

BAB II

TINJAUAN PUSTAKA DAN LANDASAR TEORI

2.1 Tinjauan Pustaka

Penelitian yang dilakukan oleh Maulana dkk bertujuan mempermudah perhitungan dan transaksi di Koperasi Cemara Lestari yang bergerak dibidang keuangan di daerah Depok Timur selain itu mempermudah pihak Koperasi untuk mengaksesnya karena sistem tersebut website dan bisa diakses melalui ponsel dan gadget lainnya. Tujuan dari penelitian ini adalah membangun sebuah Sistem Informasi Simpan Pinjam bisa dimanfaatkan oleh pihak Koperasi untuk membantu mengolah data keuangan di Koperasi tersebut dengan efektif. Untuk metodologi yang di gunakan yaitu : Analisa, Studi pustaka, Perancangan, Uji coba dan Pendukung support atau bisa disebut waterfall. Hasil dari penelitian ini bisa membantu Koperasi untuk menghitung hasil keuangan yang di inginkan dan pengolah data -data yang efektif maupun efisien dan akhirnya sistem tersebut benar benar bisa dimanfaatkan oleh Koperasi Cemara Lestari [4].

Syahrul Suci Romadhoni dkk juga melakukan penelitian menggunakan *Framework CI* Koperasi Simpan Pinjam Bumi Sejahtera perlu memiliki sebuah sistem informasi guna mempermudah proses pengolahan data, melaksanakan tugas secara tepat, dan meminimalkan kesalahan pencatatan transaksi maupun perhitungan keuangan, 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. Untuk memenuhi kebutuhan diatas maka perlu dirancang sebuah sistem informasi simpan pinjam. Tahap perancangan meliputi pembuatan deskripsi *system requirement specification, use case diagram, activity diagram, sequence diagram, class diagram*, serta perancangan database. Sistem informasi ini dibuat dengan menggunakan bahasa programam PHP melalui *framework CodeIgniter*, serta *MySQL* sebagai penyimpanan databasanya. Dengan menggunakan penyimpanan database,pada sistem informasi koperasi dapat memudahkan proses pengolahan data trasaksi simpan pinjam dikarenakan telah tersimpan pada 1 pusat basis data [5].

Mita Agnitia Lestari dkk juga melakukan penelitian membuat sistem informasi mengenai Pengolahan Data Administrasi Kependudukan pada Kantor Desa Pucung yang sangat membutuhkan adanya sistem informasi guna mempercepat proses pendataan, mencetak surat, mencari data dan membuat laporan sehingga kepuasan dalam pelayanan tidak hanya dapat dirasakan oleh masyarakat tetapi seluruh staff juga dapat merasakannya karena pekerjaan menjadi lebih efektif dan efisien. Sistem Pengolahan Data ini menggunakan metode *Waterfall* sebagai metode pengembangan perangkat lunak dengan tahap awal yaitu Analisa Kebutuhan Perangkat Lunak dan tahap akhir yaitu Desain, sistem ini telah penelitianalisa mengenai apa saja yang dibutuhkan oleh pengguna sehingga sistem ini dapat bermanfaat untuk pengguna. Dalam Sistem ini pengguna dapat dengan mudah mencari data, memasukkan dan menyimpan data dengan sistem keamanan yang menjamin, mencetak surat dengan cepat dan menghasilkan laporan dengan hasil yang akurat. Perancangan Sistem Pengolahan Data Administrasi Kependudukan ini merupakan solusi terbaik untuk memecahkan permasalahan-permasalahan yang ada di Kantor Desa Pucung Karawang. Dengan sistem ini, kinerja pelayanan akan meningkat dan menjadi lebih cepat, mudah dan rapih dibandingkan sistem yang terdahulu [6].

Penelitian lainnya dilakukan oleh Ibnu Rusdi Dkk dengan judul Rancang Bangun Sistem Informasi Simpan Pinjam Berbasis Web Pada Koperasi Unit Desa Soliamitra Depok (2020). Koperasi Unit Desa (KUD) Soliamitra membutuhkan adanya sistem pada proses bisnis berjalan dikarenakan pada sistem simpanannya sendiri sudah menggunakan sistem atau terkomputerisasi berbasis *desktop*, namun pada saat proses peminjaman masih dilakukan secara manual yaitu dengan menggunakan *Microsoft Excel* mulai dari pengajuan pinjaman, pembayaran angsuran sampai penyimpanan data-data yang berkaitan dengan peminjaman sampai dengan pembuatan laporan, sehingga pada saat merekap data pinjaman masih sering terjadi kesalahan pencatatan, dengan pemanfaatan teknologi sistem informasi simpan pinjam yang sudah berbasis web diharapkan dapat meningkatkan performa layanan kepada anggota Koperasi Unit Desa (KUD) Soliamitra. Dalam pembuatan sistem informasi ini penulis menggunakan metode SDLC dengan model *waterfall* (air terjun), karena tahapan demi tahapan dapat diselesaikan secara

terurut dan terstruktur, dan penggambaran rancang bangun menggunakan *Unified Modeling Language (UML)* sebagai analisa dan desain serta menggunakan *software NetBeans* untuk penerapan sistem berbasis objek *Java Servlet* untuk menunjang *Web Application* [7].

Penelitian lainnya juga dilakukan oleh Meli Dkk Penelitian ini bertujuan untuk membuat suatu aplikasi simpan pinjam koperasi karyawan, serta mendeskripsikan hasil uji coba sistem aplikasi. Metode penelitian yang digunakan adalah metode *Grounded Research*. Peneliti mengumpulkan data dan observasi di koperasi karyawan Indomobil dengan metode wawancara langsung kepada penanggung jawab dan petugas koperasi. Peneliti menemukan berbagai masalah yang dihadapi di koperasi tersebut yaitu kendala sistem yang masih sangat manual sehingga pekerjaan tidak efektif dan efisien. Hasil penelitian dapat disimpulkan dengan adanya aplikasi yang sudah terkomputerisasi, transaksi di koperasi semakin mudah dan cepat [8].

Pada penelitian ini penulis bermaksud Membangun sistem informasi Koperasi Simpan Pinjam yang dapat digunakan untuk membantu petugas dalam mengelola proses simpan pinjam sehingga mempercepat pelayanan kepada anggota koperasi yang mencakup semua kebutuhan anggota dalam proses simpan pinjam secara web. Dan mempermudah petugas dalam penarikan peminjaman sehingga tidak perlu mengkonfirmasi satu per satu karena tersedia *alert* jatuh tempo, dan anggota dapat membayar tepat pada waktunya.

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sistem informasi adalah suatu rangkaian sistem yang dikelompokkan dalam suatu organisasi yang terdiri dari sekumpulan komponen baik yang berbasis computer maupun manual yang dibuat untuk menghimpun dan menyiapkan data – data yang berisikan informasi keluaran untuk pemakai, atau sekumpulan perangkat keras dan perangkat lunak yang dihubungkan untuk menciptakan dan memproses data menjadi informasi yang berguna [10].

a. Akurat

Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksud dari informasi tersebut.

b. Tepat waktu

Tepat pada waktunya berarti sebuah informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi.

c. Relevan

Berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Menurut McFadden, dkk 1999 mendefinisikan informasi sebagai data yang telah diproses sehingga mempunyai arti dan dapat meningkatkan pengetahuan seseorang yang menggunakan data tersebut.

2.2.2 Website

Menurut purwanti (2008), website atau situs juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau bergerak, data animasi suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait dimana masing – masing dihubungkan dengan jaringan -jaringan halaman [11].

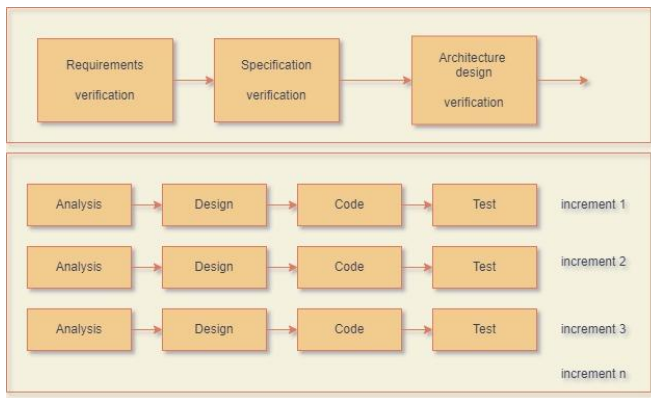
2.2.3 RPL (Rekayasa Perangkat Lunak)

Perangkat lunak adalah program komputer ditambah konfigurasi data dan file serta ditambahkan juga dokumentasi. Menurut *Ian Sommerville* : “*Computer programs and associated documentation. Software products may be developed for a particular customer or may be developed for a general market.*” Program komputer dan dokumentasi yang terkait. Produk *Software* dapat dikembangkan untuk pelanggan tertentu atau mungkin dikembangkan untuk umum. Menurut Pressman dalam bukunya *Software Engineering A Practioner’s Approach*, perangkat lunak didefinisikan lebih rinci lagi yaitu sebagai: instruksi-instruksi yang jika dieksekusi akan memberikan layanan-layanan atau fungsi seperti yang diinginkan struktur data yang memungkinkan program untuk memanipulasi informasi secara proporsional dokumen-dokumen yang menggambarkan operasi dan kegunaan program [17].

A. Metode Pengembangan Sistem

Tahap pengembangan sistem pada penelitian ini menggunakan model *incremental*. Model *incremental* adalah suatu solusi untuk mengantisipasi kondisi yang tidak diinginkan saat mengimplentasikan *incremental model*. More Risk Incremental Model menerapkan sistem kerja parallel dimana tahap awal yaitu membuat daftar kebutuhan (*requirement*) user kemudian dianalisa. Setelah spesifikasi pertama telah selesai, tim design akan membuat design untuk modul pertama. Pada saat itu juga, tim spesifikasi akan membuat spesifikasi untuk modul kedua dan jika sudah selesai, maka tim design akan membuat design untuk modul ke dua, dan seterusnya. Sehingga tidak harus menunggu modul pertama

selesai hingga dikirim ke user, baru lanjut ke modul kedua. Beberapa Kelebihan Dari Mode Incremental antara lain :



Gambar 2. 1 Desain Pemodelan Incremental

Model Incremental mempunyai tahapan – tahapan sebagai berikut :

1. *Requirement* adalah proses tahapan awal yang dilakukan pada *incremental* model untuk penentuan kebutuhan atau analisis kebutuhan.
2. *Specification*, adalah proses spesifikasi dimana menggunakan analisis

- kebutuhan sebagai acuannya.
3. *Architecture Design*, adalah tahap selanjutnya, perancangan *software* yang terbuka agar dapat diterapkan sistem pembangunan per-bagian pada tahapan selanjutnya.
 4. *Code*, setelah melakukan proses desain selanjutnya ada pengkodean/*coding*.
 5. *Test* merupakan tahap pengujian dalam model ini.

B. Metode Pengujian Sistem

Terkait dengan pengujian program, pengujian yang akan digunakan adalah pengujian *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan rekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *black-box* bukan merupakan alternatif dari teknik *white-box*, tetapi merupakan pendekatan komplementer yang memungkinkan besar mampu mengungkap kelas kesalahan daripada metode *white-box*. Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut [22] :

1. Fungsi-fungsi yang tidak benar atau hilang,
2. Kesalahan *interface*,
3. Kesalahan dalam struktur data atau akses *database* eksternal,
4. Kesalahan kinerja,
5. Inisialisasi dan kesalahan terminal.

Pengujian *black-box* ini terdapat beberapa proses. Proses-proses yang ada dalam pengujian ini diantaranya :

- a. Menganalisa kebutuhan dan spesifikasi dari perangkat lunak.
- b. Pemilihan jenis *input* yang memungkinkan menghasilkan *output* dengan benar serta jenis *input* yang memungkinkan *output* salah pada perangkat lunak yang sedang diuji.
- c. Menentukan *output* untuk satu jenis *input*.
- d. Pengujian dilakukan dengan *input-input* yang telah benar-benar diseleksi.
- e. Melakukan pengujian
- f. Perbandingan *output* yang dihasilkan dengan *output* yang diharapkan

2.2.4 PBO (Pemrograman Berorientasi Object)

Pemrograman berorientasi objek merupakan suatu metode pemrograman yang berbasiskan pada objek . Pada pemrograman berorientasi objek, antara objek yang satu berinteraksi dengan objek lainnya dengan tujuan mendapatkan informasi antar objek. Pemrograman berorientasi objek merupakan bagian dari pengembangan sistem berorientasi objek [18]

a. Pewarisan

Inheritance adalah dimana suatu entitas/obyek dapat mempunyai entitas/obyek turunan. Dengan konsep inheritance, sebuah *class* dapat mempunyai *class* turunan. Seperti halnya manusia pada umumnya terdapat orang tua dan anak pada PBO juga dikenal *parent class* atau *base class* dan *subclass* atau *child class*. Dimana *subclass* atau *childclass* mewarisi semua data yang ada di *parent class* atau *base class* atau dapat disimpulkan jika *subclass* atau *childclass* adalah perluasan dari *parent class* atau *base class*.

b. Objek

Objek merupakan teknik dalam menyelesaikan masalah yang kerap muncul dalam pengembangan perangkat lunak. Teknik ini merupakan teknik yang efektif dalam menemukan cara yang tepat dalam membangun sistem dan menjadi metode yang paling banyak dipakai oleh para pengembang perangkat lunak. Orientasi objek merupakan teknik pemodelan sistem riil yang berbasis objek. Objek adalah entitas yang memiliki atribut, karakter dan kadang kala disertai kondisi. Objek mempresentasikanseuai kenyataan seperti siswa, mempresentasikan dalam bentuk konsep seperti merek dagang, juga bisa menyatakan visualilasi seperti bentuk huruf (font).

c. Class

Kelas (class) merupakan penggambaran satu set objek yang memiliki atribut yang sama. Kelas mirip dengan tipe data ada pemrograman non objek, akan tetapi lebih komprehensif karena terdapat struktur sekaligus karakteristiknya. Kelas baru dapat dibentuk lebih spesifik dari kelas ada umumnya.kelas merupakan jantung dalam pemrograman berorientasi objek.

d. Method

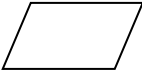

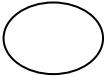

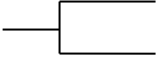
Method adalah kumpulan program yang mempunyai nama. Method merupakan sarana bagi programmer untuk memecah

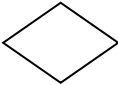





program menjadi bagian-bagian yang kecil agar jadi lebih kompleks sehingga dapat di gunakan berulang-ulang.

2.2.5 Flowchart

Flowchart adalah bagan (chart) yang menunjukkan alir (flow) di dalam program atau prosedur sistem secara logika. Bagan alir (flowchart) digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi [19].

Tabel 2. 1 Simbol-simbol flowchart

No	Simbol	Nama	Keterangan
1.		<i>Input/Output</i>	Mempresentasikan <i>input</i> data atau <i>output</i> data yang diproses atau informasi.
2.		Proses	Mempresentasikan operasi.
3.		Penghubung	Keluar ke atau masuk dari bagian lain, khususnya halaman yang sama.
4.		Anak Panah	Mempresentasikan alur kerja.
5.		Penjelasan	Digunakan untuk komentar tambahan.

6.		Keputusan	Keputusan dalam program.
7.		<i>Predefined Process</i>	Rincian operasi berada di tempat lain.
8.		<i>Preparation</i>	Pemberian harga awal.
9.		<i>Terminal Port</i>	Awal/akhir <i>Flowchart</i> .
11.		Dokumen	<i>Input/Output</i> dalam format yang dicetak.
19.		<i>Manual Operation</i>	Operasi manual.




2.2.6 UML


UML (Unified Modeling Language) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasi dari sebuah sistem pengembangan software berbasis OO (Object-Oriented). UML tidak

hanya merupakan sebuah bahasa pemrograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemrograman, seperti JAVA, C++, Visual Basic, atau bahkan dihubungkan secara langsung ke dalam sebuah object-oriented database. Berikut diagram – diagram yang ada pada UML diantaranya [14]:

A. State Machine

State machine diagram adalah diagram yang menggambarkan perubahan dan transisi keadaan (dari satu status ke status lainnya) sebuah objek pada program sebagai akibat dari stimulan yang diterima. State machine diagram berfungsi untuk memodelkan metode/*behavior (lifecycle)* sebuah kelas atau objek menunjukkan urutan kejadian sesaat (status) yang dilewati sebuah objek, transisi dari sebuah status ke status lainnya [14]. Berikut ini simbol dari *State machine* diagram adalah :

No	Simbol	Nama	Keterangan
1.		<i>State</i>	digambarkan berbentuk segi empat dengan sudut membulat dan memiliki nama sesuai kondisi saat itu.
2.		Titik awal (<i>start</i>)	Digunakan untuk menggambarkan awal dari kejadian dalam suatu diagram <i>flowchart</i> .
3.		Titik Akhir(<i>end</i>)	Digunakan untuk menjelaskan/ menggambarkan akhir (<i>end</i>) dari kejadian dalam suatu diagram


			state chart.
4.		<i>Point</i>	Digunakan untuk menggambarkan atau menjelaskan apakah akan masuk (<i>entety point</i>) ke dalam status atau keluar (<i>exit point</i>).

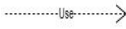
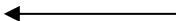
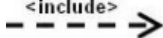
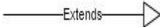


Tabel 2.2 Simbol - simbol *State Machine*

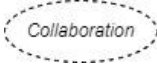

B. Use Case

Use Case diagram adalah suatu urutan interaksi yang saling berkaitan antara sistem dan aktor. *Use case* dijalankan melalui cara menggambarkan tipe interaksi antara user suatu program (sistem) dengan sistemnya sendiri. *Use case* melalui sebuah cerita yang mana sebuah sistem itu dipakai. *Use case* juga dipakai untuk membentuk perilaku (*behaviour*) sistem yang akan dibuat. Sebuah use case menggambarkan sebuah interaksi antara pengguna (aktor) dengan sistem yang sudah ada [14].

Tabel 2.3 Simbol - Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.

2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
3		<i>Generalization</i>	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
4		<i>Include</i>	Menspesifikasikan bahwa use case sumber secara eksplisit.
5		<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket menampilkan yang secara terbatas.

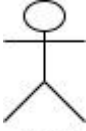

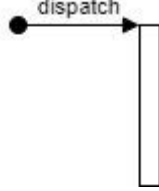
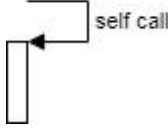
8		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemennya (sinergi).
9		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

C. *Sequence Diagram*

Sequence Diagram adalah salah satu dari diagram - diagram yang ada pada UML, sequence diagram ini adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah *object*.

Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Dalam UML, *object* pada sequence diagram digambarkan dengan segi empat yang berisi nama dari *object* yang digarisbawahi [14]. Adapun simbol atau komponen sequence digram sebagai berikut.

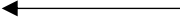

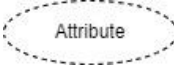

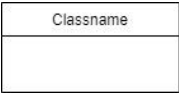

Tabel 2.4 Simbol – Simbol *Sequence Diagram*


No	Gambar	Nama	Keterangan
1	 Actor	<i>Actor</i>	Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.
2		<i>Life Line</i>	Objek <i>entity</i> , antar muka yang saling berinteraksi.
3		<i>Found Message</i>	Menggambarkan pesan hubungan antar objek yang menunjukkan urutan kejadian.
4		<i>Self Call</i>	Menggambarkan pesan atau hubungan objek yang menunjukkan urutan kejadian yang terjadi.

D. Class Diagram

Class diagram adalah gambaran alur database dalam program. Dalam sebuah laporan sistem maka class diagram wajib ada. Sequence diagram adalah bagian dari UML (Unified Modeling Language). Dalam sebuah laporan sequence diagram wajib ada karena sangat penting [14].

Tabel 2.5 Simbol - simbol *Class Diagram*

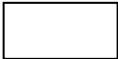
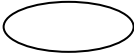
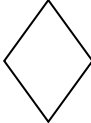
No.	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Collaboration</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
4.		<i>Relazation</i>	Operasi yang benar – benar dilakukan oleh suatu objek.
5.		<i>Class</i>	Himpunan dari objek – objek yang berbagi atribut serta operasi yang sama.
6.		<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen yang mandiri yang akan mempengaruhi elemen yang bergantung pada elemn yang tidak

			mandiri.
		<i>Association</i>	Yang menghubungkan antara objek satu dengan lainnya.

E. Erd

ERD adalah pemodelan data atau 22system dalam database, Fungsi ERD adalah untuk memodelkan struktur dan hubungan antar data yang 22 sistem 22 kompleks. Keberadaan 22system Entity Relationship Diagram sangat penting untuk perusahaan dalam mengelola data yang dimilikinya [14].

Tabel 2.6 Simbol-simbol ERD

No	Gambar	Nama	Keterangan
1		Entity	Suatu objek yang dapat diidentifikasi dalam lingkungan pemakai
2		Attribute	Mendeskripsikan karakter entitas
3		Decision	Menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda
4		Asociations	Penghubung antar relasi dan entitas, relasi dan entitas dengan atribut

F. Mockup

Mockup ini merupakan gambaran yang paling realistis & menyerupai hasil final yang ingin ditunjukkan. Mockup biasanya akan diberikan atau ditunjukkan setelah tahap perancangan konsep selesai, sebelum memasuki eksekusi final. Contohnya ketika kita mendesain packaging, kita bisa merealisasikan desain packaging tersebut dengan mockup, sehingga kita bisa menunjukkan ke klien seperti apa bentuk desain tersebut ketika sudah diaplikasikan dalam bentuk packaging tanpa harus kita cetak terlebih dahulu [15].

2.2.7 Basis Data

Basis data adalah kumpulan file-file yang mempunyai kaitan antara satu file dengan file lain sehingga membentuk satu bangun data untuk menginformasikan suatu perusahaan instansi, dalam bahasan tertentu. MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi *user* serta menggunakan perintah standar *Structured Query Language* (SQL). MySQL memiliki dua bentuk lisensi, yaitu *Free Software* dan *Shareware*. MySQL yang biasa kita gunakan adalah *MySQL Free Software* yang berada dibawah Lisensi *General Public License* (GPL).

MySQL juga dapat didefinisikan sebagai sebuah database server, dapat juga berpersion sebagai *client* sehingga sering disebut *database client / server* yang *open source* dengan kemampuan dapat berjalan baik di Operasi Sistem maupun dengan *Platform Windows* maupun *Linux*.

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah suatu bahasa (*language*) yang digunakan untuk mengakses data di dalam sebuah database relasional. SQL sering juga disebut dengan istilah query, dan bahasa SQL secara praktiknya digunakan sebagai bahasa standar untuk manajemen database relasional. Hingga saat ini hampir seluruh server database atau software database mengenal dan mengerti bahasa SQL. Dalam penggunaan SQL terdapat beberapa perintah yang berguna untuk mengakses dan memanajemen data yang terdapat dalam database. Secara garis besar, SQL Server mempunyai 2 (Dua) jenis perintah SQL yaitu :

a. *Data Definition Language (DDL)*

DDL adalah sub perintah dari bahasa SQL yang digunakan untuk membangun kerangka sebuah database, dalam hal ini database dan table. Terdapat tiga perintah penting dalam DDL, yaitu :

1. *CREATE* : Perintah ini digunakan untuk membuat, termasuk di dalamnya membuat database baru, tabel baru view baru, dan kolom baru. Contoh: `CREATE DATABASE nama_database.`
2. *ALTER* : Perintah *ALTER* berfungsi untuk mengubah struktur tabel yang telah dibuat. Mencakup di dalamnya mengubah nama tabel, menambah kolom, mengubah kolom, menghapus kolom, dan memberikan atribut pada kolom. Contoh: `ALTER TABLE nama_tabel ADD nama_kolom datatype.`
3. *DROP* : Perintah *DROP* berfungsi untuk menghapus database atau tabel. Contoh: `DROP DATABASE nama_database.`

b. *Data Manipulation Language (DML)*

DML adalah sub perintah dari bahasa SQL yang digunakan untuk memanipulasi data dalam database yang telah dibuat. Terdapat 4 (Empat) perintah penting dalam DML, yaitu :

1. *INSERT* : Perintah ini digunakan untuk memasukkan data baru ke dalam sebuah tabel. Perintah ini tentu saja bisa dijalankan ketika database dan tabel sudah dibuat. Contoh: `INSERT INTO nama_tabel VALUES (data1, data2, dst...);`
2. *UPDATE* : Perintah *update* digunakan untuk memperbaharui data pada sebuah tabel. Contoh: `UPDATE nama_tabel SET kolom1=data1, kolom2=data2,... WHERE kolom=data;`
3. *DELETE* : Perintah *delete* digunakan untuk menghapus data dari sebuah tabel. Contoh: `DELETE FROM nama_tabel WHERE kolom=data;`

4. *SELECT* : Perintah ini digunakan untuk mengambil dan menampilkan data dari tabel atau bahkan dari beberapa tabel dengan penggunaan relasi. Contoh: `SELECT nama_kolom1, nama_kolom2 FROM nama_tabel;`

2.2.8 Koperasi

Koperasi merupakan salah satu badan usaha yang berbadan hukum dengan usaha yang beranggotakan orang seorang yang berorientasi menghasilkan nilai tambah yang dapat dimanfaatkan bagi peningkatan kesejahteraan anggotanya. Selain itu, koperasi juga sebagai gerakan ekonomi rakyat yang berorientasi untuk menumbuhkan partisipasi masyarakat dalam upaya memperkokoh struktur perekonomian nasional dengan demokrasi ekonomi yang berdasarkan atas asas kekeluargaan. Oleh karena itu, untuk menyelaraskan dengan perkembangan lingkungan yang dinamis seperti era globalisasi sekarang ini maka, perlu adanya peningkatan usaha yang mampu mendorong koperasi agar dapat tumbuh dan berkembang menjadi lebih kuat dan mandiri [9]

2.2.9 Koperasi Simpan Pinjam

Koperasi Simpan Pinjam adalah koperasi yang melaksanakan kegiatan usahanya hanya menghimpun dana dan menyalurkannya melalui kegiatan usaha simpan pinjam. Kegiatan usaha simpan pinjam tersebut dilaksanakan dari dan untuk anggota koperasi yang bersangkutan, calon anggota yang memenuhi syarat dan koperasi lain/anggotanya [12].

2.2.10 Koperasi Wanita Patra

Koperasi Wanita Patra Cilacap merupakan koperasi dengan jenis simpan pinjam dan serba usaha yang berkedudukan di Jln Ir. H.Juanda Komplek Perumahan Pertamina Gunung Sumping, Kabupaten Cilacap Provinsi Jawa Tengah. Pada pertengahan tahun 1983 Ny. M.J. Pitono ketua Dharma Wanita Pertamina Cilacap mendirikan Koperasi Wanita Patra Cilacap seagai perkumpulan keluarga, karyawan, pensiunan Pertamina RU IV Cilacap dalam rangka mendukung kesejahteraan keluarga [13].