

BAB II

TINJAUAN PUSTAKA

2.1 COVID-19

Menurut artikel yang tercantum pada *website* <https://covid19.go.id> yang diakses tanggal 10 Januari 2021, COVID-19 adalah penyakit menular yang disebabkan oleh jenis corona virus yang baru ditemukan. Sebagian besar (sekitar 80%) orang yang terinfeksi berhasil pulih tanpa perlu perawatan khusus. Sekitar 1 dari 5 orang yang terinfeksi COVID-19 menderita sakit parah dan kesulitan bernapas. Orang-orang lanjut usia (lansia) dan orang-orang dengan kondisi medis penyerta seperti tekanan darah tinggi, gangguan jantung dan paru-paru, diabetes, atau kanker memiliki kemungkinan lebih besar mengalami sakit lebih serius.

Namun, siapa pun dapat terinfeksi COVID-19 dan mengalami sakit yang serius. Orang dari segala usia yang mengalami demam dan/atau batuk disertai dengan kesulitan bernapas/sesak napas, nyeri/tekanan dada, atau kehilangan kemampuan berbicara atau bergerak harus segera mencari pertolongan medis. Jika memungkinkan, disarankan untuk menghubungi penyedia layanan kesehatan atau fasilitas kesehatan terlebih dahulu, sehingga pasien dapat diarahkan ke fasilitas kesehatan yang tepat.

2.2 Pengertian Data Spasial

Menurut Adil (2017), data spasial merupakan suatu data yang mengacu pada posisi, objek, dan hubungan diantaranya dalam ruang bumi. Data spasial merupakan salah satu item dari informasi, dimana di dalamnya terdapat informasi mengenai bumi termasuk permukaan bumi, di bawah permukaan bumi, perairan, kelautan, dan bawah atmosfer. Informasi spasial memakai lokasi dalam suatu sistem koordinat tertentu sebagai dasar referensinya. Karenanya, SIG mempunyai menghubungkan berbagai data pada suatu titik tertentu di bumi, menggabungkannya, menganalisis, dan akhirnya memetakan hasilnya dengan

tujuan membantu pengambilan keputusan berdasarkan kepentingan dan tujuannya masing-masing terutama yang berkaitan dengan aspek keruangan.

2.3 Pengertian Aplikasi

Peranan aplikasi yang sering dijumpai diberbagai bidang kehidupan, seperti pendidikan, bisnis, dan hiburan. Aplikasi yang dibangun pada penelitian ini merupakan berbasis Android yang artinya aplikasi yang dibangun adalah aplikasi *mobile*. Menurut kutipan yang diambil pada penelitian yang dilakukan oleh Muslihudin dan Helmiyanto (2020), aplikasi *mobile* adalah sebuah aplikasi yang memungkinkan untuk melakukan mobilitas dengan menggunakan perlengkapan seperti *Personal Digital Assistants* (PDA), telepon seluler atau *handphone*. Dengan menggunakan aplikasi *mobile* maka dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan kantor, browsing dan lain sebagainya. Secara keseluruhan aplikasi *mobile* merupakan aplikasi yang dapat digunakan walaupun pengguna berpindah dari satu tempat ke tempat yang lain tanpa terjadi pemutusan atau terputusnya komunikasi.

2.4 API Google Maps

Penelitian ini menggunakan layanan teknologi *Google Maps* untuk mengetahui lokasi kelurahan dan kecamatan yang ada di wilayah Kota Bandar Lampung. Menurut Wadi (2020) *API Google Map* merupakan tool yang digunakan untuk membuat aplikasi yang dapat mengakses server Google Maps, guna menampilkan peta atau informasi lokasi dari suatu tempat. API Google Maps memberikan kemampuan bagi para pengembang aplikasi untuk mengakses layanan Google Maps pada aplikasi yang dibuat.

2.5 Android

Penelitian ini membangun sistem informasi yang berbasis pada Android. Menurut *Google Developer Training Team* (2016), Android adalah sistem operasi dan *platform* pemrograman yang dikembangkan oleh *Google* untuk ponsel cerdas dan perangkat seluler lainnya (seperti tablet). Android bisa berjalan di beberapa macam perangkat dari banyak produsen yang berbeda. Android menyertakan *kit*

development perangkat lunak untuk penulisan kode asli dan perakitan modul perangkat lunak untuk membuat aplikasi bagi pengguna Android.

Menurut kutipan yang diambil dari penelitian Suryadi, Sutedi, dan Melda (2019), Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti *smartphone* dan komputer tablet. Android pada awalnya dikembangkan oleh Android, Inc., dengan dukungan keuangan dari Google, yang kemudian membelinya pada tahun 2005. Antarmuka pengguna Android umumnya merupakan manipulasi langsung, menggunakan gerakan sentuh yang mirip dengan tindakan nyata, seperti menggesek, mengetuk, dan mencubit untuk memanipulasi objek di layar, seperti serta keyboard virtual untuk menulis teks. Android adalah sistem operasi sumber terbuka, dan Google merilis kodenya di bawah lisensi Apache. Kode dengan open source dan lisensi lisensi di Android memungkinkan perangkat lunak untuk menjadi dimodifikasi dan didistribusikan secara bebas oleh pembuat perangkat, operator nirkabel, dan aplikasi pengembang. Selain itu, Android memiliki sejumlah besar Komunitas pengembang aplikasi (apps) yang memperluas fungsionalitas perangkat, umumnya ditulis dalam versi yang disesuaikan dari bahasa pemrograman Java.

Menurut Karman, Mulyono, dan Taqwa (2019), Android adalah sebuah sistem operasi perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Android bisa berjalan di beberapa macam perangkat dari banyak produsen yang berbeda. Android menyertakan kit *development* perangkat lunak untuk penulisan kode asli dan perakitan modul perangkat lunak untuk membuat aplikasi bagi pengguna Android. Android Juga menyediakan pasar untuk mendistribusikan aplikasi. secara keseluruhan, Android menyatakan ekosistem untuk aplikasi seluler. Versi Android yang kompatibel dan digunakan dalam penelitian ini adalah dimulai dari versi 8.0 sampai versi Android saat ini.

Menurut Yudhanto dan Wijayanto (2019), Perangkat Android memiliki banyak fitur perangkat keras di dalamnya. Inilah yang dapat dimanfaatkan *developer* dalam membangun aplikasi, diantaranya adalah :

- a. *Touchscreen*. Perangkat Android memiliki fitur layar sentuh yang memberikan beberapa kemungkinan bagi pengguna untuk berinteraksi dengan aplikasi menggunakan jari.
- b. *GPS*. Sistem operasi Android mendukung GPS yang memungkinkan *developer* untuk mengakses lokasi pengguna.
- c. *Accelerometer*. Android mendukung *accelerometer*, yaitu perangkat yang digunakan untuk mengatur percepatan.
- d. *SD Card*. Android memiliki fitur yang memungkinkan pengguna atau aplikasi untuk mengakses (menyimpan atau membuka) *file* pada *SD Card*.

Android memiliki banyak fitur perangkat lunak yang dapat digunakan oleh *developer* dalam mengembangkan aplikasi, yakni :

- a. *Internet*. Kemampuan akses internet pada Android memiliki banyak keunggulan. Berbagai informasi secara real-time dapat diperoleh dengan mudah dengan internet.
- b. *Audio dan Video Support*. Sistem operasi Android memungkinkan developer menyertakan audio dan video dalam aplikasi dengan mudah.
- c. *Contact*. Android memungkinkan akses ke kontak yang tersimpan pada perangkat.
- d. *Security*. Android memungkinkan aplikasi untuk melakukan banyak hal. Android juga menyiapkan keamanan berupa permission berkaitan dengan beberapa tugas.
- e. *Google APIs*. Sistem operasi Android memungkinkan dengan tidak terbatas membuat panggilan telepon, mengorganisasi kontak dan instal aplikasi. *Developer* juga dapat mengintegrasikan peta ke dalam suatu aplikasi dengan menggunakan *Maps API*.

2.6 Java

Bahasa pemrograman yang digunakan dalam membuat aplikasi berbasis Android adalah Java. Menurut Yuniansyah (2020), bahasa pemrograman adalah teknik komunikasi standar untuk mengekspresikan instruksi kepada komputer. Layaknya bahasa manusia, setiap bahasa memiliki tata tulis dan aturan tertentu. Bahasa pemrograman memfasilitasi seorang programmer untuk secara spesifik membuat perintah apa yang akan dilakukan oleh komputer selanjutnya, bagaimana data tersebut disimpan dan dikirim, dan apa yang akan dilakukan apabila terjadi kondisi yang variatif.

Bahasa pemrograman *Java* merupakan bahasa pemrograman berorientasi objek dan sampai saat ini terus berkembang. Banyak paket standar yang kemudian dikembangkan untuk semua kebutuhan, seperti *Java Lang* untuk elemen dasar, *Java.net* untuk kelas yang terkoneksi ke jaringan, *Java.awt* untuk antarmuka pengguna, dan *Java.applet* untuk aplikasi berbasis web. Karakteristik dari pemrograman Java adalah :

- a. Sederhana (*simple*) bahasa Java menggunakan sintak atau penulisan kode yang sederhana sehingga mudah dimengerti oleh programmer.
- b. Berorientasi objek (*object oriented*) Java menggunakan pemrograman berorientasi objek yang membuat program dapat dibuat secara modular dan dapat dipergunakan kembali.
- c. Terdistribusi (*distributed*) Java dikembangkan untuk dapat membuat aplikasi terdistribusi secara mudah karena terdapat pustaka networking yang telah terintegrasi pada Java.
- d. *Interpreted Program Java* dijalankan menggunakan interpreter yaitu *Java Virtual Machine* (JVM). Untuk itu, setelah kompilasi program Java dapat digunakan pada platform yang berbeda-beda.
- e. *Multithreaded Java* mempunyai kemampuan untuk membuat suatu program yang dapat melakukan beberapa pekerjaan secara sekaligus dan simultan.
- f. *Dynamic Java* didesain untuk dapat dijalankan pada lingkungan yang dinamis. Perubahan pada suatu class dengan menambahkan properties ataupun method dapat dilakukan tanpa mengganggu program yang menggunakan *class* tersebut.

2.7 DBMS (*Database Management System*)

DBMS yang digunakan dalam pengelolaan data pada aplikasi yang dibangun adalah MySQL. Menurut Suryadi, Sutedi, dan Melda (2019), basis data adalah kumpulan informasi yang disimpan dalam komputer sistematis sehingga dapat diperiksa menggunakan program komputer untuk memperoleh informasi dari basis data itu. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri basis data disebut sistem manajemen basis data (DBMS). Sistem *database* dapat dipelajari dalam informasi *sains*. Istilah "*database*" berasal dari ilmu komputer.

Konsep dasar *database* adalah kumpulan catatan, atau potongan pengetahuan. Sebuah *database* memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya, penjelasan yang dimaksud adalah disebut dengan skema. Skema menggambarkan objek yang diwakili oleh *database* dan hubungan antara objek-objek tersebut. Ada banyak cara untuk mengatur skema, atau untuk memodelkan struktur *database* yang dikenal sebagai model *database* atau model data. Model umumnya digunakan saat ini adalah model relasional, yang dalam istilah awam mewakili semua informasi dalam bentuk tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom. Dalam model ini, hubungan antar tabel adalah direpresentasikan menggunakan nilai yang sama antar tabel. Model lain seperti hierarki model dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antara tabel. Istilah *database* mengacu pada kumpulan data yang saling berhubungan, dan perangkat lunak harus menyebutnya sebagai sistem manajemen basis data (DBMS). Komponen-komponen yang menyusun lingkungan DBMS terdiri atas perangkat keras, perangkat lunak, data, prosedur, dan orang. Terdapat beberapa elemen basis data, yaitu *database*, tabel, entitas, atribut, nilai data, *file*, dan *record*.

2.8 Perangkat Lunak yang Digunakan

2.8.1 Android Studio

Android studio merupakan suatu aplikasi yang digunakan dalam pembuatan (*coding*) Android. Menurut Suryadi, Sutedi, dan Melda (2019), Android Studio adalah lingkungan pengembangan terintegrasi atau Pengembangan Terpadu *Environment* (IDE) untuk pengembangan aplikasi Android, berdasarkan *IntelliJ IDEA*. Selain menjadi editor kode *IntelliJ* dan alat pengembang yang kuat, Android Studio menawarkan lebih banyak fitur untuk meningkatkan produktivitas anda saat membuat aplikasi Android, misalnya:

- a. Sistem versi berbasis *Gradle* yang fleksibel
- b. Emulator yang cepat dan kaya fitur
- c. Lingkungan terpadu untuk pengembangan semua perangkat Android
- d. *Instant Run* untuk mendorong perubahan pada aplikasi yang sedang berjalan tanpa membuat APK baru
- e. *Template* kode dan integrasi *GitHub* untuk membuat fitur aplikasi yang sama dan mengimpor sampel kode
- f. Alat dan kerangka kerja pengujian yang luas
- g. Alat *Lint* untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah lainnya
- h. Dukungan C++ dan NDK
- i. Dukungan bawaan untuk *Google Cloud Platform*, memudahkan integrasi *Google Cloud*

Messaging dan *App Engine* setiap proyek di Android Studio berisi satu atau beberapa modul dengan *file* kode sumber dan *file* sumber daya. Jenis-jenis modul tersebut antara lain:

- a. Modul aplikasi Android
- b. Modul Perpustakaan
- c. Modul Mesin Aplikasi Google

Secara *default*, Android Studio akan menampilkan *file* proyek anda dalam tampilan proyek Android. Tampilan diatur oleh modul untuk menyediakan akses cepat ke *file* sumber utama proyek anda. Semua versi *file* muncul di bagian atas di bawah *Gradle Scripts* dan setiap modul aplikasi berisi folder berikut:

a. *manifest*: berisi file *AndroidManifest.xml*.

b. *java*: berisi file kode sumber *Java*, termasuk kode pengujian *JUnit*.

res: berisi semua sumber daya non-kode, seperti *layout XML*, *string UI*, dan gambar *bitmap*..

2.9 MySQL

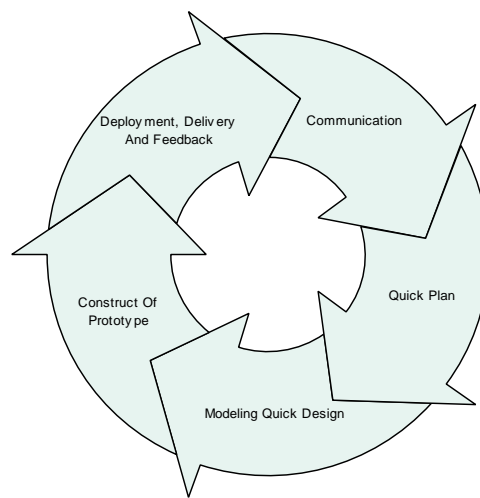
Pengelolaan DBMS yang digunakan dalam pembuatan sistem informasi ini adalah *MySQL*. Menurut kutipan yang diambil pada penelitian yang dilakukan oleh Muslihudin dan Helmiyanto (2020), *MySQL (My Structure Query Language)* adalah suatu perangkat lunak *database* relasi atau *Relational Database Management System (RDBMS)* yang didistribusikan gratis dibawah lisensi *GPL (General Public License)*. Dimana setiap orang bebas menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang dijadikan *closed source* atau komersial.

Menurut Solichin (2016), *MySQL* adalah sebuah perangkat lunak sistem manajemen basis data *SQL (Database Management System)* atau *DBMS* yang *multithread, multi-user*. Beberapa kelebihan *MySQL* antara lain: *free* (bebas di *download*), stabil dan tangguh, fleksibel dengan berbagai pemrograman, keamanan yang baik, dukungan dari banyak komunitas, dan kemudahan.

2.10 Metode Prototipe

Pengembangan perangkat lunak dalam sistem informasi ini menggunakan metode prototipe. Menurut kutipan pada penelitian Puspita (2019), *prototype* merupakan metode yang efektif dalam merancang perangkat lunak. *Prototype* dimulai dengan mengumpulkan kebutuhan yang akan di rancang. Pengembang mendefinisikan objek keseluruhan dari perangkat lunak, mengidentifikasi segala aktifitas yang diketahui dan kemudian melakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan *input* dan format *output*).

Menurut kutipan pada penelitian Gunawan dan Puspita (2017), *prototype model* adalah salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan metode prototype, pengembangan dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem. Prototipe memiliki lima tahapan siklus. Siklus atau ilustrasi dari metode prototipe dapat dilihat pada Gambar 2.1.



Gambar 2.1 Ilustrasi Model Prototipe

Pembuatan prototipe dimulai dengan dilakukannya komunikasi antara tim pengembang perangkat lunak dengan para pelanggan. Tim pengembang perangkat lunak akan melakukan pertemuan-pertemuan dengan para stakeholder untuk mendefinisikan sasaran keseluruhan untuk perangkat lunak yang akan dikembangkan, mengidentifikasi spesifikasi kebutuhan apa pun yang saat ini diketahui, dan menggambarkan area-area dimana definisi lebih jauh pada iterasi selanjutnya merupakan keharusan. Iterasi pembuatan prototipe direncanakan dengan cepat dan pemodelan (dalam bentuk "rancangan cepat") dilakukan. Suatu rancangan cepat berfokus pada representasi semua aspek perangkat lunak yang akan dilihat oleh para pengguna akhir (misalnya rancangan antar muka pengguna (*user interface*) atau format tampilan). Rancangan cepat (*quick design*) akan memulai konstruksi pembuatan prototipe. Prototipe kemudian akan diserahkan kepada para stakeholder dan kemudian mereka akan melakukan evaluasi-evaluasi tertentu terhadap prototipe yang telah dilakukan sebelumnya, kemudian akhirnya

akan memberikan umpan-balik yang akan digunakan untuk memperhalus spesifikasi kebutuhan. Iterasi akan terjadi saat prototipe diperbaiki untuk memenuhi kebutuhan dari para stakeholder, sementara pada saat yang sama memungkinkan kita untuk lebih memahami kebutuhan apa yang akan dikerjakan pada saat iterasi selanjutnya.

2.11 Pengujian *Black Box*

Metode pengujian yang dipakai sebagai uji coba aplikasi yang dibangun adalah *black box testing*. Menurut Rosa dan Shalahuddin (2018), *black box testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan pengeluaran perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus *login* maka kasus uji yang dibuat adalah:

- a. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
- b. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah.

2.12 Penelitian Terkait

Adapun penelitian terdahulu yang mendukung dan terkait dengan penelitian yang dilakukan dalam pembuatan aplikasi saat ini dapat dilihat pada tabel 2.1.

Judul	Hasil Penelitian	Kelebihan	Kekurangan
S-GIS Digitazing Skin Disease	Penelitian mengenai SIG penyebaran	Sistem ini membantu	Karna masih mengguakan

<p>Spread In Lampung Province Indonesia</p> <p>Oleh : Rahmalia Syahputri, Muhammad Said Hasibuan 2015</p>	<p>penyakit kulit menyimpulkan bahwa sistem memberikan informasi mengenai penyebaran penyakit kulit berdasarkan geografis provinsi lampung berbasiskan web.</p>	<p>Departemen Kesehatan menyiarkan informasi tersebut guna meningkatkan kesadaran Kesehatan di masyarakat.</p>	<p>website maka kadang terjadi kesalahan pengetikan sehingga website tidak bisa dibuka.</p>
---	---	--	---

Tabel 2.1 Lanjutan

<p>Pesona : Sistem Informasi Pemesanan Salon Online Berbasis Location Based Servis</p> <p>Oleh : Ani Setianni , Rahmalia Syahputri 2019</p>	<p>Aplikasi PESONA (Pemesanan Salon Online) menyimpulkan bahwa aplikasi PESONA dapat mempermudah transaksi pemesanan antara salon dan pelanggan.</p>	<p>Kelebihan dari aplikasi yang dibangun adalah penggunaan LBS untuk mengetahui posisi pelanggan, memungkinkan pelanggan dapat mengetahui harga layanan sebelum memesan.</p>	<p>Kekurangan dari aplikasi ini adalah tidak adanya fitur notifikasi konfirmasi pesanan dan notifikasi <i>full booking</i>.</p>
<p>Sistem Informasi Geografis Pemetaan Lokasi Klinik Dan Rumah Sakit Di Bandar Lampung</p> <p>Oleh : Novira, Ainun Delista 2019</p>	<p>Penelitian mengenai SIG pemetaan lokasi klinik dan rumah sakit di Bandar Lampung menyimpulkan bahwa sistem yang dibangun dapat dijadikan wadah media informasi klinik atau rumah sakit yang ada di Bandar Lampung</p>	<p>Kelebihan dari aplikasi yang dibangun adalah informasi mengenai klinik dan rumah sakit ditampilkan dengan detil seperti lokasi, nomor telepon, serta layanan dan fasilitas.</p>	<p>Kekurangannya adalah aplikasi hanya bisa diakses oleh pengguna smartphone Android saja. Ruang lingkup aplikasi hanya menampilkan informasi klinik dan rumah sakit di Bandar Lampung.</p>
<p>Rancangan Pemetaan Sebaran Covid-19 Di Kota Padang Berbasis Web Geospasial</p> <p>Oleh : Yuhefizar , Ervan Asri , Nasrullah 2020</p>	<p>Dalam penelitiannya mengenai pemetaan sebaran <i>COVID-19</i> Kota Padang berbasis web geospasial menyimpulkan bahwa aplikasi yang dibuat mampu menampilkan data <i>COVID-19</i> di Kota Padang hingga tingkat Kelurahan, sehingga informasi menyentuh kepada masyarakat paling bawah.</p>	<p>Kelebihan aplikasi yang dibangun adalah informasi yang ditampilkan mengenai data kasus <i>COVID-19</i> Kota Padang detil hingga kelurahan.</p>	<p>Tampilan website yang terlalu banyak menggunakan animasi membuat loading data lambat (penerimaan informasi lama). Jika diakses di daerah dengan jaringan internet yang buruk atau tidak stabil, maka informasi data kasus <i>COVID-19</i> Kota Padang tidak dapat diterima oleh pengakses</p>

Berdasarkan penelitian pada Perangkat Lunak Pemetaan Data Pantauan COVID-19 dibuat berbasis android yang menampilkan data pasien COVID-19 setiap kelurahan wilayah kota Bandar Lampung. Data tersebut menginformasikan data sembuh, kematian, positif dan suspek beserta zonasi berdasarkan warna. Aplikasi tersebut ringan dan tidak banyak memakan kuota data karna hanya menampilkan informasi pantauan COVID-1 kota Bandar Lampung. Berdasarkan Data Spasial Kota Bandar Lampung menyimpulkan bahwa , Keunggulan aplikasi pemetaan ini adalah Menggunakan peta yang tertitik di tiap kelurahan yang ada di Kota Bandar Lampung. Data kasus penderita COVID-19 baik itu kasus suspek, kasus konfirmasi, kasus kematian, dan kasus selesai isolasi tertera pada titik dalam peta lokasi berdasarkan kelurahan di tiap kecamatan yang berada di Kota Bandar Lampung. Grafik data jumlah kasus penderita COVID-19 yang ditampilkan setiap bulannya sampai dengan enam bulan yang lalu, dan Aplikasi ini juga dapat menghadirkan informasi/berita yang berkaitan dengan COVID-19, sehingga masyarakat/publik dapat mengetahui perkembangan kasus COVID-19.Sedangkan kelemahannya adalah tidak ada fitur notifikasi jika terdapat kasus baru dan tidak ada fitur notifikasi jika terdapat informasi/pemberitaan mengenai COVID-19 baru.

a.