



BAB II

TINJAUAN PUSTAKA DAN

LANDASAN TEORI

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Penelitian tentang pelayanan administrasi yang sudah pernah dilakukan sebelumnya yaitu penelitian dari Rima Mawarni dan Rianda Venia Audina (2018) dengan judul penelitian “Sistem Informasi Pemesanan Pementasan Seni Tari Berbasis *Website*” di Sanggar Cangget Budaya Kotabumi Lampung Utara. Sistem ini dirancang untuk memudahkan pengguna dalam memesan tarian dengan melihat informasi data tarian dan biaya pemesanan. Rancangan sistem ini menggunakan metode pengembangan sistem *Prototyping* dan berbasis *website*[3].

Penelitian lainnya dari Stevani Olivia Djustine (2020) dengan judul penelitian “Perancangan Sistem Informasi Administrasi Pendaftaran *Member* dan *Non-Member* pada *M.C Studio*”. Sistem ini memudahkan pengguna dalam pendaftaran sebagai *member* dan *non-member*, dapat melihat jadwal senam, data paket senam beserta biaya, data kehadiran, data pembayaran, terdapat data laporan member yang berisi tanggal daftar dan tanggal berhenti sebagai member, laporan pembayaran member dengan bukti pembayaran. rancangan sistem ini diimplementasikan menggunakan bahasa pemrograman PHP, *database MySQL*, dan metode pengembangan *Waterfall*[4].

Penelitian berikutnya dengan judul penelitian “Perancangan Sistem Informasi Sanggar Kesenian Pada Kota Jambi Berbasis *Website*” dari Abriyanto, A Muhammad (2019). Sistem ini dibangun agar memberikan kemudahan bagi masyarakat dalam melakukan pencarian informasi sanggar kesenian yang diinginkan seperti lokasi sanggar. Mengetahui cara pendaftaran dan pembelian tiket jika ada pertunjukan. Sistem ini dijadikan sebagai media promosi bagi pemilik sanggar. Sistem akan dibuat menggunakan bahasa pemrograman PHP dan *database MySQL* dengan metode pemodelan menggunakan UML[5].

Penelitian yang terakhir dari M. Al Farabi Irwansyah dan Grace Gata (2019) dengan judul penelitian “Sistem Informasi Administrasi dan *Membership* Pada *King's Gym*”. Sistem ini dibuat dengan menggunakan Microsoft Visual 2008 dan *database MySQL*. Pasa sistem ini terdapat proses pembayaran yang terkomputerisasi, laporan absen *member*, laporan rekapitulasi penambahan *member* baru dan laporan perbulannya.[6]

Berdasarkan penelitian – penelitian sebelumnya, penulis mengajukan judul “Sistem Informasi Manajemen Sanggar Senam Berbasis *Website* pada Sanggar Senam Juanda” yang merupakan sistem berbasis *website* yang memanfaatkan teknologi dan internet serta notifikasi Telegram. Sistem berbasis *website* yang dirancang ini akan menggunakan bahasa pemrograman PHP, *database* MySQL untuk menyimpan semua data, dan metode pengembangan *waterfall*.

Tabel 2. 1 Tabel Perbandingan Tinjauan Pustaka

| Penulis, Tahun dan Judul | Permasalahan | Tujuan Penelitian | Hasil Penelitian |
|--|--|---|--|
| Rima Mawarni dan Rianda Venia Audina (2018) “Sistem Informasi Pemesanan Pementasan Seni Tari Berbasis <i>Website</i> ” | Keterbatasan dalam menyampaikan informasi mengenai data tarian serta biaya pemesanan tiket tarian. | Membangun sistem informasi pemesanan pementasan seni tari berbasis <i>website</i> | Sistem ini mempermudah para pengguna untuk memesan tarian dimanapun selama terhubung dengan internet dan dapat meningkatkan kinerja dan kualitas pelayanan |
| Stevani Olivia Djustine (2020) “Perancangan Sistem Administrasi Pendaftaran <i>Member</i> dan <i>Non-Member</i> pada M.C <i>Studio</i> ” | Semua kegiatan pengelolaannya masih menerapkan sistem manual sehingga pelayanan yang kurang memuaskan bagi member dan non-member dan memakan waktu yang lama | Merancang sistem informasi administrasi pendaftaran member dan non-member pada M.C. <i>Studio</i> agar lebih baik dan efektif | Sistem menampilkan laporan <i>member</i> , pembyaaran member dan <i>non-member</i> setiap bulan. |
| Abriyanto, A Muhammad (2019) “Perancangan Sistem | kurangnya informasi terhadap sanggar kesenian di Kota | Merancang Sistem Informasi Sanggar Kesenian Pada | Sistem ini mempermudah pengunjung dalam melihat <i>event</i> yang akan |

| | | | |
|--|--|--|--|
| Informasi Sanggar Kesenian Pada Kota Jambi Berbasis <i>Website</i> ” | Jambi sehingga masih sulit mencari lokasi sanggar, cara pendaftaran, informasi pementasan dan cara pembelian tiket pertunjukan. | Kota Jambi Berbasis <i>Website</i> | dilaksanakan pada kesenian Jambi dan membantu pihak kesenian Jambi dalam mengelola informasi sanggar di kota Jambi |
| M. Al Farabi Irwansyah dan Grace Gata (2019) “Sistem Informasi Administrasi dan Membership Pada King’s Gym” | Pengelolaan data menggunakan pembukuan sehingga terjadi kekeliruan dalam transaksi dan perhitungan | Memudahkan proses administrasi, <i>membership</i> menjadi lebih efisien dan akurat | Sistem ini mempermudah proses pencarian data, perhitungan pembayaran, membuat rekapitulasi penambahan member. |
| Renanda Sinta N K (2022) “Sistem Informasi Manajemen Sanggar Senam Berbasis Website pada Sanggar Senam Juanda” | Pembayaran senam tidak sesuai dengan jumlah member yang hadir, tidak ada pencatatan data sewa sanggar, laporan pemasukan tidak ada, pencatatan menggunakan buku besar. | Merancang dan membangun sistem informasi manajemen sanggar senam berbasis website pada sanggar senam juanda. | Sistem ini mempermudah pengelolaan data, pembuatan laporan pemasukan, mempermudah proses pendaftaran, reservasi, pembayaran, absensi, dan mengetahui jadwal. |

2.2. Landasan Teori

Dalam penelitian ini, diperlukan adanya teori-teori yang mendasar untuk menunjang proses penelitian. Berikut merupakan teori-teori yang digunakan dalam penelitian ini:

2.2.1 Sistem Informasi Manajemen

Sistem merupakan sekumpulan sumber daya yang saling terkait yang ingin mencapai suatu tujuan (Bodnar dan Hopwood, 2006). Menurut Baridwan, Sistem adalah suatu jaringan kerja dari prosedur yang saling berhubungan. Informasi merupakan suatu data diorganisasi yang dapat mendukung ketepatan pengambilan keputusan (Bodnar dan Hopwood, 2006). Menurut Mulyadi (2001) Informasi adalah olahan data ke dalam bentuk yang dapat memberikan arti bagi penerima dan dapat dijadikan sebagai dasar pengambilan keputusan saat ini atau mendatang. Adapun kualitas dari suatu informasi tergantung dari tiga hal menurut Al Bahra yaitu :

- a) Akurat (*accurate*)
Suatu informasi harus akurat dikarenakan dalam proses penyampaian informasi dari sumber informasi ke penerima informasi mungkin mengalami banyak gangguan yang dapat mengubah atau merusak informasi tersebut.
- b) Tepat Waktu (*timelines*)
Informasi merupakan landasan dalam pengambilan keputusan sehingga informasi yang sampai ke penerima tidak boleh terlambat, laporan yang dibutuhkan dapat disampaikan tepat waktu.
- c) Relevan (*relevance*)
Informasi dapat dirasakan manfaatnya bagi yang membutuhkan informasi

Manajemen merupakan proses atau kegiatan yang dilakukan oleh pengelola perusahaan seperti merencanakan, mengorganisasikan, mengkoordinir dan mengendalikan operasi untuk mencapai tujuan yang telah ditetapkan[1].

Sistem Informasi merupakan sistem yang digunakan untuk menyimpan dan menganalisa data-data yang sudah diinput serta menghasilkan suatu laporan yang merepresentasikan data-data yang telah diinput. Sistem Informasi Manajemen adalah pengembangan dan penggunaan sistem-sistem informasi yang efektif dalam organisasi-organisasi (Kroenke, David, 1989). Sistem Informasi Manajemen merupakan serangkaian sistem informasi yang terkoordinir untuk menjadikan data menjadi informasi melalui serangkaian cara untuk meningkatkan produktivitas yang sesuai dengan gaya dan sifat atas dasar kriteria mutu yang telah ditetapkan.

2.2.2 Sanggar

Sanggar adalah sarana yang digunakan oleh suatu komunitas untuk melakukan suatu kegiatan. Sanggar Senam merupakan sarana untuk belajar senam seperti *aerobic* dan *body language*.

2.2.3 Website

Website merupakan kumpulan halaman dalam satu domain tentang berbagai informasi agar dapat dibaca dan dilihat oleh pengguna internet, biasanya *website* berisi konten gambar, ilustrasi, *video*, dan teks. *Website* digunakan sebagai sarana informasi, sarana *ecommerce*. *Website* terbagi menjadi 3 jenis yaitu *website* statis, *website* dinamis, dan *website* interaktif. *Website* Statis kontennya tidak berubah sehingga tidak membutuhkan *update* konten berkala dan tidak memerlukan *database* serta dibuat menggunakan HTML dan CSS. *Website* Dinamis berisi konten yang dapat diubah atau *update* berkala, pembuatan Web Dinamis memerlukan komponen HTML, *Javascript*, CSS dan PHP, serta basis data seperti *MySQL* untuk menyimpan data. Contoh web dinamis yaitu *website* berita dan *website* penjualan.

2.2.4 Rekayasa Perangkat Lunak (RPL)

Rekayasa Perangkat Lunak (RPL) merupakan sebuah disiplin ilmu yang berhubungan dengan proses pengembangan perangkat lunak dari tahap perancangan hingga tahap implementasi serta pasca implementasi sehingga siklus hidup perangkat lunak dapat berlangsung secara efisien dan terukur[7]. Adapun aspek yang terlibat dalam penelitian ini yang meliputi:

A. Metode Pengembangan Perangkat Lunak

Metode yang digunakan oleh penulis yaitu metode pengembangan sistem dengan menggunakan metode *waterfall*. Berikut ini adalah penjelasan mengenai *waterfall* antarlain:

1. *Waterfall*

Metode *waterfall* menggambarkan pendekatan yang sistematis dan berurutan pada pengembangan perangkat lunak. Dimulai dengan spesifikasi kebutuhan pengguna kemudian melalui tahap perencanaan, permodelan, konstruksi, serta penyerahan sistem ke pengguna yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan[2]. Tahapan-tahapan dalam metode *waterfall*:

a) *Requirement*

Pada tahap ini dibutuhkan komunikasi untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak, informasi didapat melalui wawancara, diskusi atau survei langsung kemudian dianalisis untuk mendapat data yang dibutuhkan oleh pengguna.

b) *Design*

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras dan sistem persyaratan serta membantu mendefinisikan arsitektur sistem secara menyeluruh.

c) *Implementation*

Pada tahap ini, sistem pertama kali dikembangkan dengan program yang disebut unit yang terintegrasi pada tahap selanjutnya, setiap unit dikembangkan dan diuji untuk fungsionalitas atau *unit testing*.

d) *Verification*

Pada tahap ini dilakukan verifikasi dan pengujian pada sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dikategorikan ke *unit testing* pada modul tertentu untuk melihat bagaimana sistem bereaksi ketika semua modul terintegrasi, dan penerimaan pengujian untuk melihat apakah semua kebutuhan *users* puas.

e) *Maintenance*

Tahap terakhir dalam *waterfall*, perangkat lunak yang sudah jadi dijalankan dan dilakukan pemeliharaan dengan memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

Kelebihan Metode Waterfall

- a. Pelaksanaan dilakukan secara bertahap menghasilkan kualitas sistem yang baik.
- b. Meminimalisasi kesalahan yang mungkin akan terjadi karena model proses tahap pengembangan satu demi satu.
- c. Setiap tahap harus terselesaikan sebelum melaksanakan tahap berikutnya, sehingga dokumen pengembangan sistem sangat terorganisir.

Kekurangan Metode Waterfall

- a. Proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk sehingga memerlukan manajemen yang baik.
- b. Jika terdapat kesalahan kecil yang tidak terselesaikan pada awal pengembangan yang berakibat pada tahap selanjutnya maka akan menjadi masalah besar.
- c. Waktu pengembangan lama

B. Metode Pengujian Sistem

Metode pengujian sistem yang digunakan yaitu *Black Box Testing*. *Black Box Testing* merupakan metode untuk menguji sistem dari segi fungsional apakah masukan yang diterima dan keluaran sudah sesuai dengan spesifikasi yang dibutuhkan tanpa menguji desain dan kode program.[7]

C. Unified Modeling Language (UML)

UML atau *Unified Modelling Language* merupakan metode pemodelan berupa visual sebagai sarana untuk membuat *software* berorientasi objek. UML adalah bahasa visual untuk pemodelan bahasa berorientasi objek serta perancangan dan pendokumentasian *software*[8]. Diagram yang terdapat pada UML antara lain:

a) Usecase Diagram

Use Case merupakan sebuah teknik yang digunakan untuk memodelkan dan menyatakan layanan/fungsi yang disediakan sistem ke pengguna. *Use Case* menjelaskan apa yang terjadi ketika seseorang berinteraksi dengan sistem. Berikut adalah simbol-simbol dari *usecase diagram*[11] :

Tabel 2. 2 Simbol Usecase Diagram

| No | Gambar | Nama | Keterangan |
|----|---|-----------------|---|
| 1 |  | <i>Actor</i> | Aktor merupakan peran yang pengguna mainkan ketika berinteraksi dengan sistem. |
| 2 |  | <i>Use Case</i> | Fungsionalitas yang disediakan sistem sebagai unit untuk saling bertukar pesan antar <i>actor</i> . |

| | | | |
|---|---|--------------------|---|
| 3 |  | <i>Association</i> | Komunikasi antara <i>actor</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor |
| 4 |  | <i>Extend</i> | Ekstensi merupakan relasi <i>usecase</i> tambahan yang dapat berdiri sendiri walau tanpa <i>usecase</i> tambahan ke sebuah <i>usecase</i> |
| 5 |  | <i>Include</i> | Relasi <i>usecase</i> tambahan yang memerlukan <i>usecase</i> ini untuk menjalankan fungsinya ke sebuah <i>usecase</i> |
| 6 |  | <i>System</i> | Menampilkan sistem secara terbatas. |

b) *Sequence Diagram*

Sequence Diagram merupakan *tool* dalam pengembangan sistem informasi secara *object-oriented* untuk menampilkan interaksi antar objek (Nofriyadi Jurdam, 2014). Diagram ini menggambarkan kolaborasi antar sejumlah objek untuk menunjukkan rangkaian pesan antar objek dan diatur berdasarkan waktu terjadinya. Berikut ini adalah simbol-simbol dari *Sequence Diagram* antara lain[8]:

Tabel 2. 3 Simbol – Simbol *Sequence Diagram*

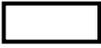
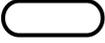
| No. | Simbol | Nama | Keterangan |
|-----|---|-----------------|---|
| 1. |  | <i>LifeLine</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi. |

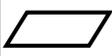
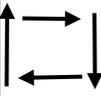
| No. | Simbol | Nama | Keterangan |
|-----|---|-------------------|---|
| 2. |  | <i>Activation</i> | Menggambarkan sebuah objek yang melakukan sebuah aksi |
| 3. |  | <i>Actor</i> | Menggambarkan pengguna. |
| 4. |  | <i>Message</i> | Komunikasi antar <i>objek</i> yang berisi informasi tentang aktivitas yang terjadi. |
| 5. |  | <i>Object</i> | <i>Instance</i> dari sebuah <i>class</i> yang tersusun secara horizontal |

2.2.5. Flowchart

Flowchart adalah diagram yang menyatakan aliran proses dengan menggunakan notasi bidang-bidang geometri untuk merepresentasikan langkah-langkah kegiatan beserta urutannya dengan menghubungkan masing-masing langkah tersebut menggunakan tanda panah. Simbol-simbol *Flowchart* yang biasanya dipakai adalah simbol-simbol *Flowchart* berikut ini [10]:

Tabel 2. 4 Simbol-simbol *Flowchart*

| No. | Simbol | Nama | Keterangan |
|-----|---|---------------------------|--|
| 1. |  | Proses/ <i>Processing</i> | Digunakan untuk menunjukkan pengolahan yang akan dilakukan dalam komputer |
| 2. |  | <i>Desicion</i> | Digunakan untuk memilih proses mana yang akan dilakukan berdasarkan kondisi tertentu |
| 3. |  | Terminal | Digunakan untuk memulai atau mengakhiri program |

| | | | |
|-----|---|---------------------------|--|
| 4. |  | <i>Predefined Process</i> | Digunakan untuk mempersiapkan penyimpanan yang sedang/akan digunakan |
| 5. |  | <i>Input/Output</i> | Digunakan untuk menyatakan <i>input</i> dan <i>output</i> tanpa melihat jenisnya |
| 6. |  | <i>Document</i> | Digunakan untuk menyatakan masukan dan keluaran yang berasal dari dokumen |
| 7. |  | <i>On Page Connector</i> | Digunakan untuk menyatakan sambungan dari proses yang satu ke proses berikutnya dihalaman yang sama |
| 8. |  | <i>Off Page Connector</i> | Digunakan untuk menyatakan sambungan dari proses yang satu ke proses berikutnya dihalaman yang berbeda |
| 9. |  | <i>Connecting Line</i> | Digunakan untuk menghubungkan simbol yang satu dengan yang lainnya untuk menyatakan alur suatu proses |
| 10. |  | <i>Manual Operation</i> | Digunakan untuk menunjukkan pengolahan yang tidak dilakukan oleh computer |
| 11. |  | <i>Manual Input</i> | Digunakan untuk menginputkan data secara manual dengan <i>keyboard</i> |

| | | | |
|-----|---|--------------------|--|
| 12. |  | <i>Preparation</i> | Digunakan untuk menunjukkan proses inialisasi/pemberian harga awal |
|-----|---|--------------------|--|

2.2.6 Rekayasa Web

Rekayasa Web adalah subdisiplin dari rekayasa perangkat lunak yang membantu menyediakan metodologi untuk merancang, mengembangkan, memelihara dan melibatkan aplikasi Web. Rekayasa Web merupakan sebuah aplikasi yang menggunakan pendekatan sistematis, disiplin dan terukur untuk pengembangan, operasi dan pemeliharaan aplikasi berbasis Web (Janner Simarmata, 2010).

2.2.7 Pemrograman Berorientasi Objek (PBO)

Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek berisi data dan operasi yang diberlakukan terhadapnya. Pemrograman berorientasi objek atau *Object Oriented Programming* (OOP) merupakan suatu cara berpikir serta logika untuk menghadapi masalah-masalah yang akan coba diatasi dengan bantuan komputer. OOP memiliki konsep dasar yang harus dipahami seperti berikut ini [12]:

a. Objek

Objek merupakan sesuatu bersifat abstrak yang mewakili dunia nyata, seperti manusia, hewan, benda, konsep, aliran, dan lain-lain. Objek mampu menyimpan informasi dan mempunyai operasi yang dapat diterapkan pada objeknya.

b. Kelas

Kumpulan objek yang memiliki kemiripan perilaku (*method*), ciri atau karakteristik (*property*). Suatu kelas dapat diwariskan ke kelas yang baru.

c. Metode (*Method*)

Method merupakan perwujudan aksi atau tindakan dari dunia nyata di dalam pemrograman komputer atau perilaku dari objek. Sebuah kelas boleh memiliki lebih dari satu metode.

d. Atribut (*Attribute*)

Atribut merupakan variabel global yang dimiliki oleh sebuah kelas dapat berupa nilai atau elemen-elemen data. Atribut sebaiknya bersifat privat guna menjaga konsep enkapsulasi. Atribut dimiliki secara individual oleh suatu objek seperti nama, berat dan sebagainya.

- e. Enkapsulasi (*Encapsulation*)
Enkapsulasi merupakan pembungkusan atribut data dan operasi-operasi pada objek. Enkapsulasi digunakan untuk menyembunyikan implementasi sehingga objek lain tidak mengetahui cara kerjanya.
- f. Abstraksi (*Abstraction*)
Abstraksi merupakan prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi bentuk yang sederhana dengan mengabaikan aspek yang tidak sesuai dengan permasalahan.
- g. Pewarisan (*Inheritance*)
Pewarisan merupakan proses yang memungkinkan objek mewarisi sebagian atau seluruh definisi dari objek lain sebagai bagian dari dirinya.
- h. Antarmuka (*Interface*)
Antarmuka memiliki persamaan dengan kelas, namun tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi serta dapat diimplementasikan oleh kelas lain. Sebuah kelas dapat mengimplementasikan lebih dari satu antarmuka dimana kelas mendeklarasikan metode pada antarmuka yang dibutuhkan dan mendefinisikan isinya pada kode program kelas tersebut.
- i. *Reusability*
Reusability atau pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan yang melibatkan objek tersebut.
- j. Komunikasi Antar Objek
Komunikasi antar objek dilakukan melalui pesan yang dikirim dari satu objek ke objek lainnya.
- k. Generalisasi dan Spesifikasi
Generalisasi dan Spesifikasi menunjukkan hubungan antara kelas dan objek yang umum dengan kelas dan objek yang khusus.
- l. *Package*
Package merupakan sebuah kemasan yang dapat digunakan untuk mengelompokkan kelas-kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam package yang berbeda.
- m. Polimorpisme (*Polymorphism*)
Polimorpisme merupakan kemampuan objek untuk digunakan pada banyak tujuan yang berbeda dengan nama sama yang dapat menghemat baris program.

2.2.8 *Framework CodeIgniter*

Framework CodeIgniter merupakan framework aplikasi web untuk membangun aplikasi PHP dinamis menggunakan konsep *Model View Controller*. *Framework CodeIgniter* termasuk *framework* tercepat dibandingkan dengan *framework* lainnya, *codeigniter* menyediakan *library* yang dapat mempermudah pengembangan aplikasi web.[12]

2.2.9 **Basis data**

Basis data merupakan sistem terkomputerisasi yang bertujuan untuk memelihara data yang telah diolah dan mempercepat proses. Basis data sebagai media untuk menyimpan data agar dapat diakses dengan mudah dan cepat yang meliputi memasukan, menyimpan, mengambil kembali data dan membuat laporan berdasarkan data yang telah disimpan.

A. **DBMS (*Database Management System*)**

DBMS merupakan perangkat lunak yang memudahkan organisasi untuk memusatkan data, mengelola data secara efisien dan menyediakan akses bagi program aplikasi. DBMS juga merupakan sekumpulan komponen untuk menetapkan, membangun dan menggerakkan suatu *database*. DBMS memiliki beberapa fasilitas sebagai berikut [13] :

1. DDL atau *Data Definition Language* merupakan metode *Query SQL* yang berguna untuk mendefinisikan data pada sebuah *database* dengan *Query* sebagai berikut :
 - a) *Create* digunakan untuk pembuatan *database* dan tabel dengan menuliskan perintah berikut ini :


```
CREATE DATABASE [nama_database];
CREATE TABLE [nama_tabel];
```
 - b) *Alter* digunakan untuk perubahan struktur tabel seperti menambah *field* (*add*), mengganti nama *field* (*change*), mengganti nama (*rename*) dan menghapus (*drop*) dengan menuliskan perintah sebagai berikut :


```
ALTER TABLE [nama_tabel] ADD COLUMN
[nama_kolom] [tipe_data];
ALTER TABLE [nama_tabel] CHANGE [kolom_lama]
[kolom_baru] [tipe_data_baru];
ALTER TABLE [nama_tabel_lama] RENAME TO
[nama_tabel_baru];
ALTER TABLE [nama_tabel] DROP COLUMN
[nama_kolom_yang_akan_dihapus];
```

- c) *Drop* digunakan untuk menghapus *database* dan tabel dengan menuliskan perintah sebagai berikut :
 DROP DATABASE nama_database;
 DROP TABEL nama_tabel;
- 2. DML atau *Data Manipulation Language* merupakan metode *Query* untuk melakukan manipulasi dan pengambilan *database* yang dapat dilakukan jika DDL telah terjadi dengan *Query* sebagai berikut :
 - a) *Insert* yang digunakan untuk penginputan data pada tabel *database* dengan penulisan perintah sebagai berikut :
 INSERT INTO [nama_tabel] VALUES (isifield1, isifield2, isifield3);
 - b) *Update* digunakan untuk pengubahan data pada tabel *database* dengan penulisan perintah sebagai berikut :
 UPDATE [nama_tabel] SET FIELD = [isi_data] WHERE [Kondisi];
 - c) *Delete* digunakan untuk penghapusan data pada tabel *database* dengan penulisan perintah sebagai berikut :
 DELETE FROM [nama_tabel] WHERE [kondisi];

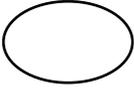
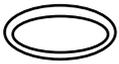
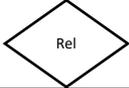
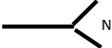
B. SQL

Dalam bahasa SQL (*Structures Query Language*) informasi tersimpan dalam tabel-tabel yang seraca logika merupakan struktur dua dimensi terdiri dari baris (*row* atau *record*) dan kolom (*column* atau *field*), sedangkan *database* terdiri dari beberapa tabel. Untuk manajemen *database MySQL* menggunakan aplikasi *phpMyAdmin* yang memiliki fitur yang sering digunakan seperti membuat, mengubah, atau menghapus *database*, *tables*, *fields* ataupun *rows*, serta dapat digunakan untuk mengeksekusi perintah SQL dan mengatur *user* dan hak akses [14].

C. ERD

ERD atau *Entity Relationship Diagram* digunakan untuk pemodelan basis data relational dalam bentuk gambar atau simbol yang mengidentifikasi tipe dari entitas dalam suatu sistem yang diuraikan dalam data dengan atributnya dan menjelaskan relasi antara atribut tersebut. Simbol ERD dapat dilihat pada table berikut [7]:

Tabel 2. 5 Simbol-simbol ERD

| No | Simbol | Nama | Keterangan |
|----|---|---|--|
| 1 |  | Atribut | Atribut adalah <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan merupakan karakteristik pada entitas |
| 2 |  | Entitas/ <i>Entity</i> | Entitas merupakan objek yang mewakili sesuatu dalam dunia nyata dan dapat dibedakan antara satu dengan yang lainnya (<i>unique</i>). Setiap entitas memiliki beberapa atribut berupa fisik (manusia), Abstrak (pekerjaan, mata kuliah) dan kejadian (penjualan, peminjaman) yang mendeskripsikan karakteristik dari objek tersebut |
| 3 |  | Atribut Multinilai | Atribut Multinilai/ <i>Multivalve</i> adalah <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu. |
| 4 |  | Relasi | Relasi menyatakan hubungan antara dua atau beberapa entitas. |
| 5. |  | Atribut Kunci Primer <i>/Key Attribute</i> | Field yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> berupa <i>id</i> , dapat lebih dari satu kolom |
| 6. |  | Asosiasi | Asosiasi/ <i>association</i> merupakan penghubung antara relasi dan entitas |

| | | | |
|----|---|--|--|
| | | | dengan kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Misalkan ada kardinalitas 1 ke N (<i>one to many</i>) yang menghubungkan entitas A ke entitas B |
| 7. |  | Atribut Turunan <i>/Derived Attribute</i> | Atribut turunan menggambarkan atribut yang merupakan turunan dari atribut lainnya |

2.2.9 Hypertext Preprocessor (PHP)

PHP merupakan salah satu bahasa pemrograman berbasis web. PHP adalah Bahasa *scripting* yang menyatu dengan HTML. Perintah PHP atau *Hypertext Preprocessor* dilaksanakan server kemudian hasilnya ditampilkan pada komputer *client*. PHP juga merupakan HTML *embedded* yaitu sintaks PHP dituliskan bersama dengan sintaks HTML. Bahasa pemrograman PHP bersifat *open source* yang bisa diunduh gratis dan *server side* yang artinya semua sintaks sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke *browser* hanya hasilnya saja. Kelebihan bahasa pemrograman PHP sebagai berikut [14]:

1. Keamanan
PHP menyediakan 3 jenis autentikasi *user*, yaitu http autentikasi, penggunaan *cookies*, dan penggunaan *session*.
2. Integritas dengan *Database*
PHP mendukung integritas, kecepatan dan efisiensi akses ke *database* seperti *MySQL*, *PostgreSQL*, *Oracle*, *SQLite* dan lain-lain.
3. *Cross-platform*
PHP mendukung berbagai jenis sistem operasi seperti *Linux*, *Microsoft Windows*, *Mac OS* dan lain-lain.
4. Reliabilitas
Dukungan dokumentasi yang lengkap, aman dan banyak komunitas *helpdesk* untuk membantu para pengembang web sistem yang menggunakan PHP.
5. Harga
PHP berada dalam lisensi GPL (*GNU Public Licence*) artinya bebas digunakan dan didistribusikan serta gratis.

6. Kemudahan bermigrasi

Banyaknya dukungan terhadap PHP sehingga PHP terus menerus dikembangkan dengan memperbaiki kinerja dan menambah fitur-fitur baru.

2.2.10. Telegram

Telegram adalah aplikasi *messaging* yang berfokus pada kecepatan, keamanan, sederhana dan dapat diunduh secara gratis. Telegram dapat digunakan untuk mengirim pesan, foto, *video*, dan beberapa jenis file (*doc*, *zip*, *mp3*, dll) serta dapat membuat grup hingga 5000 orang atau *channel* untuk *broadcasting* pada khalayak terbatas. Salah satu fitur telegram yang berbeda dengan aplikasi *messenger* lainnya yaitu Telegram menyediakan API (*Application Programming Interface*) yang terbuka 100% untuk publik yang ingin mengembangkan aplikasi menggunakan API Telegram. Telegram mempunyai 2 jenis API untuk *developer* yaitu [16] :

A. *Bot API*

Memungkinkan *developer* untuk menghubungkan Bot dengan sistem Telegram. Telegram *Bots* adalah akun khusus yang tidak memerlukan nomor telepon tambahan dalam pengaturannya yang berfungsi sebagai antarmuka untuk tempat berjalannya kode pada suatu server.

B. *Telegram API*

Memungkinkan *developer* untuk membangun sendiri Telegram *clients* yang diinginkan dan terbuka 100% untuk semua *developer* yang ingin membuat aplikasi dengan *platform* Telegram.

~HALAMAN INI SENGAJA DIKOSONGKAN~