BAB II
TINJAUAN PUSTAKA

A. Dasar Teori

1. Teori yang Berkaitan Topik Penelitian

a. Aplikasi

Menurut Plimbi (2013:8), aplikasi secara umum yaitu suatu sistem atau program komputer yang memiliki fungsi sebagai fasilitas digital yang membantu penggunanya menyelesaikan tugas atau pekerjaan berupa pengolahan kata, gambar, angka, suara, dan sebagainya. Tidak hanya itu, aplikasi juga memiliki banyak fungsi lain yang terbagi ke dalam banyak bidang atau kategori, seperti hiburan, bisnis, edukasi, dan lain-lain.

Aplikasi merupakan suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak *system* yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna.

b. Lagu Daerah

Menurut Van Hoeve (2012:1940), Lagu merupakan syair-syair yang dinyanyikan dengan irama yang menarik agar menjadi enak didengar. Lagu bisa menjadi media curahan hati orang yang membuat lagu itu tadi. Sehingga lagu yang dinyanyikan bisa bernuansa sedih, senang, maupun jenaka.

Lagu dapat dikelompokkan berdasarkan ragamnya, diantaranya :

1) Lagu anak – anak

2) Lagu daerah

3) Lagu perjuangan

4) Lagu keroncong

5) Lagu stambul

6) Lagu populer

7) Lagu seriosa

8) Lagu langgam

9) Lagu dangdut

Menurut Hoeve (2012:1940), Lagu daerah adalah lagu atau musik yang berasal dari suatu daerah tertentu dan menjadi populer dinyanyikan baik oleh rakyat daerah tersebut maupun rakyat lainnya. Bentuk lagu ini sangat sederhana dan menggunakan bahasa daerah atau bahasa setempat. Lagu daerah banyak yang bertemakan kehidupan sehari-hari sehingga mudah untuk dipahami dan mudah diterima dalam berbagai kegiatan rakyat. Pada umumnya pencipta lagu daerah ini tidak diketahui lagi alias *noname* (NN).

Menurut sifat dan keberasalannya lagu daerah dibedakan menjadi dua. Lagu rakyat dan Lagu klasik. Lagu rakyat yaitu lagu yang berasal dari rakyat di suatu daerah. Lagu rakyat tersebar secara alami yang disampaikan secara lisan dan turun-temurun.

c. Lagu Daerah Kalimantan Tengah

Menurut Rimalati (2012:9), Lagu Daerah Kalimantan Tengah merupakan lagu yang berasal dari rakyat di suatu daerah di Kalimantan Tengah yang yang dipakai untuk acara pernikahan, kematian, berladang, berlayar, menenun, dan sebagainya.Kalimantan Tengah

Berikut ini adalah kumpulan lirik lagu daerah yang berasal dari Kalimantan Tengah.

1) Naluya.

2) Tumpi Wayu.

3) Kalayar.

4) Ka Danau.

5) Isen Mulang.

6) Lewungku Utusku.

7) Malauk Manjala.

8) Manasai.

Lagu-lagu daerah dibedakan berdasarkan daerah dan kebanyakan lagu daerah tercetus karena kebudayaan atau mayoritas mata pencarian masyarakat setempat sehingga muncul sebuah tradisi yang tercipta karena rasa syukur masyarakat kepada Tuhan terhadap rezeki yang melimpah.

d. Fungsi Lagu Daerah

Fungsi lagu daerah banyak sekali, diantaranya:

1) Upacara Adat.

Di Sumba sebagai pengiring roh dalam upacara Merapu dan musik angklung dalam upacara Seren Taun (panen padi) di Sunda.

2) Pengiring tari dan pertunjukan

Lagu lagu langgam yang dipadu dengan gamelan di jawa dipakai untuk mengiringi pementasan tari Serimpi di jawa tengah. Bisa juga dipakai unuk pertunjukan wayang kulit, kethoprak, ludruk, drama dan pertunjukan lainnya.

3) Media Bermain

Contohnya cublak cublak suweng dari Jawa Tengah, ampar ampar pisang di Kalimantan Selatan, dan pok ame ame dari Betawi.

4) Sebagai media komunikasi

Pertunjukan musik atau lagu di suatu tempat dapat dipakai media komunikasi secara tidak langsung yang ditandakan dengan banyaknya orang yang melihat pertunjukan.

5) Sebagai media penerangan

Kini lagu dalam aneka iklan layanan masyarakat maupun lagu populer dipakai sebagai media penerangan. Contohnya lagu tentang pemilu, imunisasi, juga lagu bernafaskan agama menjalankan fungsi ini.

e. *Web Mobile*

Menurut Abdurahman (2016:49), *Web Mobile* adalah lanjutan dari aplikasi web tradisional. Saat online menggunakan *smartphone*, tablet, atau komputer, desainnya akan beradaptasi dengan perangkat yang digunakan tersebut. Desain yang fleksibel ini memaksimalkan kemampuan web browser yang digunakan di perangkat mobile yang digunakan.

Lebih jelasnya, *web mobile* merupakan halaman-halaman yang berisi informasi yang ditampilkan oleh browser seperti *Mozilla Firefox*, *Google Chrome* atau yang lainnya, tapi memiliki tampilan yang dapat menyesuaikan dengan perangkat pengaksesnya.

f. *Web/Website*

1) Pengertian *Website*

Menurut Arief (2011:7), *website* atau *web* adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) didalamnya yang menggunakan protokol HTTP (*Hypertext Transfer Protocol*) dan untuk mengaksesnya menggunakan perangakat lunak yang disebut *browser*.

Website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya

2) Pengertian Situs *Web*

Menurut Arief (2011:8), Pengertian *Website* adalah kumpulan dari halaman *web* yang sudah dipublikasikan di jaringan internet dan memiliki domain/URL (*Uniform Resource Locator*) yang dapat diaskes semua pengguna internet dengan cara mengetikkan alamtnya. Contoh alamt situs *web* www.amikom.ac.id, http://rudyantoarief.com.

Sebuah situs *web* (sering disingkat: *website*, *site*) adalah sebutan bagi kumpulan halaman web (*web page*), yang umumnya merupakan bagian dari suatu nama domain (*domain name*) atau subdomain di *World Wide Web* (WWW) di internet.

3) Pengertian *World Wide Web* (WWW)

Arief (2011:8), *World Wide Web* atau yang biasa disingkat WWW merupakan kumpulan situs *web* yang dapat diakses di internet berisikan semua informasi yang dibutuhkan semua pengguna internet.

World Wide Web (WWW) terdiri dari seluruh situs *web* yang tersedia kepada publik. Halaman-halaman sebuah situs *web* diakses dari sebuah URL yang menjadi *root*, yang disebut *homepage*, dan biasanya disimpan dalam *server* yang sama.

4) Jenis *website*

Dalam pengelompokkan jenis *web*, lebih diarahkan pada:

a) Jenis-jenis *web* berdasarkan sifatnya :

(1) Website Dinamis

Website Dinamis merupakan yang secara teratur diperuntukkan untuk update sesering mungkin, isi informasi interaktif dua arah dari pemilik serta pengguna *website*. Biasanya disediakan halaman backend untuk mengedit kontent dari *website*.

(2) *Website* Statis

Website Statis adalah apabila isi infomasi *website* tetap, jarang berubah, dan isi informasinya searah hanya dari pemilik *website*. Artinya adalah untuk melakukan perubahan pada suatu halaman dilakukan secara manual dengan mengedit *code* yang yang menjadi struktur dari *website* tersebut. Dalam pengembangannya, *website* statis hanya bisa diupdate oleh pemiliknya saja atau *web developer*.

b) Jenis-jenis *web* berdasarkan bahasa pemprograman :

(1) *Server Side*

Merupakan *website* yang menggunakan bahasa pemprograman yang tergantung kepada tersedianya server. Seperti PHP, ASP, dan sebagainya. Jika tidak ada server, *website* yang dibangun menggunakan bahasa pemprograman di atas tidak dapat berfungsi sebaimana mestinya.

(2) *Client Side*

Client Side Adalah *website* yang tidak membutuhkan server dalam menjalankannya.

c) Jenis-jenis *web* berdasarkan tujuannya :

(1) *Personal Web*, *website* yang berisi infomasi pribadi seseorang.

(2) *Corporate Web*, *website* yang dimiliki oleh sebuah perusahaan.

(3) *Portal web*, *website* yang mempunyai banyak layanan, mulai dari layanan berita, jasa, email, dan jasa-jasa lainnya.

(4) *Forum Web*, sebuah *web* yang bertujuan sebagai media diskusi.

5) Generasi *website*

Teknologi *web* yang sedemikian rupa berkembangnya sehingga para ahli telah memberikan penomoran untuk mengklasifikasikan generasi dari teknologi *web* yang digunakan.

a) *Web* 1.0

Web 1.0 merupakan teknologi teknologi *web* generasi pertama yang merupakan revolusi baru di dunia internet karena telah mengubah cara kerja dunia industri dan media. Pada dasarnya, *website* yang dibangun pada generasi pertama ini secara umum dikembangkan untuk pengaksesan informasi dan memiliki sifat yang sedikit interaktif.

b) *Web* 2.0

Web 2.0 pertama kalinya diperkenalkan oleh O'Reilly Media pada tahun 2004 sebagai teknologi *web* generasi kedua yang mengedepankan kolaborasi dan *sharing* informasi secara *online*.

c) *Web* 3.0 / *Semantic Web*

Web 3.0 berpotensi menjadi generasi teknologi di dunia internet. Saat ini, definisi *web* 3.0 sangat beragam mulai dari pengaksesan *broadband* secara *mobile* sampai kepada layanan *web* berisikan perangkat lunak bersifat *on-demand*. *Web* 3.0 adalah sekumpulan teknologi yang menawarkan cara

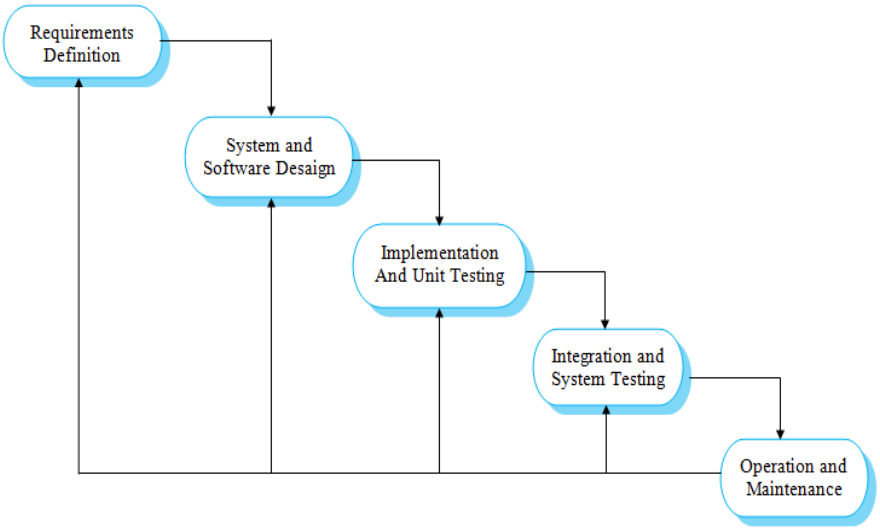
baru yang efisien dalam membantu komputer mengorganisasi dan menarik kesimpulan dari data online“. *Content web* ditampilkan tidak hanya dalam format bahasa manusia yang umum (*natural language*) tetapi juga dalam format yang dapat dibaca dan digunakan oleh mesin. Melalui *semantic web* inilah, berbagai perangkat lunak akan mampu mencari, membagi, dan mengintegrasikan informasi dengan cara yang lebih mudah. Dengan demikian, unsur kecerdasan buatan (*Artificial Intelligence* /AI) merupakan bagian penting pada *web* 3.0 / *Semantic Web*, sehingga *web* menjadi semakin cerdas.

2. Pemodelan yang Digunakan

   a. *Waterfall* Model

      Menurut Rizky (2011:61), *waterfall* model sebagai salah satu teori dasar dan seakan wajib dipelajari dalam konteks siklus hidup perangkat lunak, merupakan sebuah siklus hidup yang terdiri dari mulai fase hidup perangkat lunak sebelum terjadi hingga pascaproduksi. *Waterfall* model memiliki definisi sendiri bahwa sebuah hidup perangkat lunak memiliki sebuah proses yang *linear* dan *sekuensial*.

      Menurut Sommerville (2011: 30-31), tahapan utama dari *waterfall model* langsung mencerminkan aktifitas pengembangan dasar. Terdapat 5 tahapan pada *waterfall model*, yaitu *requirement analysis and definition*, *system and software design*, *implementation and unit testing*, *integration and system testing*, dan *operation and maintenance*.



Gambar 1. *Waterfall* Model
(Sommerville, 2011:30)

Berikut adalah penjelasan dari tahapan-tahapan tersebut:

a) *Requirement Definition*

Merupakan tahapan penetapan fitur, kendala dan tujuan sistem melalui konsultasi dengan pengguna sistem. Semua hal tersebut akan ditetapkan secara rinci dan berfungsi sebagai spesifikasi sistem.

b) *System and Software Design*

Dalam tahapan ini akan dibentuk suatu arsitektur sistem berdasarkan persyaratan yang telah ditetapkan. Dan juga mengidentifikasi dan menggambarkan abstraksi dasar sistem perangkat lunak dan hubungan-hubungannya.

c) *Implementation and Unit Testing*

Dalam tahapan ini, hasil dari desain perangkat lunak akan direalisasikan sebagai satu set program atau unit program. Setiap unit akan diuji apakah sudah memenuhi spesifikasinya.

d) *Integration and System Testing*

Dalam tahapan ini, setiap unit program akan diintegrasikan satu sama lain dan diuji sebagai satu sistem yang utuh untuk memastikan sistem sudah memenuhi persyaratan yang ada. Setelah itu sistem akan dikirim ke pengguna sistem.

e) *Operation and Maintenance*

Dalam tahapan ini, sistem diinstal dan mulai digunakan. Selain itu juga memperbaiki *error* yang tidak ditemukan pada tahap pembuatan. Dalam tahap ini juga dilakukan pengembangan sistem seperti penambahan fitur dan fungsi baru.

b. *Black Box Testing*

Menurut Pressman (2013:551) pada pengujian *Black-Box*, pengujian dilakukan dengan objek yang diuji terfokus kepada kebutuhan fungsional dari perangkat lunak. Pengujian *Black-Box* memungkinkan pembuat perangkat lunak untuk menentukan kondisi yang terjadi untuk suatu masukan yang akan menjalankan semua kebutuhan fungsional dari perangkat lunak yang dibuat.

Untuk melakukan pengujian pada sistem, penulis menggunakan metode *black box testing*, pengujian ini dilakukan untuk menentukan apakah *web* ini sudah layak atau belum untuk digunakan serta apakah sudah sesuai dengan apa yang diharapakan.

Pengujian *black box* merupakan tahap yang berfokus pada pernyataaan fungsional perangkat lunak. *Test case* ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya. Apakah masukkan data sudah berjalan sebagai mana mestinya dan apakah informasi yang tersimpan dapat dijaga. Pengujian *black box* memungkinkan pembuat perangkat lunak memanfaatkan serangkaian kondisi input yang sepenuhnya menggunakan persyaratan fungsional untuk suatu program.

Pengujian *black box* berusaha menemukan kesalahan dalam kategori :

1) Fungsi-fungsi yang tidak benar atau hilang
2) Kesalahan *interface*
3) Kesalahan dalam struktur data atau akses *database* eksternal
4) Kesalahan kerja, inisialisasi dan kesalahan terminasi.

c. PIECES *(Performance, Information, Economic, Control, Efficiency, Service)*

Menurut Jogiyanto (2012:23), dalam tahapan analisis dalam penelitian terdapat langkah-langkah dasar yang dilakukan oleh penulis yang terdiri dari analisis kelemahan sistem, analisis kebutuhan sistem, analisis kelayaakan sistem.

Bentuk Analisis yang digunakan penulis dalam penelitian ini menggunakan *PIECES (Performance, Information, Economic, Control, Efficiency, Service)* sebagai metode analisis untuk

menganalisis kelemahan-kelemahan sistem yang sedang berjalan sehingga dapat diusulkan suatu perbaikannya.

d. *Unified Modeling Language* (UML)

Menurut Lisdawati (2016:24), UML singkatan dari *Unified Modeling Language* yang berarti bahasa pemodelan standar. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan yang lainnya harus mengikuti *standart* yang ada.

Unified Modeling Language adalah bahasa untuk menspesifikasi, memvisualisasikan, serta merekonstruksi bangunan dasar sistem perangkat lunak, termasuk melibatkan pemodelan aturan-aturan bisnis.

UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

1) Merancang perangkat lunak.
2) Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3) Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4) Mendokumentasi sistem yang ada, proses-proses dan organinsasinya.

Beberapa litelatur menyebutkan bahwa UML menyediakan sembilan jenis diagram, yang lain menyebutkan delapan diagram karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan, dan diagram pewaktuan digabung

menjadi diagram interaksi. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya statis atau dinamis. Sedangkan diagram yang digunakan dalam penelitian ini adalah:

1) Diagram Kelas (*Class Diagram*)

   Bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.

2) Diagram *Use-Case*

   Brsifat Statis. Diagram ini memperlihatkan himpunan *use-case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

3) Diagram Interkasi dan *Sequence* (urutan)

   Bersifat Dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.

4) Diagram Aktivitas (*Activity Diagram*)

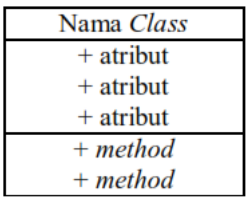
   Bersifat Dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem.diagram ini terutama penting

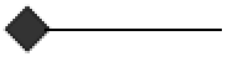
dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

Berikut ini adalah simbol-simbol dari masing-masing diagram:

1) Diagram Kelas / *Class Diagram*

Tabel 1. Simbol *Class Diagram*

Nama Komponen	Keterangan	Simbol
1	2	3
Class	*Class* adalah blok-blok pembangun pada pemrograman berorientasi obyek. Sebuah *class* digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari *class*. Bagian tengah mendefinisikan *property*/atribut *class*. Bagian akhir mendefinisikan *method-method* dari sebuah *class*.	
Association	Sebuah asosiasi merupakan sebuah *relationship* paling umum antara 2 *class* dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 *class*. Garis ini bisa melambangkan tipe-tipe *relationship* dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah *relationship*. (Contoh : *one-toone, one-to-many, many-to-many*).	

1	2	3
Composition	Jika sebuah *class* tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *Composition* terhadap *class* tempat dia bergantung tersebut. Sebuah *relationship composition* digambarkan sebagai garis dengan ujung berbentuk jajar genjang berisi/solid.	
Dependency	Kadangkala sebuah *class* menggunakan *class* yang lain. Hal ini disebut *dependency*. Umumnya penggunaan *dependency* digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain. Sebuah *dependendcy* dilambangkan sebagai sebuah panah bertitik-titik.	
Aggregation	*Aggregation* mengindikasikan keseluruhan bagian relationship dan biasanya disebut sebagai relasi	

Sumber : Grady Booch. 1999. Visual modeling With Relational Rose 2000 And UML

2) Diagram *Use Case*

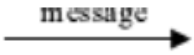
Tabel 2. Simbol *Use Case Diagram*

Nama Komponen	Keterangan	Simbol
Use Case	*Use case* digambarkan sebagai sebagai lingkaran *elips* dengan nama *use case* dituliskan didalam *elips* tersebut.	
Actor	*Actor* adalah pengguna sistem. *Actor* tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplilkasi lain dan membutuhkan *input* atau memberikan *output*, maka aplikasi tersebut juga bisa dianggap sebagai *actor*.	
Association	Asosiasi digunakan untuk menghubungkan *actor* dengan *uce case*. Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara *Actor* dengan *Use Case*.	

Sumber : Grady Booch. 1999. Visual modeling With Relational Rose 2000 And UML

3) Diagram Urutan / *Sequence Diagram*

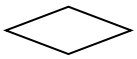
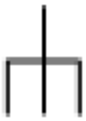
Tabel 3. Simbol *Sequence Diagram*

Nama Komponen	Keterangan	Simbol
Object	*Object* merupakan *instance* dari sebuah *class* dan dituliskan tersusun secara *horizontal*. Digambarkan sebagai sebah *class* (kotak) dengan nama *object* didalamnya yang diawali dengan sebuah titik koma.	
Actor	*Actor* juga dapat berkomunikasi dengan *object*, maka *actor* juga dapat diurutkan sebagai kolom. Simbol *actor* sama dengan simbol pada *Actor Use Case Diagram*.	
Lifeline	*Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *lifeline* adalah garis putus-putus vertikal yang ditarik dari sebuah *object*.	
Activation	*Activation* dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah *lifeline*. Mengindikasikan sebuah obyek yang akan melakukan sebuah aksi.	
Message	*Message* digambarkan dengan anak panah *horizontal* antara *Activatiion Message* mengindikasikan komunikasi antara *object-object*.	

Sumber : Grady Booch. 1999. Visual modeling With Relational Rose 2000 And UML

4) Diagram Aktivitas / *Activity Diagram*

Tabel 4. Simbol *Activity Diagram*

Simbol	Keterangan
	Titik Awal
	Titik Akhir
	*Activity*
	Pilihan untuk mengambil keputusan
	Fork : Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	*Rake* : Menunjukkan adanya dekomposisi
	Tanda waktu
	Tanda Pengiriman
	Tanda Penerimaan
	Aliran akhir (*Flow Final*)

Sumber : Grady Booch. 1999. Visual modeling With Relational Rose 2000 And UML

3. Perangkat Lunak yang Digunakan

a. *Hypertext Preprocessor* (PHP)

Menurut Rudyanto (2011:12), *Hypertext Preprocessor* (PHP) adalah bahasa *server-side scrifting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis.

Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di server dan kemudian hasilnya dikirimkan ke browser dalam format HTML dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman web yang dinamis, yaitu halaman web yang yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web*.

Hypertext Preprocessor (PHP) termasuk dalam *Open Source Product*, sehingga *Source Code* PHP dapat diubah dan didistribusikan secara bebas. Untuk versi terbaru dari PHP dapat langsung secara gratis di situs resmi PHP: http://www.php.net. PHP juga dapat berjalan pada berbagai *web server* seperti IIS (*internet information Server*), PWS (*Personal Web Server*), Apache, Xitami. PHP juga mampu lintas *platform*. Artinya PHP dapat berjalan dibanyak sistem informasi yang beredar saat ini, di antaranya : Sistem Operasi Windows, Linux, Mac OS, Solaris.

b. *My Structure Query Language* (MySQL)

Menurut Arif (2011:151), MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber pengelolaan datanya.

My Structure Query Language (MySQL) adalah *database* yang paling populer dan program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multi-user*.

My Structure Query Language adalah salah satu jenis *database sever* yang sangat terkenal dan banyak digunakan untuk mebangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya. Karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database* nya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan skala menengah kecil. MySQL juga bersifat *open source* dan *free* (tidak perlu membayar untuk menggunakannya).

My Structure Query Language merupakan *database* yang pertama kali didukung oleh bahasa pemprograman *script* untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan *software* pengembangan aplikasi web yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya memnggunakan bahasa pemprograman *script* PHP.

c. Twitter Bootstrap

Menurut Core Team (2016), Bootstrap adalah sebuah *library framework CSS* yang dibuat khusus untuk bagian pengembangan *front-end website*.

Bootsrap merupakan salah satu framework HTML, CSS, *javascript* yang paling populer dikalangan *web developer*. Bootstrap telah menyediakan kumpulan komponen *class interface* dasar yang telah dirancang sedemikian rupa untuk menciptakan tampilan menarik, bersih, dan ringan. Selain komponen *class interface*, *bootstrap* juga memiliki fitur grid yang berfungsi untuk mengatur *layout* pada halaman *website* yang bisa digunakan dengan sangat mudah dan cepat.

Salah satu contoh *website* besar yang menggunakan *framework bootstrap* adalah twitter, *interface* twitter dibangun dengan menggunakan bootstrap karena sebenarnya *bootstrap* dikembangkan oleh *developer twitter* itu sendiri sehingga *bootstrap* juga disebut "*twitter bootstrap*". Hampir semua *web developer* telah menggunakan *bootstrap* untuk membuat tampilan front-end menjadi lebih mudah dan sangat cepat.

1) *Cascading Style Sheet* (CSS)

Cascading Style Sheet (CSS) adalah salah satu bahasa pemprograman desain *web* (*style sheet language*) yang

mengontrol format tampilan sebuah halaman *web* yang di tulis dengan menggunakan bahasa penanda (*markup language*). Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang bahasa pemprograman CSS diaplikasikan untuk segala dokumen XML, termasuk SVG dan XUL.

Cascading Style Sheet (CSS) dibuat untuk memisahkan konten utama (biasanya dibuat dengan menggunakan bahasa HTML atau sejenisnya) dengan tampilan dokumen yang meliputi *layout*, warna dan *font*. Pemisahan ini dapat meningkatkan daya akses konten pada *web*, menyediakan lebih banyak fleksibilitas dan kontrol dalam spesifikasi dari sebuah karakteristik dan sebuah tampilan, memungkin untuk membagi banyak halaman untuk sebuah *formatting* dang mengurangi kerumitan dalam penulisan kode dan struktur dari konten, contohnya teknik *tableless* pada *layout* desain *web* (*layout* tanpa tabel).

Tujuan utama CSS diciptakan adalah untuk membedakan konten dari dokumen dan tampilan dari dokumen, dengan itu pembuatan ataupun pemprograman ulang *web* akan lebih mudah dilakukan.

2) *Hyper Text Markup Language* (HTML)

a) Pengertian *Hyper Text Markup Language* (HTML)

Hyper Text Markup Language (HTML) merupakan bahasa pemprograman *web* yang memiliki sintak atau aturan tertentu dalam menuliskan *script* atau kode-kode, sehingga *browser* dapat menampilkan informasi dengan membaca kode-kode HTML.

> Menurut Anhar (2010:40), HTML (*Hyper Text Markup Language*) adalah sekumpulan simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang digunakan untuk menampilkan halaman pada *web browser*. Tag-tag HTML selalu diawali dengan <x> dan diakhiri dengan </x> dimana x *tag* HTML itu seperti b, i, u, dan sebagainya.

b) Penulisan Kode HTML

Program yang digunakan untuk membuat dokumen HTML menggunakan HTML *editor*. Ada banyak HTML *editor* yang bisa digunakan , antaranya : Notepad, Notepad++, FrontPage, Dan DreamWeaver.

Setiap penulisan dokumen HTML harus diawali dan ditutup dengan tag HTML <HTML></HTML>. *Tag* HTML memberi tahu *browser* bahwa yang ada di dalam kedua *tag* tersebut adalah dokumen HTML. Bagian *header* dari dokumen HTML diapit oleh *tag* <HEAD></HEAD> di dalam bagian ini biasanya ada dimuat *tag* TITLE yang menampilkan judul dari halaman pada *title browser*. Bagian *body* dari

dokumen HTML diapit oleh *tag* <BODY></BODY>. Dokumen *body* digunakan untuk menampilkan *text image link*, dan semua yang akan ditampilkan pada halaman *web*.

d. NotePad++

Notepad++ Team (2016), menjelaskan bahwa *Notepad++* adalah aplikasi pemprograman yang berguna untuk mengedit teks dan skrip kode-kode pemprograman. *Notepad++* juga memiliki nomor urut barisan kode program sehingga memudahkan dalam mencari baris kesalahan pada program.

e. XAMPP

Menurut Arif (2011:154), XAMPP adalah sebuah *software web server apache* yang didalamnya sudah tersedia *database server mysql* dan *support* php programming. XAMPP merupakan software yang mudah digunakan, gratis dan mendukung instralasi di Linux dan Windows. Sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP *Server*, MySQL *Database*, dan penerjemah bahasa yang ditulis dengan bahasa pemprograman PHP dan Perl.

B. Penelitian Relevan

Penulis menelusuri beberapa hasil penelitian terdahulu yang memiliki keterkaitan dengan penelitian yang akan penulis lakukan. Daftar penelitian relevan tersebut adalah sebagai berikut :

Tabel 5. Penelitian Relevan

No.	Penulis/ Tahun	Topik Penelitian	Metode Pengembangan PL	Hasil	Perbedaan
1	2	3	4	5	6
1	Reza Hanifan Griswidianto /2015	Aplikasi Pengenalan Lagu Daerah Berbasis Mobile dalam Upaya Melestarikan Lagu-Lagu Daerah pada Siswa Sekolah Dasar Di Bandung	*Waterfall*	merancang dan membuat sebuah aplikasi berisi lagu-lagu daerah yang menarik untuk dilihat dan didengar agar dapat bermanfaat bagi siswa untuk mengenal dan melestarikan lagu-lagu daerah di Indonesia	Pada penelitian ini hanya memperkenalkan judul lagu, suara dan asal daerah saja tidak disertai video sedangkan penulis menyajikan informasi asal daerah disertai dengan video dan lirik lagu yang dapat dilihat.
2	Hendri Ahmadian dkk/2017	Rancang Bangun Aplikasi Lagu Daerah di Indonesia Berbasis Android	*Waterfall*	membuat rancangan aplikasi lagu-lagu daerah yang terdapat di Indonesia berbasis android yang memiliki fitur berupa asal lagu daerah, lirik lagu daerah serta arti lagu daerah dalam bahasa Indonesia dan bahasa inggris. Hasil dari penelitian ini adalah sebuah aplikasi android yang memiliki halaman awal berupa tampilan pulau-pulau di Negara Indonesia dan setiap lagu memiliki lirik.	Pada penelitian ini hanya memperkenalkan judul lagu, suara, lirik dan asal daerah saja tidak disertai video sedangkan penulis menyajikan informasi asal daerah disertai dengan video dan lirik lagu yang dapat dilihat.

1	2	3	4	5	6
3	Ulya Ilhami Arsyah dkk/ 2017	Perancangan Media Pembelajaran Lagu-Lagu Daerah pada Yayasan Pendidikan Al-Mawaddah Teladan Kisaran Berbasis Multimedia	*Waterfall*	Membuat inovasi baru dalam proses belajar mengajar di sekolah khususnya pada anak usia dini agar lebih menarik, fun, efektif, serta tidak membuang banyak tenaga bagi para guru untuk mengajar dan juga dapat menyingkat waktu proses belajar mengajar.	Pada penelitian ini menggunakan Adobe Flash Professional CS6 dimana aplikasi yang dibuat tidak bisa lagi menambahkan lagu baru karena tidak tersedia halaman kelola data lagu disini hanya hanya asal lagu, judul saja diserta dengan lirik lagu tanpa suara sedangkan penulis menyajikan informasi asal daerah disertai dengan video dan lirik lagu yang dapat dilihat dan dikelola.

Kesimpulan:

Dapat disimpulkan bahwa penelitian yang dikembangkan dari beberapa penelitian sebelumnya dengan penelitian yang dibuat penulis memiliki perbedaan yaitu informasi yang digunakan penelitian-penelitian yang ada hanya memperkenalkan judul lagu, suara, lirik dan asal daerah saja tidak disertai video sedangkan penulis menyajikan informasi asal daerah disertai dengan video dan lirik lagu yang dapat dilihat dan dikelola.