

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu ini menjadi satu acuan penulis dalam membuat laporan akhir sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan beberapa jurnal yang terkait dengan judul laporan akhir penulis.

Pada penelitian sebelumnya yang dilakukan (Wedowati, 2018) dalam jurnal yang berjudul **“Aplikasi Simpan Pinjam Koperasi Pegawai Republik Indonesia IAIN Raden Fatah Palembang Berbasis Web”**. Permasalahannya ialah semua proses pendataan masih dilakukan secara manual (menggunakan buku dan microsoft excel), mulai dari pendataan anggota, pendataan simpanan, pendataan pinjaman, pendataan angsuran dan dalam pembuatan laporan, karena itu koperasi simpan pinjam pada Koperasi Pegawai Republik Indonesia (KPRI) IAIN Raden Fatah Palembang mengalami masalah pada beberapa bagian yaitu dalam pengaksesan data atau informasinya akan lambat, bahkan data atau informasi belum tentu terjamin akurasi karena sebagian data tidak tersimpan dengan baik dan sering terjadinya kehilangan data. Melihat kebutuhan yang ada merasa diperlukan adanya sebuah Aplikasi berbasis web yang mampu mengolah data simpan pinjam koperasi sehingga mampu menyampaikan informasi simpan pinjam yang dibutuhkan oleh anggota.

Pada penelitian sebelumnya yang dilakukan (Romadhon, 2019) dalam jurnal yg berjudul **“Perancangan Website Aplikasi Simpan Pinjam Menggunakan Framework Codeigniter pada Koperasi Bumi Sejahtera Jakarta”**. Permasalahannya ialah pada sistem administrasi yang terdapat pada koperasi bumi sejahtera sebelumnya sudah menggunakan aplikasi, walaupun sudah menggunakan aplikasi akan tetapi masih berjalan secara manual, dikarenakan sistem yang ada tidak dapat mendukung aktifitas yang dilakukan

oleh koperasi bumi sejahtera, terlebih lagi permasalahan pada program yang berjalan yaitu program yang digunakan sudah tidak diperpanjang dengan pihak aplikasi, dikarenakan program yang berjalan saat ini kurang berfungsi secara optimal sesuai dengan kebutuhan pada koperasi bumi sejahtera, banyak feature yang berada aplikasi tersebut tidak dipergunakan, serta biaya yang digunakan untuk memperpanjang license dan service kontrak cukup besar setiap tahunnya, dan juga jika terjadi masalah pada program akan dikenakan biaya pemanggilan oleh pihak aplikasi.

Oleh karena itu pihak koperasi bumi sejahtera tidak menggunakan aplikasi tersebut dan pada proses pendataan dilakukan secara manual menggunakan aplikasi excel yang tentunya memerlukan waktu untuk mengolah data tersebut dikarenakan tidak terpusat pada basis data yang akan diolah, sehingga akan mempengaruhi waktu untuk membuat laporan para anggotanya.

Pada penelitian sebelumnya yang dilakukan (Prasadha, 2020) dalam jurnal yang berjudul **“Perancangan Aplikasi Koperasi Simpan Pinjam Berbasis Web Dengan Metode Waterfall Menggunakan Framework Codeigniter (Studi Kasus : Koperasi Bhakti Sedana Simpan Pinjam)”**. Permasalahannya pada Aplikasi yang ada di KOBINA masih menggunakan metode manual sehingga masih memiliki berbagai kekurangan dan kendala yang dihadapi pada saat pencarian data simpanan dan pinjaman membutuhkan waktu yang lama. Dalam KOBINA tidak lepas dari proses pembuatan laporan yang terkait dengan proses yang dijalankan dalam koperasi ini. Laporan yang dihasilkan dalam koperasi ini meliputi laporan dalam hal simpan pinjam yang meliputi laporan data anggota, simpanan, pinjaman uang, dan sisa angsuran uang anggota koperasi.

Pada penelitian sebelumnya yang dilakukan (Novianto, 2019) dalam jurnal yang berjudul **“Aplikasi Simpan Pinjam Koperasi Karyawan Universitas Kristen Petra Berbasis Website”**. Pada hasil penelitian ini ditarik kesimpulan bahwa sistem yang dibuat telah mampu mengajukan permohonan secara online. Sistem telah mampu melakukan perhitungan simpanan berkala, simpanan swadaya berjangka, dan pinjaman., Sistem telah mampu membantu pihak koperasi dalam melihat kondisi keuangan anggota dalam hal simpan dan pinjam.

Pada penelitian sebelumnya yang dilakukan (Salsabila, 2020) dalam jurnal yang berjudul “**Aplikasi Simpan Pinjam di Koperasi Pendidikan Wonogiri**”. Permasalahannya ialah setiap kegiatan masih dilakukan secara manual atau belum adanya komputerisasi sehingga dapat menimbulkan masalah dalam input, proses maupun output. Proses transaksi seperti penghitungan tabungan simpanan tiap anggota, penghitungan pinjaman angsuran, dilakukan dan dicatat secara manual dan belum tertata rapi. Kondisi ini menimbulkan permasalahan yang kemungkinan terjadi adanya kesalahan penulisan dan pencatatan serta lamanya waktu dalam proses pencarian data. Cara yang tepat untuk mengatasi permasalahan ini dengan menggunakan komputerisasi. Oleh karena itu dibuatlah suatu Aplikasi berbasis web yang terintegrasi untuk menggantikan pekerjaan yang awalnya masih secara manual menjadi terkomputerisasi. Aplikasi ini diharapkan dapat mempermudah dan mengatur jalannya koperasi untuk kedepannya

Judul laporan akhir penulis mengimplementasikan sistem yang akan dibuat didasari dengan penelitian-penelitian terdahulu untuk dapat diterapkan pada penelitian ini.

2.2 Koperasi

Koperasi adalah badan usaha yang beranggotakan orang seorang atau badan koperasi dengan melandaskan kegiatannya berdasarkan prinsip koperasi sekaligus sebagai gerakan ekonomi rakyat yang berlandaskan atas asas kekeluargaan. Pada umumnya sistem yang diterapkan dalam sebuah koperasi sama. Saat ini masih banyak koperasi yang melakukan pengolahan datanya secara manual dengan segala kekurangannya padahal kemajuan teknologi informasi dapat membantu pengolahan yang semula manual menjadi terkomputerisasi dengan segala kelebihanannya. Kekurangan pengolahan data pada sebuah koperasi yang dilakukan secara manual diantaranya kesalahan dalam pencatatan transaksi simpan pinjam, lambatnya proses pencarian data dan pengolahan data transaksi simpan pinjam, adanya duplikasi data serta penyediaan laporan memerlukan waktu yang cukup lama.

Pada hakekatnya koperasi merupakan suatu lembaga ekonomi yang diperlukan dan penting. Koperasi merupakan usaha bersama yang berlandaskan asas kekeluargaan untuk meningkatkan kesejahteraan anggotanya. Untuk menyelaraskan dengan perkembangan keadaan ketentuan tentang perkoperasian di Indonesia telah diperbaharui, yaitu dengan UU perkoperasian No. 25/1992, yang dimaksud dengan koperasi adalah “Badan usaha yang beranggotakan orang seorang atau koperasi, sekaligus sebagai gerakan ekonomi rakyat yang berdasar atas asas kekeluargaan” (Sukamdiyo, 1996).

2.3 Simpan Pinjam

Koperasi simpan pinjam atau koperasi kredit adalah yang bergerak dalam lapangan usaha pembentukan modal melalui tabungan para anggotanya dengan cara yang mudah, murah, cepat, dan tepat untuk tujuan produktivitas dan kesejahteraan (Widiyanti, 1998). Koperasi simpan pinjam mendapat modal dari berbagai simpanan, pinjaman, penyisaan dari hasil usaha termasuk cadangan serta sumber-sumber lainnya. simpanan-simpanan tersebut diantaranya adalah sebagai berikut :

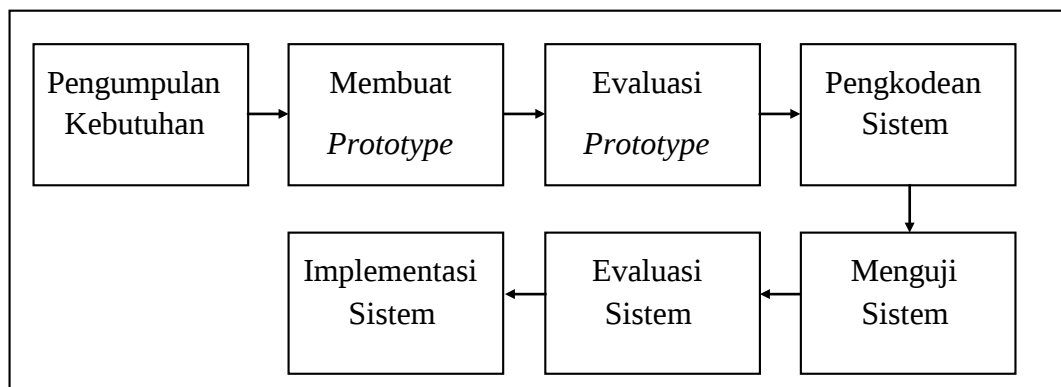
1. Simpanan Pokok yaitu simpanan yang diberikan anggota pada awal setoran dan menjadi simpanan yang berbentuk permanen.
2. Simpanan Wajib yaitu simpanan yang dapat diambil sewaktu-waktu dalam jangka waktu tertentu.
3. Simpanan Sukarela adalah simpanan yang diterima bukan dari anggota koperasi itu sendiri.

2.4 Metode Pengembangan Sistem dengan Metode Prototyping

Metode yang digunakan dalam perancangan sistem ini yaitu prototyping. Hanif (2007) mengatakan *prototyping* sebagai proses iteratif dalam pengembangan sistem dimana kebutuhan diubah ke dalam sistem yang bekerja yang secara terus menerus diperbaiki melalui kerjasama antara pengguna dan analisa.

Menurut Sri (2016), *prototyping* merupakan teknik pengembangan sistem yang menggunakan *prototype* untuk menggambarkan sistem, sehingga pengguna atau pemilik sistem mempunyai gambaran pengembangan sistem yang akan dilakukannya. Dengan teknik *prototyping*, pengembang bisa membuat *prototype* terlebih dahulu sebelum mengembangkan sistem yang sebenarnya.

Berdasarkan pendapat para ahli diatas dapat ditarik kesimpulan bahwa *Prototype* didefinisikan sebagai satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam membentuk yang telah selesai. Semua rancangan diagram atau model yang dibuat tidak diharuskan telah sempurna dan final dalam *prototype*. Tujuan utama dari penyiapan rancangan adalah sebagai alat bantu dalam memberi gambaran sistem seperti materi dan menu yang perlu dimasukkan dalam *prototype* yang akan dikembangkan. Adapun tahapan-tahapan dalam *prototype* adalah sebagai berikut:



Gambar 2.1 Tahap Metode *Prototyping*

1. Pengumpulan Kebutuhan

Pengguna dan pengembang sama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan dan garis besar sistem yang akan dibuat.

2. Membuat *Prototype*

Membangun *Prototype* dengan perancangan sementara yang berfokus pada penyajian kepada pengguna. Contohnya dengan membuat *input* dan format *output*.

3. Evaluasi *Prototype*

Evaluasi ini dilakukan oleh pengguna apakah *prototype* yang sudah dibangun sesuai dengan keinginan pengguna. Jika sudah sesuai maka langkah 4 akan diambil. Jika tidak *prototype* direvisi dengan mengulangi langkah 1,2,dan 3.

4. Pengkodean Sistem

Dalam tahap ini *prototype* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

5. Menguji Sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus di uji dahulu sebelum digunakan.

6. Evaluasi Sistem

Pengguna mengevaluasi apakah sistem yang dibangun sudah sesuai dengan yang diharapkan. Jika telah selesai maka dilanjutkan dengan melakukan langkah ke 7. Jika belum maka mengulangi langkah 4 dan 5.

7. Implementasi Sistem

Implementasi sistem atau menggunakan sistem yang berarti sistem telah diuji dan siap untuk digunakan.

2.5 Perancangan Sistem

Ada banyak pendapat mengenai pendefinisian desain sistem. Sebagai contoh, dalam buku Hartono (1989) :

1. Menurut Jhon Bruch & Gary Grudnitski, desain sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.
2. Menurut Robert J. Verzello/Jhon Reuter III, tahap setelah analisis dari siklus pengembangan sistem : pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi; menggambarkan bagaimana suatu sistem dibentuk.

3. Dan menurut George M. Scott pada buku, desain sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan; tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem.

Dengan demikian desain sistem dapat disimpulkan sebagai pendefinisian dari kebutuhan-kebutuhan fungsional untuk mempersiapkan rancang bangun implementasi yang berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen terpisah kedalam satu kesatuan yang utuh dan berfungsi.

2.6 *Data Flow Diagram (DFD)*

Kristanto (2008) mengatakan bahwa:

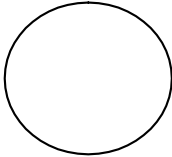
Data Flow Diagram merupakan suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Sukanto dan Shalahuddin (2014) mengatakan bahwa:

Data Flow Diagram atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengatur dari masukan (input) dan keluaran (output). DFD tidak sesuai untuk memodelkan sistem yang menggunakan pemrograman berorientasi objek.

Sukanto dan Shalahuddin (2014) mengatakan bahwa notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

NOTASI	KETERANGAN
	Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja
	<i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel basis data (<i>Entity Relationship Diagram (ERD)</i>, <i>Conceptual Data Model (CMD)</i>, <i>Physical Data Model (PDM)</i>) Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda
	Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai atau berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan. Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda

	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.
---	--

Menurut Sukanto dan Shalahuddin (2014), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga Context Diagram DFD

Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara system yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil breakdown DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di breakdown menjadi DFD Level 2. Modul mana saja yang harus di breakdown lebih detail tergantung pada tingkat kedetilan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di breakdown lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di breakdown.

4. Membuat DFD Level 3 dan seterusnya.

DFD Level 3, 4, 5 dan seterusnya merupakan breakdown dari modul pada DFD Level di atasnya. Breakdown pada level 3, 4 dan 5 dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level.

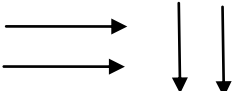
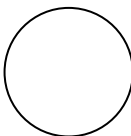
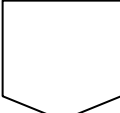
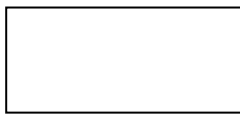
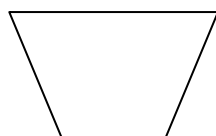
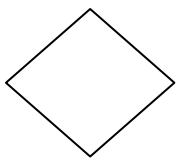
2.7 Flowchart

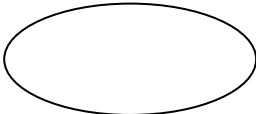
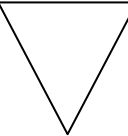
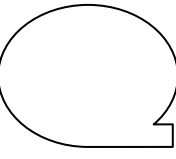
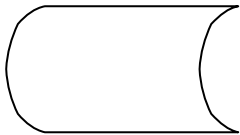
Kadir (2017) menjelaskan diagram alir dapat dikelompokkan ke dalam dua kategori, yaitu diagram alir sistem (system *flowchart*) dan diagram alir program (program *flowchart*).

1. Diagram alir sistem biasa digunakan oleh analis sistem untuk menggambarkan aliran data atau struktur file di dalam suatu sistem.
2. Diagram alir program biasa dipakai oleh pemrogram untuk menjelaskan langkah-langkah di dalam suatu program.

Adapun simbol-simbol yang digunakan pada diagram alir adalah sebagai berikut:

Tabel 2.3 Tabel *Flowchart*

NO.	SIMBOL	KETERANGAN
1.		Simbol arus/ <i>flow</i> , berfungsi untuk menyatakan jalannya arus suatu proses
2.		Simbol <i>connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
3.		Simbol <i>offline connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
4.		Simbol <i>process</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang dilakukan oleh computer
5.		Simbol <i>manual</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer
6.		Simbol <i>decision</i> , berfungsi untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya/tidak

7.		Simbol <i>teminal</i> , berfungsi untuk menyatakan permulaan atau akhir suatu program
8.		Simbol <i>predefined process</i> , berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
9.		Simbol <i>keying operation</i> , berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i>
10.		Simbol <i>offline-storage</i> , berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
11.		Simbol <i>predefined process</i> , berfungsi sebagai permulaan sub program dan proses menjalankan sub program
12.		Simbol <i>input/output</i> , berfungsi untuk menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
13.		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis
14.		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>
15.		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)

Sumber : (Rosaly, 2019).

2.8 Basis Data (*Database*)

Basis Data terdiri dari kata basis dan data. Basis dapat diartikan sebagai markas atau gudang. Sedangkan data adalah catatan atas kumpulan fakta dunia nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasinya. Himpunan kelompok data yang saling terhubung dan diorganisasikan sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah. Kumpulan data dalam bentuk file/tabel/arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronis, untuk kemudahan dalam pengaturan, pemilahan, pengelompokan dan pengorganisasian data sesuai tujuan.

Database mempunyai 8 operasi dasar diantaranya adalah *Create database*, *Drop database*, *create table*, *Drob table*, *Insert*, *Read*, *Update* dan *Delete*. Sebuah database menyimpan data di dalam tabel, dimana setiap tabel memiliki baris dan kolom. Data merupakan fakta atau angka. Tabel berisi tentang berbagai jenis hal, sebagai contoh data barang, data dosen. Setiap baris pada tabel yang berisi data yang disebut record dan setiap kolom yang menyimpan karakteristik umum untuk semua baris disebut *fields*.

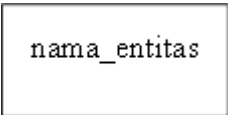
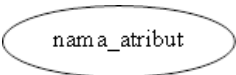
Field adalah kumpulan dari karakter yang membentuk satu arti, maka jika terdapat *field* misalnya seperti NomorBarang atau NamaBarang, maka yang dipaparkan dalam *field* tersebut harus yang berkaitan dengan nomor barang dan nama barang. Atau definisi *field* yang lainnya yaitu tempat atau kolom yang terdapat dalam suatu tabel untuk mengisi nama-nama (data) *field* akan di isikan.

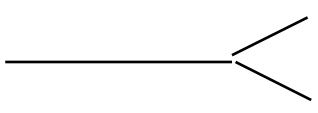
“*Record* adalah kumpulan *field* yang sangat lengkap, dan biasanya dihitung dalam satuan baris. Tabel adalah kumpulan dari beberapa *record* dan juga *field*. File adalah terdiri dari *record-record* yang menggambarkan dari satu kesatuan data yang sejenis. Misalnya seperti file nama barang berisikan data tentang semua nama barang yang ada” (Rachmadi:2020).

2.9 Entity Relationship Diagram (ERD)

Sukanto dan Shalahuddin (2014) “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.” Komponen-komponen pembentuk ERD dapat di lihat pada tabel Tabel 2.3 di bawah ini.

Tabel 2.3 Simbol-simbol *Entity Relationship Diagram* (ERD)

Simbol	Deskripsi
Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja

Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.
--	---

Sumber : Sukamto dan Shalahuddin (2018:50)

2.10 SQL (*Structured Query Language*)

Sebuah bahasa untuk mengakses data didalam sebuah *database* relasional. Berikut ini akan diuraikan beberapa pengertian *Structured Query Language* (SQL) menurut para ahli.

Menurut Utami & Sukrisno (2008) “SQL (*Structured Query Language*) pada dasarnya adalah bahasa komputer standar yang ditetapkan untuk mengakses dan memanipulasi sistem *database*”.

Sedangkan menurut Priyadi (2014) “*Structured Query Language*(SQL) adalah bahasa pemrograman khusus untuk mengirim pencarian data (*query*) ke *database*”.

Berdasarkan penjelasan diatas dapat disimpulkan bahwa, *Structured Query Language* (SQL) adalah bahasa standar yang digunakan untuk mengakses atau mengirim pencarian data ke database.

2.11 MySQL

Enterprise (2016), “MySQL merupakan server yang melayani database. Untuk membuat dan mengolah *database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah)”. Sedangkan Adi dalam Radillah (2018), “MySQL adalah sebuah program *database server* yang mampu menerima dan

mengirim datanya dengan sangat cepat, multi *user* serta menggunakan perintah standar SQL (*Structure Quered Language*)”.

Berdasarkan pendapat para ahli diatas dapat disimpulkan bahwa, MySQL adalah program database yang digunakan untuk membuat dan mengelola *database* dengan cepat, multi user serta menggunakan *query* (perintah).

2.12 *Framework Codeigniter*

Dalam pembuatan *website* untuk memudahkan dalam pengerjaan bisa menggunakan *framework*. Dalam hal ini penulis menggunakan *framework codeigniter* untuk memudahkan dalam mengelompokkan *code* program. Menurut Novianto (2016) menyatakan bahwa “*Framework* adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *website*, diharuskan mengikuti aturan dari *framework* tersebut”.

Menurut Jubilee (2015) menyatakan bahwa “*Framework* merupakan banyak kode, yang disimpan dalam beberapa file yang terpisah, dan memudahkan dalam penggunaan kode yang digunakan secara berulang-ulang”. Sedangkan Menurut Andriyani (2016) “*CodeIgniter* (CI) merupakan aplikasi open source yang berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis menggunakan PHP”.

Codeigniter memiliki banyak fitur yang membantu para pengembang PHP untuk dapat membuat aplikasi web secara mudah dan cepat. *Codeigniter* mengizinkan para pengembang untuk menggunakan *framework* secara parsial maupun secara keseluruhan. Adapun keunggulan dari *Codeigniter* sebagai berikut:

1. *Codeigniter* adalah *framework* yang bersifat *free* dan *opensource*.
2. *Codeigniter* memiliki ukuran kecil dibandingkan dengan *framework* lain.
3. Aplikasi yang dibuat menggunakan *codeigniter* bisa berjalan cepat.
4. *Codeigniter* menggunakan pola design *Model-View-Controller* (MVC) sehingga satu file tidak terlalu berisi banyak kode.
5. *Codeigniter* dapat diperluas sesuai dengan kebutuhan

6. *Codeigniter* terdokumentasi dengan baik. Informasi tentang pustaka kelas dan fungsi yang disediakan oleh *codeigniter* dapat diperoses melalui dokumentasi yang disertai di dalam paket distribusinya.

2.13 PHP

Abdulloh (2016), "PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses disisi *server*". Sedangkan menurut Putratama (2016), "PHP (*PHP: Hypertext Processor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam HTML".

Berdasarkan pendapat para ahli diatas dapat disimpulkan bahwa, PHP adalah bahasa pemograman berbasis *server-side* yang bisa kita gunakan untuk membuat aplikasi *website* yang ditanamkan kedalam HTML, yang dijalankan di dalam sebuah *webserver*, dan juga bisa menghasilkan tampilan *website* yang dinamis.

2.14 JavaScript

Suatu bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam *website browser* menjadi lebih interkatif. Berikut ini akan diuraikan beberapa pengertian JavaScript menurut para ahli.

Menurut Wahyono (2009) "JavaScript adalah yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML". Sedangkan Menurut Saputra (2012) "Javascript adalah bahasa pemograman yang digunakan untuk membuat website lebih dari dinamis dan interaktif".

Berdasarkan penjelasan diatas dapat disimpulkan bahwa, JavaScript adalah kumpulan skrip yang fungsinya berjalan pada suatu dokumen HTML dan dapat digunakan dalam membuat website agar lebih dinamis.

2.15 Bootstrap

Sebuah framework HTML dan CSS yang berfungsi untuk situs dan aplikasi website. Berikut ini akan diuraikan beberapa pengertian Bootstrap menurut para ahli.

Menurut Ridha (2007) “Twitter Bootstrap adalah sebuah alat bantu (framework) HTML dan CSS untuk membuat sebuah tampilan halaman website yang elegan dan support segala macam device.

Sedangkan Menurut Rivaldi (2015) “Bootstrap merupakan Framework ataupun Tools untuk membuat aplikasi website ataupun situs website responsive secara cepat, mudah dan gratis”.

Berdasarkan teori para ahli diatas dapat disimpulkan bahwa, Bootstrap adalah sebuah alat bantu untuk membuat tampilan halaman website menjadi elegan, cepat,dan mudah.

2.16 XAMPP

Madcoms (2018), “XAMPP adalah sebuah paket kumpulan software yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla dan lain-lain”. Menurut Sidik (2019), menyatakan bahwa XAMPP (Windows/Linux) Apache MySQL PHP dan Perl) merupakan paket server website PHP dan database MySQL yang paling populer dikalangan pengembang website dengan menggunakan PHP dan MySQL sebagai database.

Berdasarkan penjelasan di atas, dapat disimpulkan bahwa XAMPP merupakan sebuah paket kumpulan aplikasi server website PHP dan database MySQL.

2.17 HTML

Kustiyahningsih & Anamisa (2017), Menjelaskan bahwa HTML kependekan dari *Hyper Text Markup Language* yang file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *website page*. File-file HTML ini berisi instruksi-instruksi yang kemudian diterjemahkan oleh

browser yang ada dikomputer client (*user*) sehingga isi formasinya dapat ditampilkan secara visual dikomputer pengguna (*user*).

Hidayatulah dan Kawistara (2018), menjelaskan tentang HTML sebagai berikut :

Hypertext Markup Language (HTML) adalah Bahasa standar yang digunakan untuk menampilkan halaman website. Yang bisa dilakukan dengan HTML yaitu:

1. Mengatur tampilan dan halaman website dan isinya.
2. Membuat tabel dalam halaman website.
3. Mempublikasikan dalam halaman website secara online.
4. Membuat form yang bisa digunakan untuk menangani registrasi dan transaksi via website.

Contoh: Setiap dokumen HTML diawali dan diakhiri dengan tag HTML. Dari penjelasan di atas dapat disimpulkan bahwa *Hyper Text Markup Language* (HTML) adalah bahasa standar yang merupakan teks murni yang berisi instruksi yang diterjemahkan oleh *browser* dan ditampilkan secara visual di komputer pengguna.

2.18 Pengertian Pengujian Sistem

Al Fatta (2007:169), Pengujian sistem merupakan proses mengeksekusi sistem perangkat lunak untuk menentukan apakah sistem perangkat lunak tersebut cocok dengan spesifikasi sistem dan berjalan sesuai dengan lingkungan yang diinginkan. Pengujian sistem sering diasosiasikan dengan pencarian bug, ketidaksempurnaan program, kesalahan pada baris program yang menyebabkan kegagalan pada eksekusi sistem perangkat lunak.

2.19 *Black Box Testing*

Menurut Pressman (2010), ” *Blackbox* testing berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program”. Menurut Budiman (2012), “Pengujian *blackbox* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dibangkitkan, dieksekusi pada perangkat lunak dan kemudian keluaran dari perangkat lunak diuji apakah telah sesuai dengan yang diharapkan.”

Black-Box testing berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan perilaku (behavior) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

2.20 *Pengertian Hosting*

Menurut Arifin dan Yolanda Krisnadita (2017), “*Hosting* merupakan tempat penyimpanan data *website* dimana didalamnya meliputi kapasitas penyimpanan, *bandwidth* yang merupakan sebuah kapasitas yang digunakan untuk mengukur jumlah pengunjung *website* serta *database*”.

Menurut Aliyun (2014) dalam Samsul Arifin Dkk, “*Hosting* juga memiliki arti layanan berbasis internet sebagai tempat penyimpanan data atau tempat menjalankan aplikasi ditempat terpusat yang disebut dengan *server* dan dapat diakses melalui jaringan internet”.