

BAB 2 LANDASAN TEORI

2.1 Penelitian Terdahulu

Pertama, penelitian ini dilakukan oleh Nina Nuraprian, Meriska Defriani, Dayan Singasatia (2021) yang berjudul “Rancang Bangun Aplikasi Pemandu Wisata Di Kota Purwakarta Berbasis *Mobile*” perancangan yang dibuat tersebut bertujuan untuk mempermudah wisatawan dalam mencari wisata di kota purwakarta dengan menggunakan metode *waterfall* dan menggunakan pemodelan *Unified Modeling Language (UML)* yang terdiri dari 4 *use case* yaitu, *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*. Hasil dari penelitian tersebut menjadikan sebuah *aplikasi* berbasis *mobile* dengan memiliki informasi tentang wisata alam, wisata kuliner, dan wisata kebudayaan.

Selain itu, pada pembangunan aplikasi tersebut memiliki kekurangan seperti titik koordinat sehingga pengguna masih bingung dalam menentukan lokasi wisata tersebut.

Kedua, penelitian ini dilakukan oleh Abdul Basith, Muhyiddin Zainul Arifin, Moh.Anshori Aris Widya yang berjudul “Aplikasi Pemandu Wisata Religi Di Jawa Timur Berbasis *Android*” permasalahan pada penelitian penelitian tersebut yaitu banyaknya wisatawan religi yang tidak mengetahui lokasinya dan kurangnya informasi. Sehingga tujuan penelitian tersebut adalah memudahkan wisatawan religi dengan membangun aplikasi yang berfungsi sebagai *tour guide* dan memiliki fitur denah lokasi, informasi wisata, gambar wisata, lokasi wisata dan deskripsi wisata. Dalam pembangunan ini menggunakan metode *waterfall* dan menggunakan bahasa *java* dan *XML*.

Selain itu, pada pembangunan aplikasi tersebut memiliki kekurangan seperti panorama *Sphere*, *virtual audio mp3* sehingga tak hanya membaca wisatawan pun dapat mendengarkan informasi atau deskripsi tempat tersebut.

Ketiga, Penelitian yang dilakukan oleh Agus Salim, Rayhan Rahmat Aziz Gamawanto yang berjudul “Pembuatan Aplikasi Pemandu Pariwisata Berbasis *Mobile* Menggunakan *Flutter* Bagi Masyarakat Jawa Barat ” permasalahan dalam penelitian tersebut adalah dengan adanya kekayaan wisata dan budaya yang dimiliki oleh Provinsi Jawa Barat. Kebanyakan wisatawan berkunjung ke provinsi Jawa Barat untuk melihat kebudayaan dan tempat wisata yang alami dan terjaga keasriannya, hal ini peneliti bertujuan bagaimana cara menyampaikan informasi mengenai daya tarik atau objek pariwisata yang ada di provinsi Jawa Barat. Metode penelitian yang digunakan adalah *Rapid Application Development (RAD)* dan aplikasi yang dibangun berbasis *mobile*.

Kekurangan dalam pembangunan aplikasi tersebut seperti panorama *Sphere*, *virtual audio* sehingga tak hanya membaca wisatawan pun dapat mendengarkan informasi atau deskripsi tempat tersebut.

Keempat, Penelitian ini dilakukan oleh Hendra Surasa, Musfirah Putri Lukman, Hasrwati dengan judul “*Aplikasi Pemandu Wisata Kabupaten Kepulauan Selayar Menggunakan Google Maps API Pada Platform Android* ” kabupaten Kepulauan Selayar yang berada di provinsi Sulawesi Selatan memiliki begitu banyak wisata yang populer. Potensi wisata tersebut meliputi wisata sejarah, wisata budaya, wisata alam dan wisata bahari. Wisatawan yang ingin datang ke tempat wisata di kabupaten Selayar kekurangan informasi sehingga mencari informasi dahulu di internet atau wisatawan yang pernah berkunjung ke Kepulauan Selayar informasi yang dicari seperti objek wisata,



kegiatan tahunan pulau selayar, toko oleh-oleh, hotel dan lain-lain. masalah yang sering terjadi kurangnya informasi yang detail mengenai objek wisata yang sedang di cari dan belum ada sistem pemandu wisata di kabupaten selayar. Dari penelitian ini bertujuan untuk membantu wisatawan mencari informasi yang efektif dan dapat menemukan lokasi wisata secara tidak langsung juga dapat meningkatkan pelayanan pada bidang pariwisata di kabupaten selayar.

Kekurangan dalam pembanguna aplikasi ini adalah tidak ditemukan metode yang digunakan dan juga tampilan interface kurang menarik selain itu juga belum menggunakan virtual audio dan panorama *Sphere*.

Kelima, Penelitian ini dilakukan oleh Chiki Stelys Rajagukguk dengan judul “ Rancang Bangun Aplikasi Pemandu Wisata Danau Toba Berbasis *Web* Menggunakan Metode *Agile*.” Danau toba adalah salah satu tujuan tempat wisata yang berada di Sumatera Utara, danau toba juga memiliki berbagai objek wisata yang indah dan alami jadi permasalahan yang di alami adalah kurannya penerepan teknologi informasi sehinga mengakibatkan kurangnya wisatawan yang berkunjung. Tujuan dari penelitian ini adalah untuk mempermudah wisatawan dalam berkunjung ke danau toba dengan cara membuat apliaksi berbasis *web*, metode yang digunakan adalah *agile* yang berperan sebagai pengembangan sistem.

Kekurangan dalam pembangunan *aplikasi* ini adalah masih berbasis *web* sehinga kurang efisien karena harus membuka browser dahulu dan belum menerapkan *virtual audio* didalamnya.



Tabel 2 1 Penelitian Terdahulu

NO	Nama Peneliti	Judul	Perbedaan	Persamaan
1.	Nina Nurapriana, Meriska Defriani, Dayan Singasati (2021)	Rancang Bangun <i>Aplikasi</i> Pemandu Wisata Di Kota Purwakarta Berbasis <i>Mobile</i>	Sistem belum dilengkapi <i>API Maps</i> (titik <i>koordinat</i>) dan <i>Audio</i>	Dalam pembangun a sistem sama-sama menggunakan metode <i>waterfall</i> dan berbasis <i>Android</i>
2.	Abdul Basith, Muhyiddin Zainul Arifin, Moh. Ans hori Aris Widya	<i>Aplikasi</i> Pemandu Wisata Religi Di Jawa Timur Berbasis <i>Android</i>	<i>Sistem</i> belum dilengkapi dengan <i>virtual audio</i> dan panorama <i>Sphere</i>	Sistem sama sama menggunakan metode <i>waterfall</i> dan sudah dilengkapi <i>API Maps</i>
3.	Agus Salim, Rayhan Rahmat Aziz Gamawanto	Pembuatan <i>Aplikasi</i> Pemandu Pariwisata Berbasis <i>Mobile</i> Menggunakan <i>Flutter</i> Bagi Masyarakat Jawa Barat	Pembangunan sistem menggunakan metode <i>RAD</i>	Pembangunan <i>sistem</i> sama-sama berbasis <i>android</i>
4.	Hendra	Aplikasi	Sistem	Sistem

NO	Nama Peneliti	Judul	Perbedaan	Persamaan
	Surasa, Musfirah Putri Lukman, Hasrwati	Pemandu Wisata Kabupaten Kepulauan Selayar Menggunakan <i>Google Maps API</i> Pada <i>Platform Android</i>	belum menggunakan <i>virtual audio</i> untuk mendengarkan deskripsi dan fitur panorama <i>Sphere</i>	sama sama menggunakan <i>platform android</i> dan sudah menggunakan <i>Google maps API</i>
5.	Chiki Stelys Rajagukguk	Rancang Bangun Aplikasi Pemandu Wisata Danau Toba Berbasis Web Menggunakan <i>Metode Agile</i> .	Pembangunan sistem menggunakan <i>web</i> dan metode menggunakan <i>agile</i>	Sistem telah menggunakan <i>google maps API</i>

Berdasarkan penelitian terdahulu yang sudah dilakukan dengan kelebihan serta kelemahan sistem dapat dikembangkan melalui penelitian ini dengan menambahkan fitur laporan aduan, menunjukkan rute atau arah menuju destinasi wisata, menambahkan deskripsi berupa audio Mp3 dan panorama *sphere*.



2.2 Kajian Pustaka

Bagian ini memuat rangkuman teori-teori yang diambil dari buku atau literatur yang mendukung penelitian, serta memuat penjelasan tentang konsep dan prinsip dasar yang diperlukan untuk pemecahan permasalahan.

2.2.1 Sistem

Dalam Jurnal (Surasa et al., 2018), Scott Menurutnya, sistem terdiri dari komponen seperti masukan (*input*), pengolahan (*processing*), dan keluaran (*output*). Menurut Gapsper, ciri utama sistem adalah bahwa sistem beroperasi dalam suatu lingkungan, terdiri dari komponen yang saling berhubungan, dan memiliki satu fungsi atau tujuan utama.

Sistem informasi adalah kumpulan dari sub-sub sistem baik fisik maupun non fisik yang saling berhubungan satu sama dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna.

Adapun tiga aktifitas yang terjadi pada sistem informasi adalah *input*, *processing*, *output* dan penjelasannya adalah sebagai berikut :

- a. *Input* adalah sekumpulan data mentah dalam organisasi maupun luar organisasi untuk diproses dalam suatu sistem informasi.
- b. *Processing* adalah konversi atau pemindahan, manipulasi dan analisa *input* mentah menjadi bentuk yang lebih berarti bagi manusia.
- c. *Output* adalah distribusi informasi yang sudah diproses ke anggota organisasi dimana *output* tersebut akan digunakan. Informasi dalam hal ini juga membutuhkan umpan balik (*feedback*) yaitu *output* yang dikembalikan ke anggota organisasi yang berkepentingan untuk membantu mengevaluasi atau memperbaiki *input* (Jogiyanto & MBA, 2003).



2.2.2 Platform Android

Android merupakan sistem operasi untuk perangkat seluler berbasis *Linux* yang terdiri dari *aplikasi*, *middleware*, dan sistem operasi. (Andreanto et al., 2012). Kelebihan dari *android* adalah sistem terbuka atau *open source* bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang di gunakan berbagai pihak piranti. Karena OS (*operating sistem*) *android* ini salah satu sistem yang berkembang yang sangat pesat di antara OS lainnya seperti *Iphone Os*, *windows mobile*, *Symbian*.

Dalam Jurnal (Voutama & Novalia, 2021) Pengembangan aplikasi *Android* diperbolehkan untuk mendistribusikan aplikasi mereka dibawah skema lisensi apapun yang mereka inginkan. Pengembang memiliki beberapa pilihan dalam membuat aplikasi yang berbasis *Android*. Namun kebanyakan pengembang menggunakan *Eclipse* sebagai *IDE* untuk merancang *aplikasi* mereka. Hal ini dikarenakan *Eclipse* mendapat dukungan langsung dari *Google* untuk menjadi *IDE* pengembangan *aplikasi Android*. Aplikasi *Android* dapat dikembangkan pada berbagai sistem operasi, diantaranya adalah *Windows XP/Vista/7/8/10*, *MacOS X (MacOS X10.48* atau yang lebih baru), dan *Linux*. *Android* menjadi sistem operasi *mobile* yang populer hingga saat ini karena sifat *platform android* yang terbuka bagi para pengembang untuk mengembangkan aplikasi buatan sendiri. Arsitektur yang tersedia pada sistem operasi *Android* adalah sebagai berikut (Muhammad Sadeli, 2014):

- a. *Linux Kernel* Digunakan *Google* untuk membangun sistem *android* yang mencakup *memory management*, *security setting*, *power management* dan beberapa *driver hardware*.
- b. *Android Runtime* Terdiri dari *Core Libraries* dan *Dalvik Virtual Machine*.

- c. *Libraries Android* menyertakan satu set *library-library* dalam bahasa C/C++ yang digunakan pada komponen komponen *Android*.
- d. *Application Framework* Programmer memiliki akses penuh untuk memanfaatkan API (*Android Protocol Interface*) dalam mengatur fungsi-fungsi dasar *smartphone*.
- e. *Application* Merupakan fungsi teratas *Android* dan fungsi yang paling sering di akses bagi para pengguna.

2.2.3 Stisla

Stisla merupakan sebuah *HTML5* admin template responsif yang dibuat berdasarkan Bootstrap 4. Stisla diciptakan oleh Muhammad Nauval Azhar untuk membantu developer dalam pembuatan UI *dashboard web* tanpa membuat dari awal sehingga mempercepat proses pembuatan *web*

Stisla Dashboard Admin merupakan template gratis (*open source*) yang dibuat oleh orang Indonesia yaitu Muhamad Nauval Azhar. yang sering digunakan seorang *developer web* yang berfungsi sebagai *backend* proyek yang digunakan. Jadi *template Stisla Dashboard Admin* merupakan *dashboard* administrator yang dibuat dengan *bootstrap*.



Gambar 2. 1 Tampilan Bootstrap Stisla

2.2.4 Pemandu Wisata

Berdasarkan Keputusan Menparpostel Nomor KM.82/PW/102/MPPT-88, pramuwisata adalah orang yang bertanggung jawab untuk memberikan bimbingan, penerangan, dan petunjuk tentang objek wisata serta memiliki kemampuan untuk membantu wisatawan dalam hal apa pun yang mereka butuhkan. Menurut Dr. Adi Soenarno, M.B.A., Kamus Istilah Pariwisata dan Perhotelan, "Pramuwista" berarti mengantar tamu ke tempat wisata dan memberikan penjelasan tentang tempat wisata tersebut. (Muhajir, 2005)

Pemandu wisata atau disebut pramuwisata adalah seseorang yang bertugas memberikan bimbingan, penerangan dan petunjuk tentang objek wisata, serta membantu segala sesuatu yang diperlukan wisatawan (Menteri Pariwisata, Pos dan Telekomunikasi, 1988). Lingkup kegiatan pariwisata terdiri dari *transfer guide, walking guide / tour guide, local / expert guide, common guide dan driver guide* (Suyitno, 2005)

2.2.5 Google Maps API

Google Maps memberikan open source yang disediakan oleh google yang digunakan untuk pengembangan atau pembangunan aplikasi maupun web yang tidak di buat oleh pihak google. *Google Maps API* adalah *library javascript* yang memungkinkan Anda dengan mudah mengubah peta *Google Maps* yang ada. (Elian et al., 2012). Dalam penggunaan *Google Maps API* dapat menghemat waktu dalam pembangunan aplikasi sehingga perfokus pada data data yang akan ditampilkan yang artinya kita hanya menyiapkan datanya sedangkan peta yang akan di pakai dalam milik google sehingga tidak pusing dalam membuat peta suatu lokasi. Kelebihan dari *Google Maps Api* yaitu:

1. Didukung penuh oleh google sehingga terjamin dan bervariasi fitur yang ada pada *Google Maps API*.
2. Banyak pengembang yang membuat *Google Maps API* sehingga mudah dalam mencari referensi dalam pembangunan *aplikasi* atau *web*.

2.2.6 Virtual Audio

Audio yaitu suara yang dihasilkan dari getaran suatu benda supaya biasa ditangkap oleh manusia harus kuat minimal 20 kali perdetik. Suara merupakan getaran yang dihasilkan dari gesekan atau pantulan. *audio* merupakan elemen yang penting dalam membangun suatu sistem komunikasi dengan bentuk suara yang berupa suara elektrik yang membawa unsur unsur bunyi.

(Susilana & Riana, 2008) menerangkan bahwa media merupakan sebuah alat untuk memberikan perangsang bagi siswa supaya terjadi proses belajar. Dengan kata lain, media dapat diartikan sebagai sesuatu yang dapat menjembatani guru mengajar dan merangsang siswa untuk belajar.

Media pembelajaran dapat dikelompokan berdasarkan jenis dan cara penggunaannya. *Bretz* dalam Rohani (1997) mengelompokan media ke dalam tujuh kelas, yaitu:

a. Media *audio – motion – visual*

Media yang paling lengkap dalam arti penggunaan di kelas dalam segala kemampuan *audio* dan visual. Misalnya: *televisi*, *film*, video tape, dan *film TV recording*.

b. Media *audio – still – visual*

Media ini dapat menampilkan suara maupun gambar tanpa gerak. Misalnya: *sound film strip*, *sound slide set*, rekaman *still TV*.

c. Media *audio – semioticon*

Yaitu media yang berkemampuan untuk menampilkan suatu motion yang berupa titik-titik, tidak secara utuh. Misalnya: *telewriting*, dan *recorder telewriting*.

d. Media *motion – visual*

Mempunyai kemampuan seperti media kelas 1, kecuali suara (*audio*) yaitu berupa media *silent film*.

e. Media *still – visual*

Berkemampuan untuk menyampaikan informasi secara *visual*, tetapi tidak dapat menyajikan *motion*. Yang termasuk media ini ialah: halaman cetakan, film-strip, gambar.

f. Media *audio*

Yaitu media yang menggunakan suara semata-mata. Misalnya radio, *telepon*, *audio tape recorder*.

2.2.7 Panorama 360°

Menurut (Vinod et al., 2007) Intensity Based Image Mosaicing merupakan gambar panorama adalah gambar yang menciptakan kerealistisan foto, 3 dimensional, dan lingkungan yang bisa dinavigasi. Sebuah gambar panorama menangkap sekeliling lokasi dengan pandangan 360° pandangan *silindris* atau *spherical*.

Panorama di bagi menjadi beberapa jenis proyeksinya. Dalam jurnal “Rancang Bangun Peta Virtual 3D Kampus UNTAN dengan Fitur Panorama 360 X 180” (Panggabean et al., 2015) ada 4 jenis panorama:

1. *Planner* atau *Flat*

Hanya dapat di lihat dari sudut tertentu atau datar





Gambar 2. 2 Panorama Planner

2. *Cylinder* atau Silindris

Panorama silindris yaitu panorama yang dapat di lihat secara kurva disekeliling bagian dalam *silindris*. Jika melihat gambar secara datar, dapat di lihat ada kurva di sepanjang garis *axis horizontal*.



Gambar 2. 3 Panorama Cylinder

3. *Sphere*

Gambar *spherical* merupakan tipe gambar panorama yang memiliki format *equirectangular*, yang dimana menggambarkan sesuai dengan keadaan gambar axis horizontal secara 360° dan axis vertical secara 180° . Jika dipraktikan gambar panorama dapat

disimpan di proyeksi *Spherical* , yang dimana dapat di lihat didalam *sphere*. Panorama dengan gambar *vertical* yang lebar lebih efektif menggunakan metode *sphere* dari pada *silindris*.



Gambar 2. 4 Panorama Sphere 360° x 180°



Gambar 2. 5 Panorama Sphere 360° x 180°

4. *Cube* (Kubus)

Panorama yang telah di populerkan oleh *software QuickTime5*, metode yang diperkenalkan dengan panorama berbentuk kubus yang dapat di lihat dari bawah atau atas berbeda dengan *spherical*, ada 6 gambar dalam panorama berbentuk kubus panorama berbentuk kubus sangat efektif dengan sudut pandang *vertical* yang luas, panorama berbentuk kubus sangat mudah di modifikasi dengan *software* karena pada setiap bagian kubus terlihat normal.



Gambar 2. 6 Panorama Cube (Kubus)

2.2.8 Framework7

Framework7 merupakan kerangka kerja yang bersifat *Open Source* serta dapat digunakan secara *free* guna menciptakan ataupun meningkatkan aplikasi *mobile*, *desktop* ataupun *website* dengan tampilan asli, serta *framework7* ini serta menggambarkan perlengkapan *prototyping* yang sangat dibutuhkan buat menampilkan *prototype* aplikasi yang berperan guna terbuat ataupun digunakan sesegera serta secepat mungkin apabila benar dibutuhkan, pastinya tentang ini bakal membagikan kemudahan buat pengembang *android* ataupun *programmer* yang umumnya melaksanakan pengembangan ataupun membuat suatu aplikasi, khususnya *android* dan *ios*, karna disitu sudah dimasukkan sebagian model yang siap buat dibesarkan dan hanya tinggal meningkatkan fitur serta peranan cocok apa yang sudah diperlukan, dimana disitu kita dapat dengan gampang meniru ataupun apalagi mengcopy *source code* model *interface* dari tampilan *template* tersebut, tinggal gimana kita mengaktifkan fungsi- fungsi serta membiasakan fitur yang kita butuhkan nantinya (Aminulloh et al., 2020).

Fitur-fitur dalam *Framework7* meliputi:

1. *Framework7* merupakan *framework* dengan sumber terbuka (*open source*) serta bisa digunakan secara *free*. *Framework7* mempunyai *syntax* yang gampang dan mirip dengan *jQuery* buat memulai tanpa penundaan.



2. *Framework7* mempunyai pustaka *built- in FastClick* yang bermanfaat buat menerapkan kontrol terhadap penundaan klik pada saat meraba *UI*.
3. *Framework7* mempunyai sistem *grid layout built- in* guna mengendalikan *elemen* kamu secara *responsive*.
4. *Framework7* bisa secara dinamis muat taman dari *template* lewat *flexible router API(router API yang fleksibel)*.

2.2.9 MySQL

Menurut (Sibero, 2013) “MySQL atau dibaca “My Sekuel” dengan adalah suatu *RDBMS (Relational Database Management Sistem)* yaitu aplikasi sistem yang menjalankan fungsi pengolahan data”. Menurut (Kawistara & Hidayatullah, 2015) “MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi *web*. Contoh DBMS lainnya adalah : *PostgreSQL (freeware)*, *SQL Server*, *MS Access* dari *Microsoft*, *DB2* dari *IBM*, *Oracle* dan *Oracle Corp*, *Dbase*, *FoxPro*, dsb”. Dari pemaparan di atas dapat di simpulkan bahwa *MYSQL* yaitu aplikasi *DBSM* yang berfungsi untuk mengelola data guna membangun aplikasi .

2.2.10 User Interface

Menurut jurnal (Loman & Erandaru, 2022) Aprilia menjelaskan *User Interface* ataupun yang kerap disebut dengan *UI* merupakan tampilan *visual* dari produk di mana sistem serta *user* (pengguna) dihubungkan. Tampilan *UI* dirancang semenarik bisa jadi dengan mencermati aspek ataupun *elemen* berbentuk pemilihan warna, *tipografi*, *layout*, foto, serta perihal yang lain yang membuat cantik tampilan. Oleh sebab itu, bisa disimpulkan kalau *UI* ialah gimana tampilan sesuatu produk di amati oleh pengguna.

Menurut jurnal (Loman & Erandaru, 2022) Sutanto menyatakan Desain *UI* terfokus pada apa saja yang bisa jadi dicoba oleh pengguna serta membenarkan kalau antarmuka mempunyai elemen yang gampang diakses, dimengerti, serta digunakan buat memfasilitasi aksi tersebut. *UI* menyatukan



konsep-konsep dari desain interaksi, desain *visual*, serta *arsitektur data*. Desainer butuh mencermati sebagian ciri desain *UI* selaku pertimbangan buat mendesain tampilan yang menarik serta menyenangkan pengguna. Sebagian ciri desain *UI* yang baik antara lain :

1. Jelas serta ringkas.
2. *Responsive*.
3. Tidak berubah- ubah serta memakai elemen *UI* yang *universal*.
4. Pemakaian warna serta tekstur yang strategis.
5. Memakai tipografi sebagai hirarki serta kejelasan.
6. Bisa mengomunikasikan apa yang terjadi

2.2.11 Desain Basis Data

Menurut (Wijaya et al., 2021) Desain basis data merupakan proses merancang struktur serta organisasi basis data buat membenarkan jika data ditaruh secara efektif, gampang, serta andal. Perancangan basis data meliputi pemilihan *tipe* basis data, perancangan skema basis data, penentuan hubungan antar tabel, serta penetapan kriteria integritas informasi. Sebagian langkah umum dalam desain basis data

1. Analisis kebutuhan bisnis
Langkah awal dalam mendesain *database* ialah menguasai kebutuhan bisnis serta gimana data hendak digunakan. Ini mengaitkan identifikasi entitas bisnis, atribut, serta ikatan antar entitas.
2. Buat skema basis data
Sesudah analisis kebutuhan bisnis berakhir, langkah berikutnya merupakan membuat skema *database*. Skema *database* ini berisi tabel, kolom, serta kunci utama.
3. Normalisasi:
Normalisasi merupakan proses guna membenarkan kalau *database* mempunyai struktur yang efektif serta tidak terjadi redundansi data. Normalisasi bisa dicoba dalam sebagian langkah (tingkat 1 hingga 3) normalisasi bergantung kompleksitas *database*.



4. Integritas data

Sesudah proses normalisasi berakhir, langkah berikutnya merupakan memastikan kriteria integritas informasi. Ini tercantum pembuatan kunci asing, ketentuan validasi, serta batas referensi.

2.2.12 Balai Pelestarian Cagar Budaya (BPCB)

Menurut (Novrisa, 2014) BPCB Trowulan merupakan lembaga yang bertugas dalam bidang pemeliharaan, perlindungan, pemugaran, dokumentasi, penyuluhan, pengamanan terhadap peninggalan purbakala yang bergerak maupun yang tidak bergerak serta situs termasuk yang berada di lapangan, ruangan, peninggalan arkeolog bawah air ataupun sebagainya. BPCB dipimpin oleh seorang Kepala yang berada di bawah dan bertanggung jawab kepada Direktur Jenderal Kebudayaan. Selain itu pelaksanaan zonasi cagar budaya juga dilaksanakan oleh pihak BPCB. Pemanfaatan zona pada cagar budaya ini pun dapat dilakukan untuk tujuan rekreatif, edukatif, apresiasif atau religi. Penetapan luas, tata letak, dan fungsi zona ditentukan berdasarkan hasil kajian dengan mengutamakan peluang peningkatan kesejahteraan rakyat.

2.2.13 CodeIgniter

Dalam Jurnal (Destiningrum & Adrian, 2017) Menurut Betha Sidik (2012) *CodeIgniter* merupakan “ Suatu *framework php* yang bersifat *open source* serta memakai tata cara *MVC* (*Model, View, Controller*) guna mempermudah pengembang maupun *programmer* dalam membangun suatu aplikasi berbasis *website* tanpa wajib buatnya dari awal”

Dalam situs resmi *codeigniter*, (*Official Website CodeIgniter*,2002) mengatakan kalau *CodeIgniter* ialah *framework PHP* yang kuat serta sedikit bug. *CodeIgniter* ini dibentuk buat para pengembang dengan bahasa pemrogram *PHP* yang memerlukan perlengkapan guna membuat *website* dengan fitur lengkap. *Framework CodeIgniter* dibesarkan oleh



RickEllis, CEO Ellislab, Inc. kelebihan dari *framework codeigniter* bila dibanding dengan *framework* lain merupakan sebagai berikut :

1. *Free (Open Source)* Kerangka kerja *Codeigniter* mempunyai lisensi dibawah *Apache/ BSD open source* sehingga bersifat bebas ataupun *free*.
2. Berukuran kecil Ukuran yang kecil ialah keunggulan tertentu bila dibanding *framework* lain yang berukuran besar serta memerlukan *resource* yang besar serta pula dalam eksekusi ataupun penyimpanannya.
3. Memakai konsep *M- V- C Codeigniter* ialah konsep *M- V- C (Model- View- Controller)* yang mengizinkan pemisahan antara *layer application- logic* serta *presentation*. Dengan konsep ini kode *PHP, query MySQL, Javascript* serta *CSS* bisa saling dipisah-pisahkan sehingga ukuran file jadi lebih kecil serta lebih gampang dalam perbaikan kedepannya ataupun *maintenance*.
 - a. Model Kode menggambarkan program (berbentuk *OOP class*) yang digunakan guna berhubungan dengan *database MySQL* sekalian buat memanipulasinya(*input- edit- delete*).
 - b. *View* menggambarkan kode program berbentuk *template* ataupun *PHP* buat menampilkan informasi pada *browser*.
 - c. *Controller* menggambarkan Kode program (berbentuk *OOP class*) yang digunakan buat mengendalikan aliran ataupun dengan kata lain sebagai pengontrol model serta *view* (Destiningrum & Adrian, 2017).

2.2.14 Unified Modeling Language (UML)

UML(Unified Modeling Language) adalah standar bahasa pemodelan dalam mengembangkan perangkat lunak yang menggunakan teknik pemrograman berbasis objek (Shalahuddin & Rosa, 2018). Di buatnya *UML* agar memenuhi kebutuhan dalam pemodelan visual sistem yang






akan di bangun. Jenis- jenis diagan yang ada dalam *UML* antara lain *use case diagaram*, *activity diagram*, *sequence diagram*, dan *class diagram*

a. *Use Case Diagram*

Use Case Diagram berfungsi dalam memberikan keterangan atau deskripsi dari interaksi diantara beberapa pelaku dengan sistem informasi yang akan dirancang (Shalahuddin & Rosa, 2018) *Use Case diagram* yang dimana akan memberikan keterangan kegunaan yang ada dalam suatu sistem dan orang yang akan menggunakan *tools* tersebut.

Tabel 2 2 *Use Case Diagram*


NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan kedudukan yang pengguna mainkan pada saat korelasi dengan <i>use case</i>
2		<i>Generalization</i>	Diaman ikatan objek anak (<i>descendent</i>) bermacam sikap serta struktur informasi dari objek yang terdapat diatasnya objek induk (





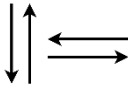
NO	GAMBAR	NAMA	KETERANGAN
			<i>ansector)</i>
		<i>Use Case</i>	Deskripsi dari urutan aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .

b. *Activity Diagram*

Activity Diagram adalah diagram proses -proses yang terjadi di *sistem*, *activity diagram* yaitu pengembangan dari *use case* yang memiliki alur aktivitas / proses bisnis yang terjadi pada sistem tersebut. Dalam buku *Rekayasa Perangkat Lunak* karangan Rosa A.S mengatakan, "*Diagram* aktivitas tidak menjelaskan kelakuan aktor. Dapat diartikan bahwa dalam pembuatan *activity diagram* hanya dapat dipakai untuk menggambarkan alur kerja atau aktivitas sistem saja."

Tabel 2 3 *Komponen Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing masing <i>class</i> antar muka saling berinteraksi satu sama lain.







2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3		<i>Initial node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Active final node</i>	Bagaimana objek dibentuk atau diakhiri.
5		<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan atau Tindakan yang harus diambil pada kondisi tertentu.
6		<i>Line connector</i>	Digunakan untuk menghubungkan satu <i>symbol</i> dengan <i>symbol</i> lainnya.

c. *Sequence Diagram*

Sequence Diagram merupakan diagram yang menampilkan interaksi antara objek yang ada di dalam sistem yang disusun dalam sebuah urutan atau rangkaian. Dimensi *sequence* terdiri dari *vertical* (waktu) dan *horizontal* (objek-objek yang terkait). Berikut gambar dari *sequence diagram*:

Tabel 2 4 Komponen *Sequence Diagram*


NO	GAMBAR	NAMA	KETERANGAN
----	--------	------	------------


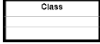



1		<i>Actor</i>	Menggambarkan orang yang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan.
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari <i>form</i> .
4		<i>Control Class</i>	Menggambarkan penghubung antara <i>boundary</i> dengan <i>Tabel</i> .
5		<i>A focus of Control & A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya <i>message</i> .
6		<i>A Message</i>	Menggambarkan pengiriman pesan.

d. *Class Diagram*

Class Diagram adalah *diagram* struktur karena sistem yang dimodelkan dengan macam macam komponen yang ada pada *UML* dengan menunjukkan *sistem class*, atributnya, metode dan hubungan antar objek.

Tabel 2 5 *Komponen Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Ikatan dimana objek anak (<i>descendent</i>)

NO	GAMBAR	NAMA	KETERANGAN
			berbagi perilaku serta struktur informasi dari objek yang terdapat di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya guna menjauhi asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
5		<i>Realization</i>	Operasi yang memang dicoba oleh sesuatu objek
6		<i>Dependency</i>	Ikatan dimana perubahan yang terjalin pada sesuatu elemen mandiri (<i>independent</i>) hendak



NO	GAMBAR	NAMA	KETERANGAN
			mempengaruhi <i>element</i> yang bergantung padanya elemen yang tidak mandiri.
7	—	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.





