masalah

Dari latar belakang yang sudah dipaparkan diatas, bahwa rumusan masalah yang diangkat adalah :

1. Bagaimana merancang dan membangun *Sistem Informasi Penyebaran Covid 19 Berbasis Web* dalam pemetaan penyebaran daerah di indonesia.
2. Bagaimana menganalisa kasus covid setiap provinsi.
3. Bagaimana pengaruh PSBB dalam kecepatan penyebaran.

1.3. Batasan Masalah

Batasan masalah dalam merancang dan membangun sistem informasi ini adalah Pemetaan untuk penyebaran covid 19 setiap provinsi di Indonesia.

1.4. Tujuan

Tujuan dari pembuatan sistem informasi ini adalah :

1. Merancang sistem informasi berbasis web menggunakan PHP dan MySQL pada *penyebaran covid*.
2. Membangun sistem informasi berbasis web untuk membantu menampilkan dan pemetaan *penyebaran covid*.
3. Menampilkan kecepatan peningkatan kasus covid setiap provinsi per bulan
4. Menampilkan Analisa perbandingan kecepatan sehingga bisa di lihat apakah PSBB yang diterapkan berhasil.

1.5. Manfaat

Dengan adanya sistem informasi ini, diharapkan dapat memberikan solusi bagi pemerintah untuk mengambil keputusan yang tepat setelah melihat penyebaran dan pengaruh PSBB yang di terapkan dalam penyebaran covid setiap provinsi di Indonesia. Sehingga;

1. Pemerintah bisa memutuskan apakah PSBB memang harus di terapkan atau tidak.

1.6. Metodologi Pengumpulan Data

Dalam penyusunan Laporan Proyek Perangkat Lunak ini untuk mengumpulkan data dengan teknik:

Metode Dataset

Metode Dataset adalah pencarian data dan informasi yang bersumber dari data yang telah ada di internet sebagai bahan acuan pembuatan laporan dan system. Dalam pembuatan system ini kami menggunakan data bentuk API yang telah di sediakan BNPB di situs resmi penanganan covid yaitu ; <https://bnpb-inacovid19.hub.arcgis.com>.

Dalam penggunaan data yang di sediakan sehingga menghasilkan informasi yang bisa di tampilkan, perlu untuk di connect kan terlebih dahulu. Sehingga perlu code pemanggilan sebagai berikut :

```
$Variabel = json_decode(file_get_contents('https://link API yang  
digunakan'), true);
```

Di sini Rest API yang kami gunakan untuk data covid adalah sebagai berikut :

```
$nasional = json_decode(file_get_contents('https://apicovid19indonesia-v2.vercel.app/api/indonesia/more'), true);  
$provinsi = json_decode(file_get_contents('https://services5.arcgis.com/VS6HdKS0VfIhv8Ct/arcgis/rest/services/COVID19_In  
donesia_per_Provinsi/FeatureServer/0/query?where=1%3D1&outFi  
elds=*&outSR=4326&f=json'), true);  
$update = json_decode(file_get_contents('https://data.covid19.go.id/public/api/update.json'), true);
```

Gambar 1.1 API DataSet Covid 19

Sesuai Gambar 1.1 API yang digunakan pada system ini untuk menampilkan data yang ada, sehingga menghasilkan informasi. Dimana variabel \$nasional untuk menampilkan data kasus kejadian covid 19 di indonesia secara nasional, \$provinsi digunakan untuk menampilkan data kasus covid masing-masing tiap provinsi dan penyebarannya. Dan \$update untuk menampilkan data perkembangan covid 19 dari awal kasus hingga sekarang.

Sedangkan, untuk pemetaan digunakan API berikut :

```
var mymap = L.map('mapid').setView([0.502464, 118.841877], 5);
L.tileLayer('https://api.mapbox.com/styles/v1/{id}/tiles/{z}/{x}
/{y}?access_token=pk.eyJ1IjoibWFwYm94IiwiaWUiYSI6ImNpejY4NXVycTA2
emYycXBndHRqcmZ3N3gifQ.rJcFIG214AriISLbB6B5aw', {
  maxZoom: 18,
  attribution: 'Map data &copy; <a href="https://www.openstreetm
ap.org/">OpenStreetMap</a> contributors, ' +
  '<a href="https://creativecommons.org/licenses/by-sa/2.0/">CC-
BY-SA</a>, ' +
  'Imagery &copy; <a href="https://www.mapbox.com/">Mapbox</a>',
  id: 'mapbox/streets-v11',
  tileSize: 512,
  zoomOffset: -1
}).addTo(mymap);
```

Gambar 2.2 API Pemetaan

Pada Gambar 1.2, merupakan API untuk pembuatan map pada pemetaan yang kami gunakan pada system untuk menampilkan informasi penyebaran di Indonesia dengan menampilkan peta map sederhana.

1.7. Sistematika Penulisan

Penulisan laporan Kegiatan Proyek Perangkat Lunak ini disusun dalam 5 bab, dengan tujuan untuk mempermudah pembacaan yang lebih akurat, dan garis besarnya adalah:

BAB I. PENDAHULUAN

Memuat latar belakang yang menyebabkan munculnya persoalan sehingga dibutuhkan sistem informasi, selain itu dijelaskan rumusan masalah, batasan masalah, tujuan, manfaat sistem dan metode penelitian.

BAB II. LANDASAN TEORI

Dalam Bab ini membahas kajian-kajian teoritis sebagai dasar analisa dan perancangan system.

BAB III. ANALISA DAN PERANCANGAN SISTEM

Bab ini membahas tentang analisis sistem yang akan dibuat dan juga perancangan Sistem.

BAB IV. HASIL DAN PEMBAHASAN

Bab ini berisi tentang implementasi dari sistem informasi, cara pengoperasian serta output yang dihasilkan.

BAB V. PENUTUP

Memuat beberapa kesimpulan yang dapat diambil dari hasil perancangan dan pembangunan Sistem Informasi yang dibuat.

BAB II

LANDASAN TEORI

2.1 Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan (Pender, 2002).

2.2 Pemetaan

Pemetaan adalah suatu proses menyajikan informasi muka Bumi yang berupa fakta, dunia nyata, baik bentuk permukaan buminya maupun sumberdaya alamnya, berdasarkan skala peta, sistem proyeksi peta, serta simbol-simbol dari unsur muka Bumi yang disajikan.

2.3 API

API adalah singkatan dari *Application Programming Interface*, dan memungkinkan *developer* untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan. *API* terdiri dari berbagai elemen seperti *function*, *protocols*, dan *tools* lainnya yang memungkinkan *developers* untuk membuat aplikasi.

2.4 MySQL

MySQL merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* atau *DBMS multithread, multi-user*, dengan sekitar enam juta instalasi di seluruh dunia. Setiap orang bebas menggunakan *MySQL*, namun tidak boleh di jadikan produk turunan salah satu konsep utama dalam *Database* sejak lama yaitu *Structured Query Language SQL* (Ramadhaniet. al., 2013).

2.5 Web Server

Web Server adalah sebuah perangkat lunak server yang berfungsi menerima permintaan *HTTP* atau *HTTPS* dari klien yang di kenal dengan browser *web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman yang umumnya berbentuk dokumen *HTML* .

2.6 Flowchart

Flowchart adalah adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program.

Dalam perancangan *flowchart* sebenarnya tidak ada rumus atau patokan yang bersifat mutlak (pasti). Hal ini didasari oleh *flowchart* (bagan alir) adalah sebuah gambaran dari hasil pemikiran dalam menganalisa suatu permasalahan dalam komputer. setiap perancangan *flowchart* selalu terdiri dari tiga bagian, yaitu input, proses dan output.

- *System Flowchart*

System flowchart dapat didefinisikan sebagai bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan-urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan di sistem.

- *Document Flowchart*

Bagan alir dokumen (*document flowchart*) atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya.

- *Schematic Flowchart*

Bagan alir skematik (*schematic flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem.

- *Program Flowchart*

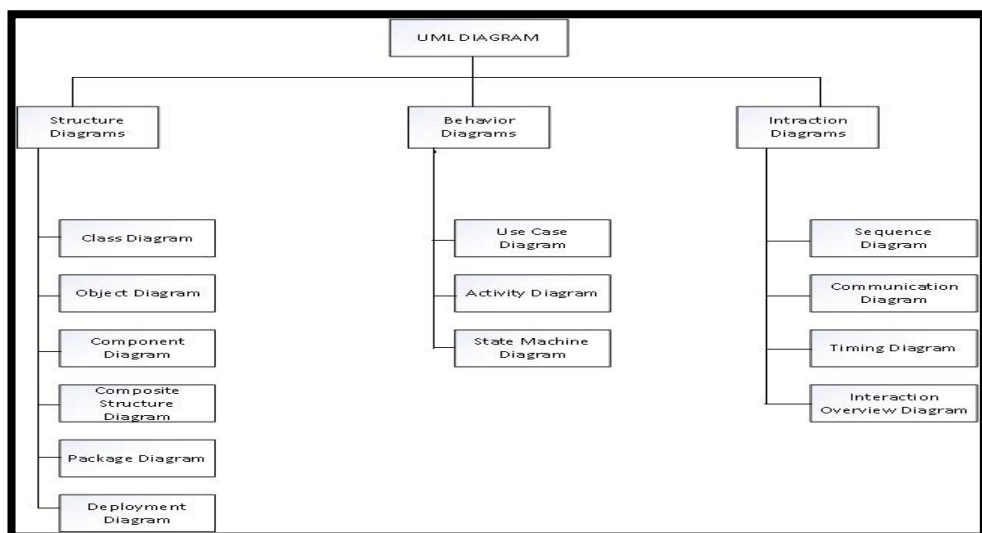
Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derivikasi bagan alir sistem.

- *Process Flowchart*

Bagan alir proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri. Bagan alir ini juga berguna bagi analisis sistem untuk menggambarkan proses dalam suatu prosedur.

2.7 Unified Modeling Language (UML)

UML terdiri atas banyak elemen-elemen grafis yang digabungkan membentuk diagram. Tujuan representasi elemen-elemen grafis ke dalam diagram adalah untuk menyajikan beragam sudut pandang dari sebuah sistem berdasarkan fungsi masing-masing diagram tersebut. Kumpulan dari beragam sudut pandang inilah yang kita sebut sebuah model. Dengan menggunakan UML dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa - bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. (Munwar, 2006)



Gambar 2.1 Diagram UML (A.S. & M. Shalahuddin, 2018)

Berikut ini penjelasan singkat dari pembagian kategori tersebut :

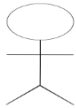

- 1) *Structure diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
- 2) *Behavior diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
- 3) *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem (A.S. & M. Shalahuddin, 2018).


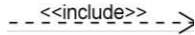



Dari tiga belas diagram tersebut, ada empat diagram yang sering digunakan. Pada dasarnya penggunaan UML tersebut digunakan sesuai dengan kebutuhan masing– masing. Berikut ini adalah penjelasan 13 UML yang ada

a. Use case diagram

Salah satu kontributor terhadap diagram *use case* dalam UML adalah Ivar Jacobsen. *Use case* menggambarkan *external view* dari sistem yang akan kita buat modelnya. Model *use case* dapat dijabarkan dalam diagram *use case*, tetapi yang perlu diingat, diagram tidak identik dengan model karena model lebih luas dari diagram.

Tabel 2.1 Keterangan Simbol *Use case Diagram* (A.S. & M. Shalahuddin, 2018)

SIMBOL	DESKRIPSI
<i>Actor</i> 	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>usecase</i> .
<i>Depedency</i> 	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung pada elemen yang tidak mandiri (<i>dependent</i>).
<i>Generalization</i>	Hubungan dimana objek anak

	(<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
<i>Include</i> 	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
<i>Extend</i> 	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
<i>Association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
<i>Usecase</i> 	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .

b. Class diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* memiliki tiga area pokok yaitu :





- Nama (dan *stereotype*)
- Atribut
- Metode

Atribut dan metode memiliki salah satu dari sifat berikut:

- 1) *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan

- 2) *Protected*, hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya
- 3) *Public*, dapat dipanggil oleh siapa saja

Tabel 2.2 Keterangan Simbol *Class Diagram* (A.S. & M. Shalahuddin, 2018)


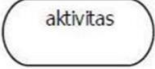



SIMBOL	DESKRIPSI
<p><i>Generalization</i></p> 	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
<p><i>Association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
<p><i>Class</i></p> 	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
<p><i>Dependency</i></p> 	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) juga akan mempengaruhi elemen yang tidak bergantung padanya

c. *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Tabel 2.3 Keterangan Simbol *Activity Diagram* (A.S. & M. Shalahuddin, 2018)


Simbol	Deskripsi
	Status awal aktivitas sitem, sebuah diagram aktivitas memiliki sebuah




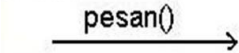

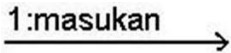
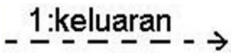

status awal 	diagram aktivitas memiliki sebuah status awal.
aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
percabangan / decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
penggabungan / join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

d. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

Tabel 2.4 Keterangan Simbol *Sequence Diagram* (A.S. & M. Shalahuddin, 2018)

SIMBOL	DESKRIPSI
	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang dibuat itu sendiri, jadi walaupun symbol dari <i>actor</i> adalah gambar orang, tapi <i>actor</i> belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.

<p>objek</p> 	Menyatakan objek yang berinteraksi pesan
<p>Garis hidup objek</p> 	Menandakan kehidupan objek selama urutan diakhiri tanda X pada titik dimana kelas tidak lagi berinteraksi.
<p>Objek sedang aktif berinteraksi</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya
<p>pesan</p> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek lain.
<p>Pesan typesend</p> 	Menyatakan bahwa suatu objek mengirimkan masukan ke objek lainnya arah panah mengarah pada objek tertentu, arah panah mengarah pada objek yang dikirim.
<p>Pesan tipereturn</p> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah. Mengarah pada objek yang menerima kembalian
<p>Pesan tipedestroy</p> 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

2.8 Database

Database atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur, dan juga batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data, dan juga *update* yang rumit.

Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*Database management system*). DBMS merupakan sistem perangkat lunak yang memungkinkan *user* untuk memelihara, mengontrol, dan mengakses data secara praktis dan efisien. Dengan kata lain semua akses ke basis data akan ditangani oleh DBMS. Ada beberapa fungsi yang harus ditangani DBMS yaitu mengolah pendefinisian data, dapat menangani permintaan pemakai untuk mengakses data, memeriksa sekuriti dan integriti data yang didefinisikan oleh DBA (*Database Masyarakatistrator*), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun disk, dan menangani unjuk kerja semua fungsi secara efisien.

Tujuan utama dari DBMS adalah untuk memberikan tinjauan abstrak data kepada *user* (pengguna). Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara, dan tetap dapat

diambil (akses) secara efisien. Pertimbangan efisien di sini adalah bagaimana merancang struktur data yang kompleks tetapi masih tetap bisa digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya.

2.9 Konsep Basis Data

1. Pemahaman Mengenai Tabel

Pada suatu basis data, tabel direpresentasikan menjadi suatu bentuk segiempat berupa matriks yang terdiri dari kolom dan baris. Dalam suatu basis data, pengertian kolom memiliki analogi dengan nama *field* yang pembacaan datanya dilakukan secara vertical. Sedangkan *record* pembacaan datanya secara horizontal.

- a. Kolom → *field*
- b. Baris → *record*

2. Pemahaman mengenai field

Field dalam suatu tabel dapat dikategorikan menjadi *field key* dan *field non key*. Pada *Primary key* dan *Foreign Key* memiliki sifat yang unik, nilai dan jenis *field* tidak memiliki kesamaan nilai dengan *field* yang lainnya. Sehingga *field* tersebut berfungsi sebagai patokan dalam melakukan manipulasi datanya.

3. Pemahaman mengenai record

Satu baris pada sebuah tabel merupakan data yang dimiliki oleh satu record. Nilai-nilai yang dimiliki oleh sebuah record merupakan gabungan dari semua field yang terdapat dalam tabel.

4. Penerapan Konsep Kardinalitas Dalam Merelasikan Tabel

Kardinalitas merupakan batasan dari banyaknya hubungan, yang dapat dilakukan oleh suatu himpunan entitas dalam melakukan relasi dengan himpunan entitas lainnya.

Kardinalitas terdiri dari 4 macam, yaitu :

- a) Satu ke Satu
- b) Satu ke Banyak
- c) Banyak ke Satu
- d) Banyak ke banyak

2.10 PHP

PHP merupakan aplikasi perangkat lunak *opensource*, di mana kepanjangan dari PHP adalah *Hypertext Preprocessor* yang diatur dalam aturan *general purpose licences* (GPL). Kemampuan dan fitur PHP yang paling mendukung banyak basis data yaitu MSSQL, MySQL, *Oracle*, dan *Postgre SQL*. Secara teknologi, bahasa pemrograman PHP sangat mirip dengan bahasa pemrograman yang berbasis *web* lain, contohnya bahasa ASP (*Active Server Page*), *Cold Fusion*, JSP (*Java Server Page*), ataupun *Perl*.
Invalid source specified.

PHP mendukung delapan jenis tipe data yang dibagi atas tiga bagian, yaitu :

- a. Tipe data dasar
 - *Boolean*
 - *Integer*
 - *Float*
 - *String*
- b. Tipe *compound* / tipe campuran
 - *Array*
 - *Object*
- c. Tipe *special* / tipe khusus
 - *Resource*
 - *Null* **Invalid source specified.**

Variabel adalah tempat yang digunakan untuk menyimpan data. Untuk bisa menggunakan suatu variable di PHP, harus diawali dengan tanda \$. Penamaan variabel memiliki beberapa aturan, antara lain :

- a) Nama variabel tidak boleh ada *spasi*, dan jika ingin dipisah harus diberi tanda *underscore* (*_*).
- b) Nama variabel tidak boleh diawali dengan angka, namun diperbolehkan menggunakan tanda *underscore* (*_*) diawali nama variabel.
- c) Di dalam penulisan nama variabel, tidak boleh ada operator aritmatika.

- d) Tidak boleh mengandung karakter-karakter khusus seperti : %, @, #, ;, !
- e) Aturan – aturan diatas harus diterapkan kalau kita membuat variabel. Inisialisasi variabel adalah melakukan pengisian nilai pada saat variabel tersebut dibuat. **Invalid source specified.**

2.11 HTML

HTML kependekan dari *Hyper Text Markup Language*. Dokumen HTML adalah *file* teks murni yang dapat dibuat dengan *editor* teks sembarang. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*.

HTML merupakan suatu *script* dimana kita bisa menampilkan informasi dan daya kreasi kita lewat internet. HTML tak memiliki batasan teks.

HTML adalah Bahasa pemrograman yang fleksibel di mana kita bisa meletakkan skrip dari bahasa pemrograman lain seperti *JAVA*, *Visual Basic*, *C* dan lain-lain. Jika HTML tersebut tidak dapat mendukung suatu perintah pemrograman tertentu. *Browser* tidak akan menampilkan kotak dialog "*Syntax Error*" jika terdapat penulisan kode yang keliru pada skrip HTML sepanjang kode-kode yang kita tuliskan merupakan kode-kode HTML tanpa penambahan kode-kode dari luar seperti *java*. Oleh karena itu, jika terjadi *syntax error* pada skrip HTML, efek yang paling jelas adalah HTML tersebut tak akan ditampilkan pada halaman jendela *browser*.

2.12 Bootstrap

Bootstrap adalah *front-end framework* yang bagus dan luar biasa yang mengedapankan tampilan untuk *mobile device* (*Handphone*, *smartphone* dll.) guna mempercepat dan mempermudah pengembangan *website*. *Bootstrap* menyediakan *HTML*, *CSS* dan *Javascript* siap pakai dan mudah untuk dikembangkan.

Bootstrap framework untuk membangun desain *web* secara responsif dan cepat. Artinya, tampilan *web* yang dibuat oleh *Bootstrap*

akan menyesuaikan ukuran layar dari *browser* yang kita gunakan baik di desktop, tablet ataupun *mobile device*. Sehingga, *user* akan mendapatkan pengalaman yang lebih baik dalam berselancar tanpa mempertimbangkan perangkat apa yang harus digunakan.

Sejatinya, apabila kita menggunakan *Bootstrap*, kita tinggal menggunakan nama class (untuk css) dan library (javascript) yang sudah ditentukan oleh *Bootstrap* tanpa perlu menulis kode dari 0 (awal) sehingga, bisa menghemat waktu dalam pengembangan *website* untuk urusan UI (*User Interface*). *Bootstrap* memiliki tampilan yang indah dan dapat di *customisasi*.

2.13 JAVA

Java adalah bahasa pemrograman tingkat tinggi yang berorientasi objek dan program *java* tersusun dari bagian yang disebut kelas. Kelas terdiri atas metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para pemrogram *Java* banyak mengambil keuntungan dari kumpulan kelas di pustaka kelas *Java*, yang disebut dengan *Java Application Programming Interface* (API). Kelas-kelas ini diorganisasikan menjadi sekelompok yang disebut paket (*package*). *Java API* telah menyediakan fungsionalitas yang memadai untuk menciptakan *applet* dan aplikasi canggih. Jadi ada dua hal yang harus dipelajari dalam *Java*, yaitu mempelajari bahasa *Java* dan bagaimana mempergunakan kelas pada *Java API*. Kelas merupakan satu-satunya cara menyatakan bagian eksekusi program, tidak ada cara lain. Pada *Java* program *javac* untuk mengkompilasi file kode sumber *Java* menjadi kelas-kelas *bytecode*. File kode sumber mempunyai ekstensi *.java. Kompilator *javac* menghasilkan file *bytecode* kelas dengan ekstensi *.class. Interpreter merupakan modul utama sistem *Java* yang digunakan aplikasi *Java* dan menjalankan program *bytecode* *Java*.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisa Permasalahan

Sistem Informasi Dan Visualisasi Data Penyebaran Covid 19 *website* ini dibuat serta dirancang dikarenakan melihat pandemic covid yang semakin bertambah dan masyarakat perlu informasi mengenai daerah yang terjangkit covid 19, karena hal itulah diangkat permasalahan yang ada.

3.2 Analisa kebutuhan

Sistem yang akan dibuat memerlukan kebutuhan fungsional dan kebutuhan nonfungsional. Tujuan analisis kebutuhan adalah sebagai batasan dari sistem yang akan dibuat, sesuai dengan kebutuhan *user* dan fasilitas - fasilitas yang merupakan nilai tambah yang ada pada sistem yang dibangun. Kebutuhan ini membantu membangun sistem .

3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan fasilitas yang dibutuhkan serta aktivitas apa saja yang dilakukan oleh sistem secara umum. Desain sistem yang akan dibuat memerlukan beberapa kebutuhan data masukan, kebutuhan data keluaran dan kebutuhan antar muka. Tujuan analisis kebutuhan adalah untuk menentukan spesifikasi fungsi, kemampuan serta fasilitas dari program. Analisis kebutuhan juga bermanfaat sebagai dasar evaluasi setelah program selesai disusun.

1. *Masyarakat Pada Web*

Fungsionalitas sistem untuk *Masyarakat* adalah sebagai berikut :

a. Fungsi *Melihat Data Covid 19*

Digunakan untuk melihat informasi data covid19 dalam bentuk tabel.

b. Fungsi *Melihat Penyebaran Covid 19*

Digunakan untuk melihat penyebaran informasi penyebaran covid 19 dalam bentuk map.

c. Fungsi Melihat *Claustering*

Digunakan untuk melihat daerah berdasarkan clustering dalam tampilan map.

3.2.2 Kebutuhan Non-Fungsional

1. Perangkat Lunak

Dalam membangun Sistem Informasi Dan Visualisasi Data Penyebaran Covid 19 berbasis Website ini, digunakan perangkat lunak antara lain :

- a) MySQL
- b) Git Bash
- c) Web Browser : Mozilla Firefox / Chrome
- d) Astah Community
- e) Microsoft Visual Studio
- f) API
- g) Bahasa pemrograman : PHP , Java
- h) Sistem Operasi Windows 10

2. Perangkat Keras

Perangkat keras yang dibutuhkan untuk menjalankan Sistem Informasi Dan Visualisasi Data Penyebaran Covid 19 berbasis Website, antara lain :

- a) Monitor minimal VGA atau SVGA
- b) Processor Intel Core i3
- c) RAM 4.00 GB
- d) Free space 500 MB
- e) Harddisk 32 GB
- f) Mouse
- g) Keyboard

3.3 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website adalah dengan menggunakan metode Model *Agile*. Model *Agile* adalah model pengembangan jangka pendek yang memerlukan adaptasi cepat dan pengembangan terhadap perubahan dalam bentuk apapun.

Ada beberapa tahapan dalam *Agile Development Methods*, yaitu :

1. Perencanaan, pada langkah ini pengembang dan klien membuat rencana tentang kebutuhan dari perangkat lunak yang akan dibuat.
2. Implementasi, bagian dari proses dimana programmer melakukan pengkodean perangkat lunak.
3. Tes perangkat lunak, disini perangkat lunak yang telah dibuat di tes oleh bagian kontrol 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 dideployment.
6. Pemeliharaan, langkah terakhir yaitu pemeliharaan. Tidak ada perangkat lunak yang 100% bebas dari bug, oleh karena itu sangatlah penting agar perangkat lunak dipelihara secara berkala.

3.4 Deskripsi Sistem

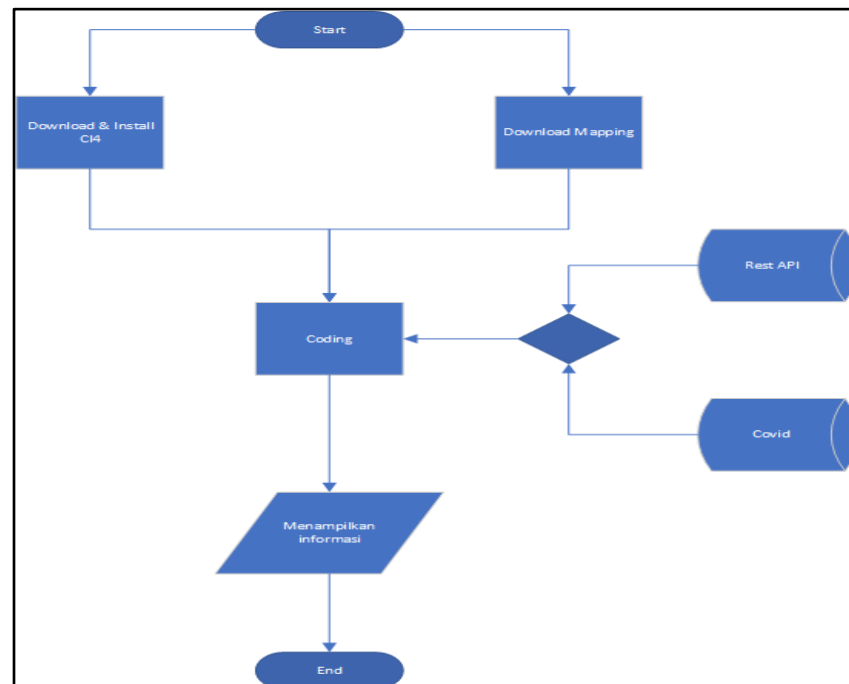
Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website dapat digambarkan sebagai bentuk fasilitas yang menyediakan kemudahan informasi dalam melihat penyebaran covid 19 di Indonesia.

Berdasarkan kebutuhan fungsional sistem dan non fungsional sistem, maka dapat diasumsikan Sistem Informasi ini adalah sebagai berikut :

1. Pihak yang berinteraksi dengan sistem informasi ini adalah *user* yang terdiri dari *masyarakat*.
2. *Masyarakat* dapat mengelola bisa melihat penyebaran covid 19 di Indonesia dalam tampilan map.

3.5 Perancangan Sistem

Perancangan sistem ini adalah tahap konseptualisasi, yaitu suatu tahap yang mengharuskan perancangan sistem (perangkat lunak) untuk berusaha tahu pasti mengenai hal-hal yang menjadi kebutuhan dan harapan pengguna sehingga nanti aplikasi yang dibuat memang dibutuhkan oleh *user* serta memuaskan kebutuhan dan harapannya. Yang dapat dilihat melalui flowchart berikut.

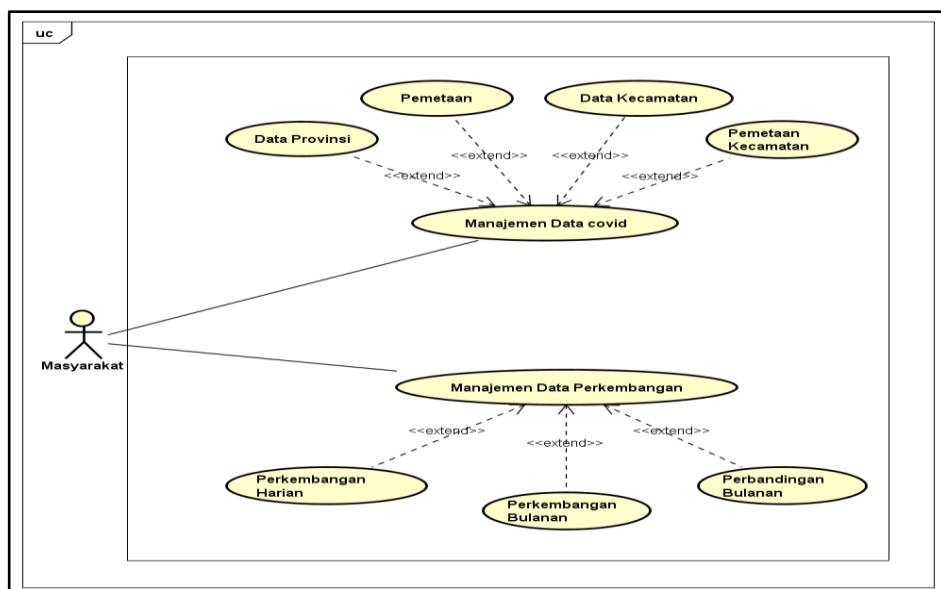


Gambar 3.1 Flowchart tahap pembuatan Sistem Informasi Covid 19

Tahap awal dalam pembuatan system ini adalah mendownload dan install CI4 beserta mapping, setelah itu di hubungkan menjadi satu dengan cara di coding, dimana pada proses coding ini kita bisa memilih database yang digunakan agar bisa dihubungkan ke codingan baik melalui rest API ataupun database covid. Sehingga bisa menampilkan informasi berupa data-data yang ingin ditampilkan, dan pembuatan system informasi covid 19 selesai. Adapun konseptualisasi dalam analisis sistem ini menggunakan konsep UML (*Unified Modeling Language*) yaitu salah satu *tool* atau model untuk merancang pengembangan *software* yang berbasis *object oriented*. Beberapa literatur menyebutkan bahwa UML menyediakan tiga belas jenis diagram, yang lain menyebutkan sembilan karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan dan diagram perwaktuan digabung menjadi diagram interaksi. Adapun diagram yang digunakan pada Sistem Informasi ini antara lain :

3.5.1 Use Case Diagram

Use case Sistem Informasi ini melibatkan sebuah aktor yang berinteraksi langsung dalam sistem dan melakukan manajemen langsung terhadap sistem. Berikut adalah gambar dari Diagram *UseCase* Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website:



Gambar 3.2 Usecase Diagram

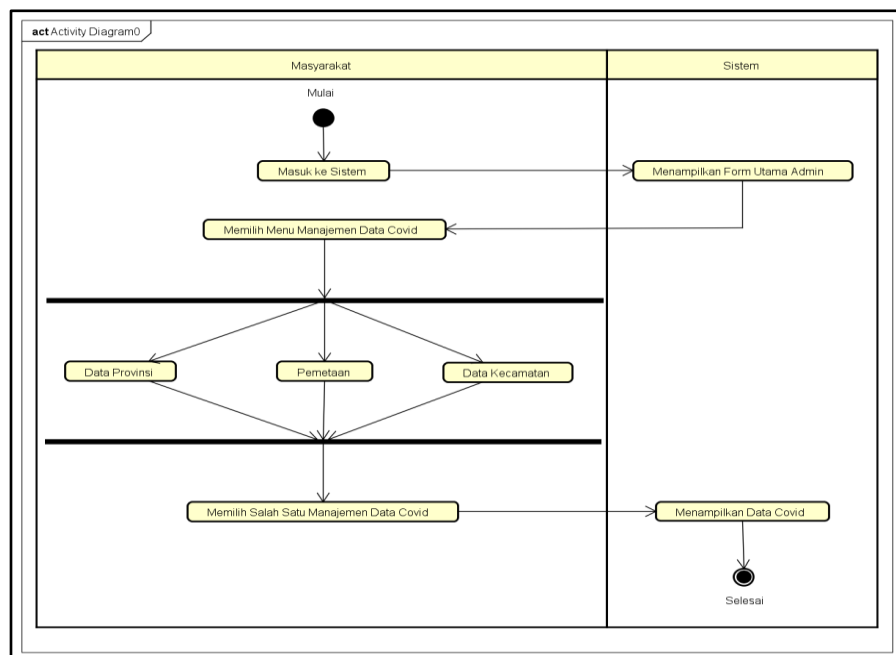
Pada Diagram Use case Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website ini terdapat sebuah aktor yaitu masyarakat, selain itu terdapat 2 manajemen yaitu manajemen data Covid, dan manajemen data Perkembangan, yang masing- masing manajemen tersebut memiliki fungsi yang berbeda beda. Dimana didalam manajemen data covid *masyarakat* dapat mengelola data provinsi, pemetaan, data kecamatan, dan pemetaan kecamatan. Sedangkan, pada manajemen data Perkembangan ini *masyarakat* dapat mengelola data perkembangan harian, perkembangan bulanan, dan perbandingan bulanan.

3.5.2 Activity Diagram

Pada Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website, *Activity Diagram* menjelaskan tentang proses berjalannya sistem dari berbagai operasi yang terdapat pada Sistem Informasi ini. *Activity Diagram* yang ada adalah *Activity Diagram Masyarakat* pada Manajemen Data Covid, dan Manajemen Data Harian.

Berikut diagramnya:

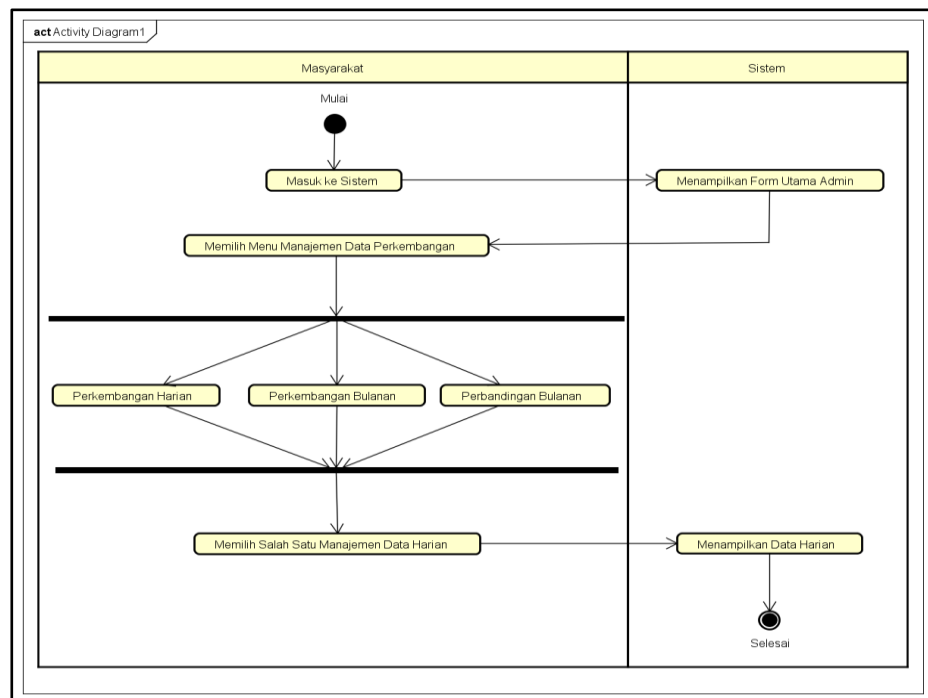
1) Manajemen Data Covid (*Masyarakat*)



Gambar 3.3 Activity Diagram *Masyarakat* Pada Manajemen Data Covid

Pada gambar 3.3 dapat dijelaskan sebagai berikut, setelah *masyarakat* membuka sistem maka sistem akan menampilkan halaman utama. Pada halaman utama, *masyarakat* masuk ke manajemen data covid dan bisa memilih untuk melihat data Provinsi, pemetaan, data Kecamatan, atau pemetaan Kecamatan. Kemudian setelah memilih aktifitas tersebut akan di proses di sistem dan sistem akan menampilkan data yang dipilih. Jika aktifitas telah selesai maka akan dikonfirmasi sistem dan *logout*/keluar dari sistem.

2) Manajemen Data Perkembangan (*Masyarakat*)



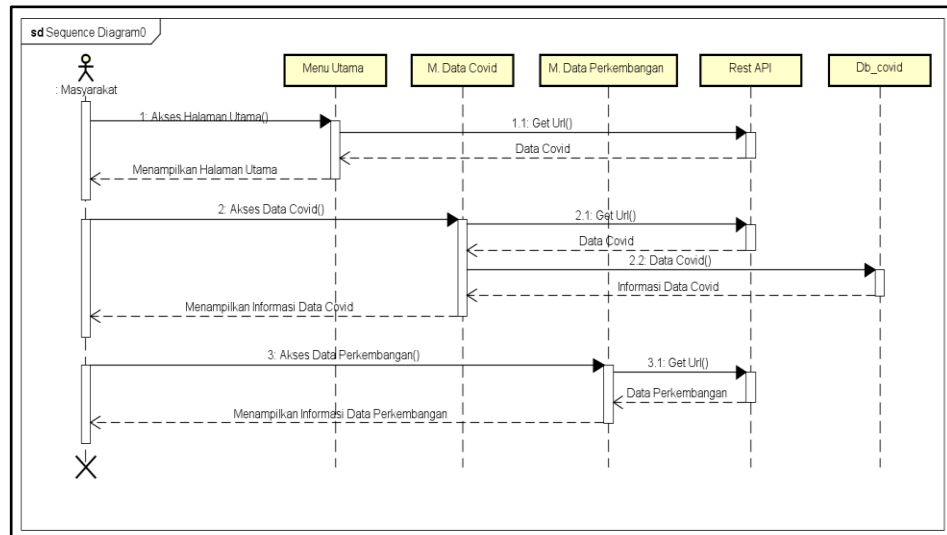
Gambar 3.4 Activity Diagram *Masyarakat* Manajemen Data Perkembangan

Pada gambar 3.4 dapat dijelaskan sebagai berikut, setelah *masyarakat* membuka sistem maka sistem akan menampilkan halaman utama. Pada halaman utama, *masyarakat* masuk ke manajemen data perkembangan dan bisa memilih untuk melihat data perkembangan harian, perkembangan bulanan, atau perbandingan bulanan. Kemudian setelah memilih aktifitas tersebut akan di proses di sistem dan sistem akan menampilkan data yang dipilih. Jika aktifitas telah selesai maka akan dikonfirmasi sistem dan *logout*/keluar dari sistem.

3.5.3 Sequence Diagram

Sequence diagram dari Sistem Informasi Dan Visualisasi Data Penyebaran Covid 19 berbasis Website. *Sequence* Diagram yang ada adalah *Sequence* Diagram untuk aktor *masyarakat*. Berikut merupakan Gambar *Sequence* Diagram aktor *masyarakat* dari sistem informasi ini:

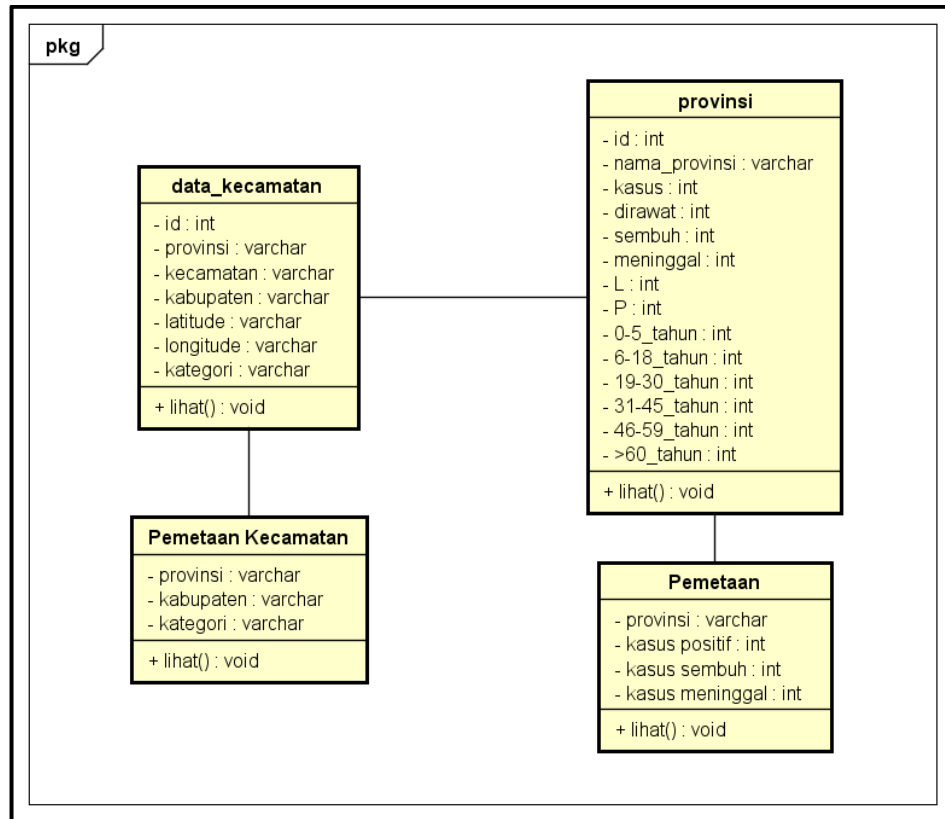
1. Masyarakat



Gambar 3.5 *Sequence* Diagram *Masyarakat*

Pada diagram *Sequence* yang diaktori oleh *masyarakat*, *masyarakat* akan mengakses halaman utama. Setelah itu *masyarakat* dapat memilih manajemen data covid, dan manajemen data perkembangan. Dimana *masyarakat* akan mengakses data covid berupa tampilan data. Data tersebut diambil melalui link *url* yang telah di *setting* dan juga akan mengambil melalui *database* covid dan akan menjadi informasi data covid. Sehingga *masyarakat* bisa melihat tampilan data yang ada. *Masyarakat* juga bisa akan mengakses data covid berupa tampilan data. Data tersebut diambil melalui link *url* yang telah di *setting* dan akan menjadi informasi data covid. Sehingga *masyarakat* bisa melihat tampilan data yang ada. Begitu pula untuk manajemen data perkembangan hanya saja tampilan data yang ada nanti nya akan muncul dalam bentuk perkembangan yang di ambil melalui link *url* yang telah di *setting*. Kemudian ketika semua aktivitas yang dilakukan *masyarakat* selesai maka *masyarakat* dapat mengakhiri sistem tersebut

3.5.4 Class Diagram



Gambar 3.6 Class Diagram Data Covid

Sistem ini terdiri dari 4 kelas, yaitu kelas *provinsi*, *data_kecamatan*, *pemetaan kecamatan*, dan *pemetaan*. Kelas *provinsi* memiliki atribut id, nama_provinsi, kasus, dirawat, sembuh, meninggal, L, P, 0-5_tahun, 6-18_tahun, 19-30_tahun, 31-45_tahun, 46-59_tahun, dan >60_tahun. *data_kecamatan* memiliki atribut seperti id, provinsi, kecamatan, kabupaten, lalitude, longitude, dan kategori. *Pemetaan Kecamatan* memiliki atribut provinsi, kabupaten, dan kategori. *Pemetaan* memiliki atribut provinsi, kasus positif, kasus sembuh, dan kasus meninggal.

Pada *class* diagram tersebut kelas *data_kecamatan* dapat berelasi ke kelas *provinsi* karena mengelola data dari kelas *provinsi*. Lalu pada kelas *pemetaan* berelasi ke kelas *provinsi* karena *pemetaan* dilakukan melalui data provinsi sedangkan, kelas *pemetaan kecamatan* berelasi ke kelas *data_kecamatan* karena *pemetaan* dilakukan melalui data kecamatan.

3.6 Struktur Database

Struktur *Database* yang terdapat pada Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website. Sebagai berikut:



Gambar 3.7 Struktur *Database*

3.6.1 Rancangan Tabel *Database*

Struktur *Database* yang terdapat Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website.

Tabel 3.1 Database provinsi

Nama Kolom	Tipe	Panjang Data	Keterangan
id	int	11	PK, AI
nama_provinsi	varchar	40	
kasus	int	8	
dirawat	int	8	
sembuh	int	8	
meninggal	int	8	
L	int	8	
P	int	8	
0-5_tahun	int	8	
6-18_tahun	int	8	
19-30_tahun	int	8	
31-45_tahun	int	8	
46-59_tahun	int	8	
≥60_tahun	int	8	

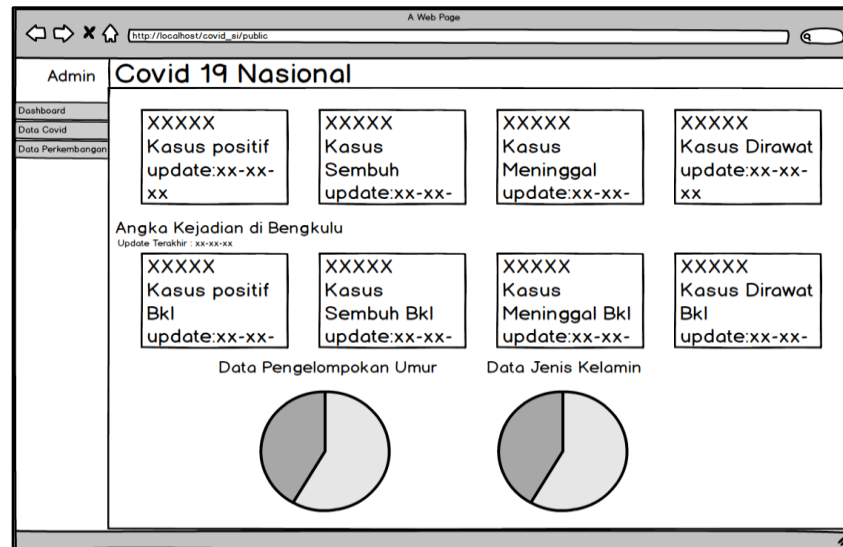
Tabel 3.2 Database data_kecamatan

Nama Kolom	Tipe	Panjang Data	Keterangan
id	int	11	PK, AI
kecamatan	varchar	50	
kabupaten	varchar	50	
provinsi	varchar	50	
latitude	varchar	50	
longitude	varchar	50	
kategori	varchar	50	

3.7 Desain *Interfacae*

Desain Antarmuka Pengguna (*User Interface Design*) atau rekayasa antarmuka pengguna adalah desain untuk komputer, aplikasi perangkat lunak, dan situs web yang berfokus pada pengalaman pengguna (*User Experience*) dan interaksi Halaman.

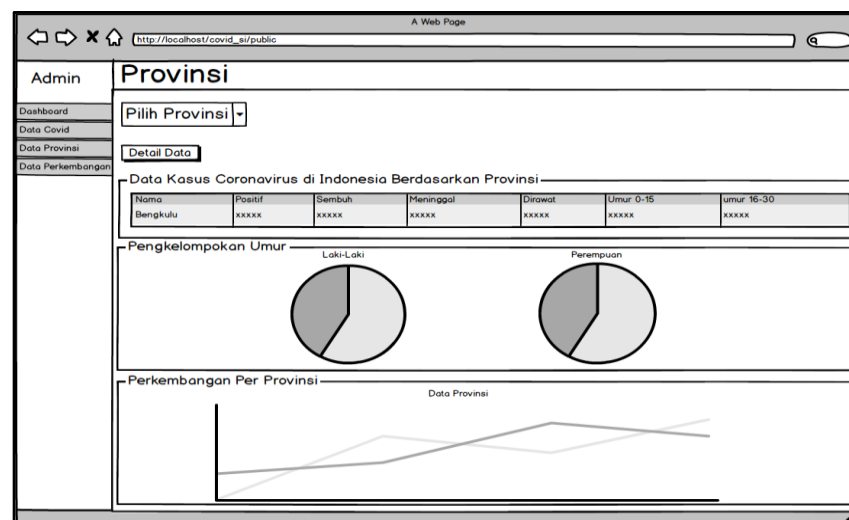
3.7.1 *Dashboard Masyarakat*



Gambar 3.8 *Desain Interface Dahsboard Masyarakat*

Pada gambar 3.8, merupakan tampilan dari *dashboard masyarakat*, *masyarakat* dapat melihat berbagai informasi covid terdiri dari 2 manajemen yaitu, data covid, dan data perkembangan.

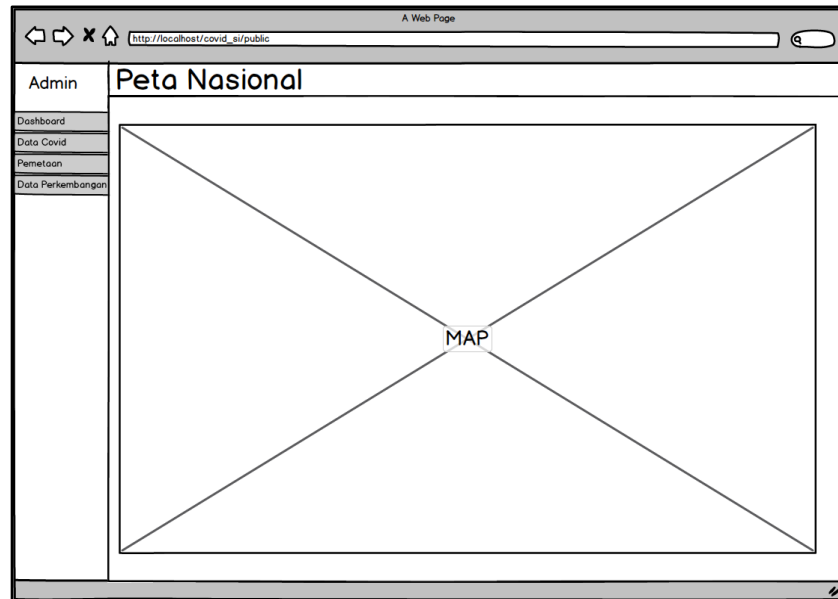
a. Data Provinsi pada Manajemen Data Covid



Gambar 3.9, *Desain Interface Data Provinsi*

Pada gambar 3.9, merupakan tampilan dari data Provinsi , masyarakat dapat melihat data provinsi yang telah terdampak Covid mulai dari nama provinsi, positifnya, sembuh, meninggal, dsb. Masyarakat juga dapat melihat perbandingan banyak nya umur yang terkena covid dan melihat grafik perkembangan per provinsi.

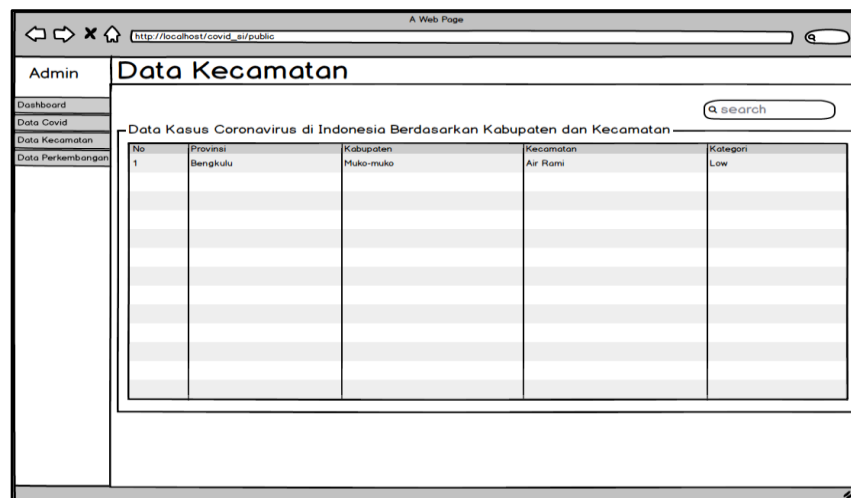
b. Pemetaan pada Manajemen Data Covid



Gambar 3.10 Desain *Interface Pemetaan*

pada gambar 3.10, pada halaman ini masyarakat bisa Melihat titik penyebaran Covid 19 berdasarkan provinsi di seluruh indonesia melalui map.

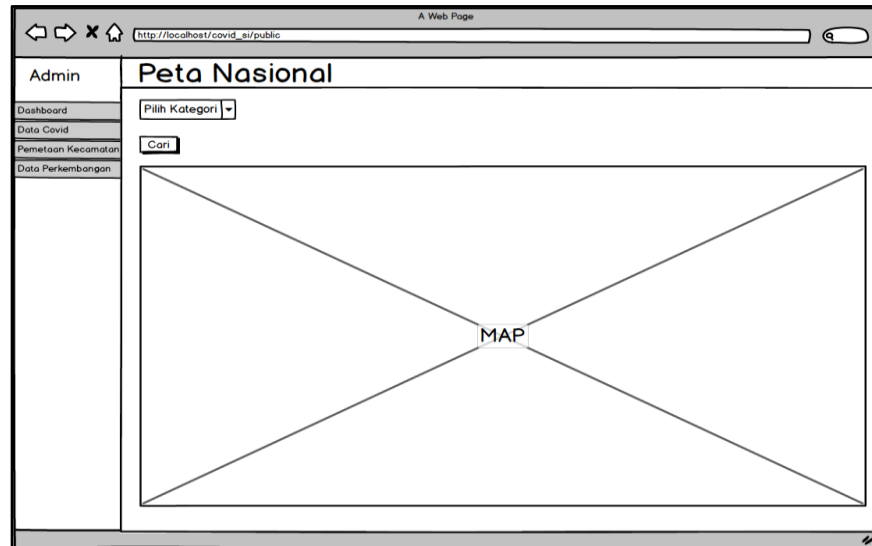
c. Data Kecamatan pada Manajemen Data Covid



Gambar 3.11 Desain *Interface Data Kecamatan*

Pada gambar 3.11, merupakan tampilan dari data Kecamatan, masyarakat dapat melihat data tiap-tiap kecamatan yang telah terdampak Covid mulai dari nama provinsi, kabupaten, kecamatan, dan kategori.

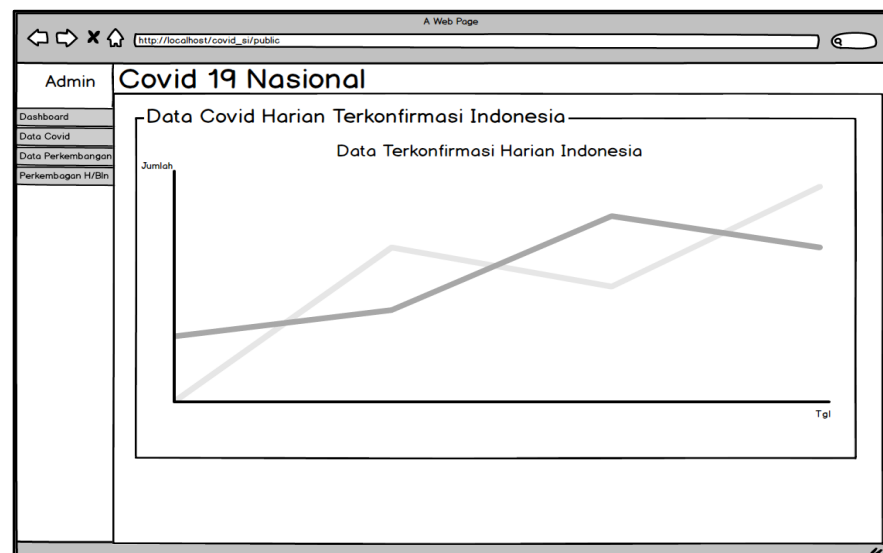
d. Pemetaan Kecamatan pada Manajemen Data Covid



Gambar 3.12 Desain *Interface* Pemetaan Kecamatan

Pada gambar 3.12, pada halaman ini masyarakat bisa Melihat titik penyebaran Covid 19 sesuai kategori penyebaran berdasarkan kecamatan di seluruh indonesia melalui map.

e. Perkembangan H/Bln Pada Manajemen Data Perkembangan



Gambar 3.13 Desain *Interface* Perkembangan H/Bln

Pada gambar 3.13, merupakan tampilan grafik Perkembangan Data Covid harian atau bulan berdasarkan jumlah kasus yang tercatat.

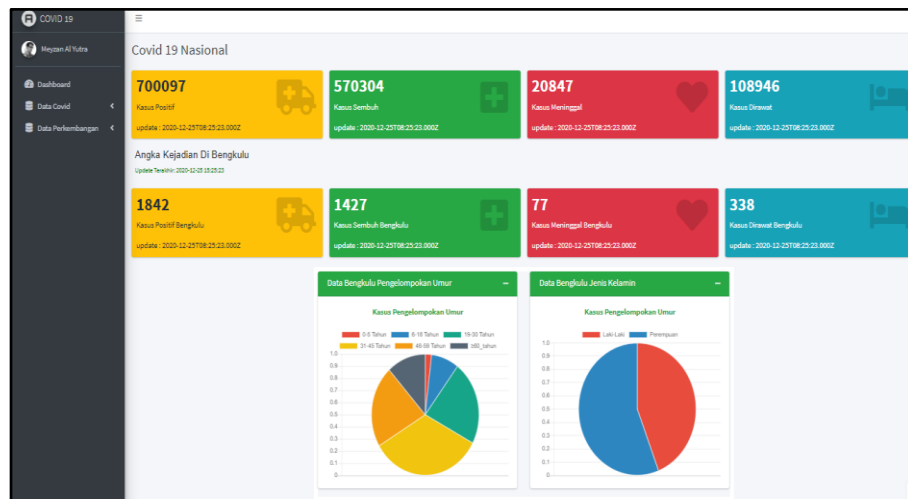
BAB IV

HASIL DAN PEMBAHASAN

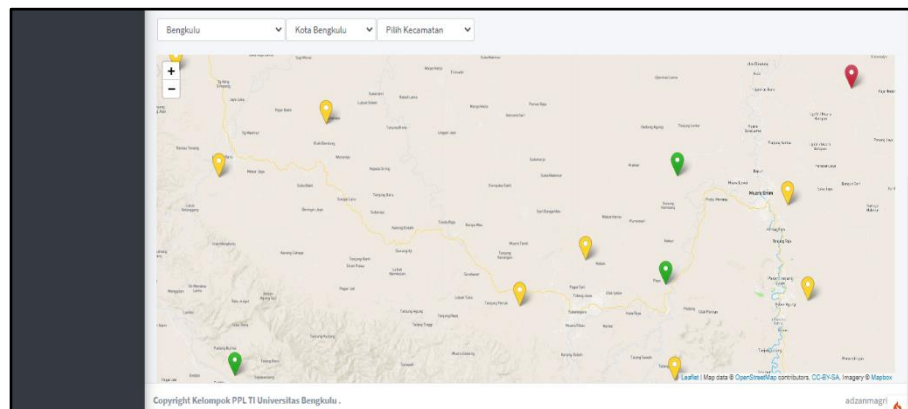
4.1. Implementasi Sistem dan Uji Coba Sistem

Analisis dan implementasi sistem untuk menganalisa kebutuhan sistem aplikasi pada perangkat lunak yang akan digunakan oleh masyarakat, dalam Sistem Informasi Dan Visualisasi Data Penyebaran Covid 19 berbasis Website. Aktivitas ini perlu dilakukan agar sistem informasi yang dibuat dapat berjalan dengan lancar tanpa kendala dan dapat diimplementasikan dengan baik.

4.1.1. *Dashboard Masyarakat*

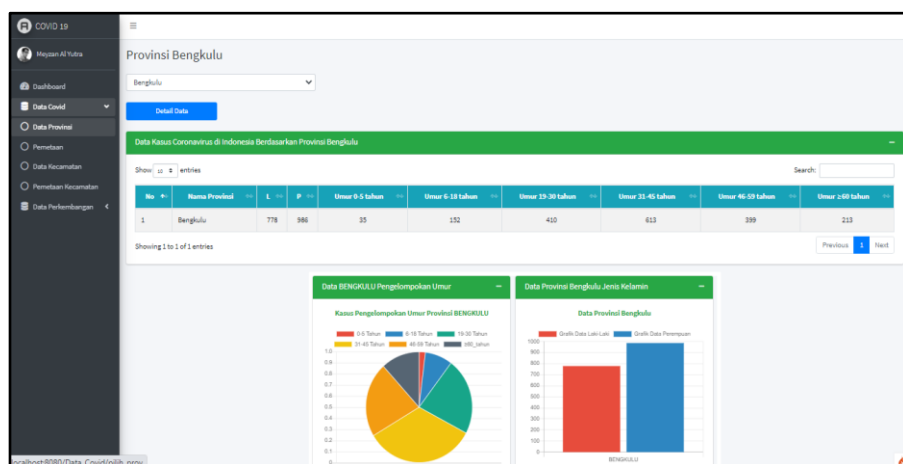


Gambar 4.1 *Dashboard Masyarakat*



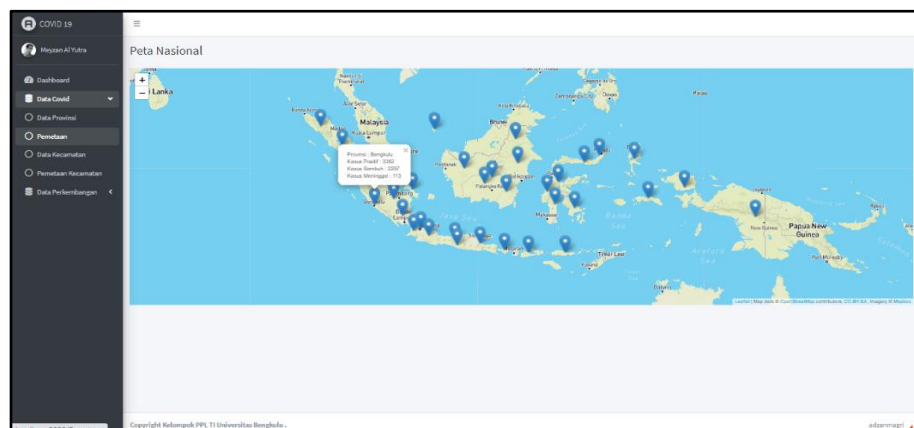
Gambar 4.2 *Pemetaan pada Dashboard Masyarakat*

Pada gambar 4.1, setelah masuk ke sistem maka *Masyarakat* akan masuk kedalam *Dashboard Masyarakat*, pada *Dashboard masyarakat* terdapat beberapa data covid seperti kasus positif, sembuh, meninggal, dsb. Masyarakat juga akan disajikan grafik lingkaran dimana berdasarkan jenis kelamin. Sedangkan pada gambar 4.2, pada tampilan dashboard ini terdapat map pemetaan penyebaran covid dimana kita bisa memilih wilayah yang ingin kita lihat. Berdasarkan provinsi, kabupaten, dan kecamatan. Nanti map akan menampilkan daerah yang terdampak covid 19 dengan klarifikasi low, medium, dan high.



Gambar 4.3 Data Provinsi pada Manajemen Data Covid Masyarakat

Pada gambar 4.3, merupakan tampilan halaman data covid untuk data provinsi, masyarakat dapat melihat data covid setiap provinsi yang ada. dan juga akan di sajikan grafik perbandingan.

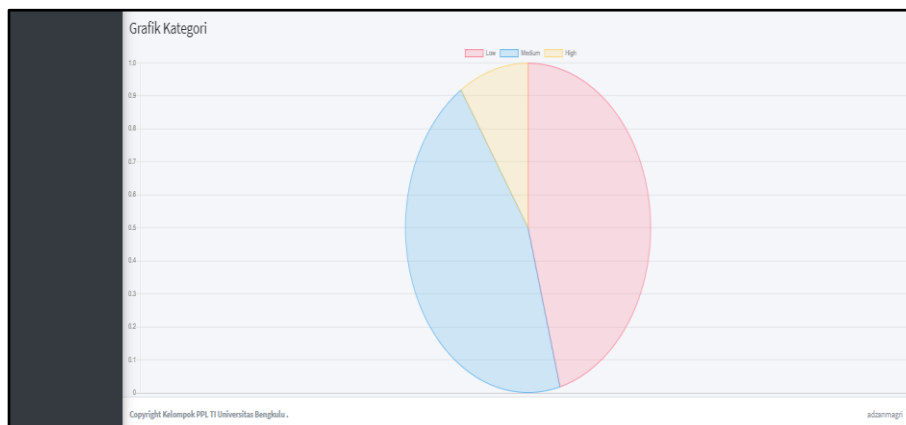


Gambar 4.4 Pemetaan pada Manajemen Data Covid Masyarakat

pada gambar 4.4, pada halaman ini masyarakat bisa Melihat titik penyebaran Covid 19 berdasarkan provinsi di seluruh indonesia melalui map, dan bisa menampilkan informasi mengenai covid masing provinsi. Seperti kasus covid, dan yang meninggal

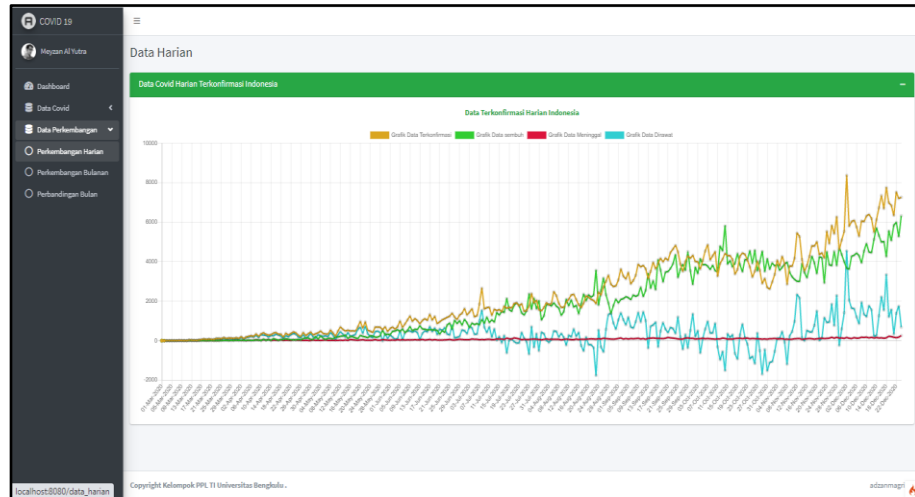
No	Provinsi	Kabupaten	Kecamatan	Kategori
468	BENGKULU	BENGKULU	BENGKULU	Low
526	BENGKULU	BENGKULU	BENGKULU	Low
555	BENGKULU	BENGKULU	BENGKULU	Low
583	BENGKULU	BENGKULU	BENGKULU	Low
598	BENGKULU	BENGKULU	BENGKULU	Low
599	BENGKULU	BENGKULU	BENGKULU	Low
600	BENGKULU	BENGKULU	BENGKULU	Low
606	BENGKULU	BENGKULU	BENGKULU	Low
628	BENGKULU	BENGKULU	BENGKULU	Low
657	BENGKULU	BENGKULU	BENGKULU	Low

Gambar 4.5 Data Kecamatan pada Manajemen Data Covid Masyarakat



Gambar 4.6 Grafik Data Kecamatan pada Manajemen Data Covid Masyarakat

Pada gambar 4.5 dan gambar 4.6 merupakan halaman dimana masyarakat dapat melihat data tiap-tiap kecamatan yang telah terdampak Covid di seluruh indonesia dimana ada informasi dari nama provinsi, kabupaten, kecamatan, dan kategori. Serta di sediakan grafik lingkaran perbandingan sesuai kategori.



Gambar 4.7 Perbandingan Hari Manajemen Data Perbandingan Masyarakat

Pada gambar 4.7 manajemen data perbandingan masyarakat, merupakan data perbandingan harian yang menampilkan perkembangan setiap hari covid mulai dari kasus positif, sembuh, meninggal, dan dirawat



Gambar 4.8 Perbandingan Bln Manajemen Data Perbandingan Masyarakat

Pada gambar 4.8 halaman dimana masyarakat memilih bulan dan tahun yang di inginkan informasinya, sehingga masyarakat dapat melihat data bulanan berupa grafik perbandingan setiap harinya di bulan tersebut yang menampilkan perkembangan mulai dari kasus positif, sembuh, meninggal, dan dirawat.



Gambar 4.9 Perbandingan Data *Manajemen Data Perbandingan Masyarakat*

Pada gambar 4.9 halaman dimana masyarakat melihat data perbandingan masing-masing bulan dari awal hingga sekarang yang menampilkan perkembangan mulai dari kasus positif, sembuh, meninggal, dan dirawat.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Sistem Informasi Dan Visualisai Data Penyebaran Covid 19 berbasis Website merupakan suatu sistem pengolahan data yang memanfaatkan teknologi komputerisasi. Berikut kesimpulan yang didapat dari proses analisis, perancangan dan implementasi sistem yang telah mampu menangani :

1. Menampilkan Data *Covid* dan Pemetaan *penyebaran covid*.
2. Sistem ini dikembangkan dengan mengikuti kerangka *client-server*. Desain lengkap dari Sistem dibuat terlebih dahulu, diikuti dengan implementasi sistem yang sebenarnya baik di server maupun perangkat *website*.
3. Penggunaan Sistem Informasi secara *Online*.

5.2. Saran

Saran yang diberikan agar sistem informasi yang telah dibangun dapat dikembangkan menjadi lebih baik lagi kedepannya yaitu dengan menambah jangkauan tampilan penyebaran covid, kemudian fungsi mencari rumah sakit yang menangani kasus covid di seluruh daerah. Sehingga informasi yang di dapatkan lebih banyak lagi.

DAFTAR PUSTAKA

- Buana, I. S. (2014). *Jago Pemrograman PHP Untuk Pemula dan Orang Awam*. Jakarta Timur: Dunia Komputer.
- Jigiyanto, H. (2005). *Analisis dan Desain Sistem Informasi : Penekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Andi.
- Munawar. (2006). *Pemodelan Visual dengan UML*. Yogyakarta Graha Ilmu.
- Satzinger, J. B. (2010). *System Analysis and Design with the Unified*. USA: Course Technology, Cengage Learning.
- Shalahuddin, M. (2018). *Rekayasa Perangkat Lunak Trsetruktur Dan Berorientasi Objek Edisi Revisi*. Bandung: Informatika.