PERANCANGAN PERANGKAT LUNAK SISTEM *E-COMMERCE* (STUDI KASUS TOKO A4 SPORT PADANG)

Tugas Akhir

Diajukan kepada Tim Penguji Tugas Akhir Jurusan Teknik Elektronika
Sebagai Salah Satu Persyaratan Guna Memperoleh Gelar Sarjana Pendidikan

Oleh :

WIDIA FEBRYANTI

NIM : 53076 / 2010

PROGRAM STUDI PENDIDIKAN TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI PADANG
2015

PERSETUJUAN TUGAS AKHIR

PERANCANGAN PERANGKAT LUNAK SISTEM *E-COMMERCE*
(STUDI KASUS TOKO A4 SPORT PADANG)

Nama	: Widia Febryanti
TM/NIM	: 2010/53076
Program Studi	: Pendidikan Teknik Informatika
Jurusan	: Teknik Elektronika
Fakultas	: Teknik

Padang, Mei 2015

Disetujui Oleh

Pembimbing I,

Drs. Denny Kurniadi, M.Kom
NIP.19630606 198903 1 001

Pembimbing II,

Yeka Hendriyani, S.Kom, M.Kom.
NIP. 19840520 201012 2 003

Mengetahui,
Ketua Jurusan Teknik Elektronika
FT-UNP

Drs. Putra Jaya, M.T.
NIP. 19621020 198602 1 001

PENGESAHAN

Dinyatakan lulus setelah dipertahankan di depan Tim Penguji Tugas Akhir
Bukan Skripsi Program Studi Pendidikan Teknik Informatika
Jurusan Teknik Elektronika FakultasTeknik
Universitas Negeri Padang

Judul	**: Perancangan Perangkat Lunak Sistem *E-Commerce* (Studi Kasus Toko A4 Sport Padang)**
Nama	: Widia Febryanti
TM/NIM	: 2010/53076
Program Studi	: Pendidikan Teknik Informatika
Jurusan	: Teknik Elektronika
Fakultas	: Teknik

Padang, Mei 2015

Tim Penguji

	Nama	Tanda Tangan
1. Ketua	: Oktoria, S.Pd, MT	1.
2. Sekretaris	: Drs. Denny Kurniadi, M.Kom	2.
3. Anggota	: Yeka Hendriyani, S.Kom, M.Kom	3.
4. Anggota	: Asrul Huda, S.Kom, M.Kom	4.

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa tugas akhir ini benar-benar karya saya sendiri. Sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang ditulis atau diterbitkan orang lain kecuali sebagai acuan atau kutipan dengan mengikuti tata penulisan karya ilmiah yang lazim.

Padang, 10 Mei 2015

Yang menyatakan,

Widia Febryanti

ABSTRAK

Widia Febryanti 2010**. Perancangan Perangkat Lunak Sistem *E-Commerce* (Studi Kasus Toko A4 Sport Padang).** Prodi Pendidikan Teknik Informatika, Jurusan Elektronika, Fakultas Teknik, Universitas Negeri Padang.

Keterbatasan pegawai yang menangani penjualan secara manual, mengurangi produktivitas penjualan. Selain itu, Keterbatasan waktu menjadi kendala untuk pembuatan laporan stok produk dan laporan penjualan oleh kasir. Dimana beragamnya format dokumentasi, juga menambah waktu dalam penyusunan laporan. Perkembangan teknologi internet yang sangat pesat dapat mendukung penerapan sistem *e-commerce* dalam penjualan produk. Tujuan Tugas Akhir ini adalah untuk membuat *website e-commerce* Toko A4 Sport Padang, yang merupakan sarana pemasaran dan penjualan produk secara *online*. Dimana pada perancangan sistem ini, penulis menggunakan konsep *Object Oriented Programming* (OOP) dan menggunakan pemodelan *Unified Modelling Language* (UML). Setelah diuji coba, perancangan yang menggunakan PHP *Framework Codeigniter* dan *MySQL* sebagai DBMS untuk *database*, berhasil dengan baik. Sehingga dapat diimplementasikan oleh Toko A4 Sport Padang, dalam hal promosi dan penjualan *online,* serta dapat melakukan manajemen seluruh data produk, data laporan dan data *user.*

Kata Kunci : *E-Commerce, Database, Object Oriented Programming, Unified Modelling Language*

KATA PENGANTAR

Syukur Allhamdulillah atas kehadirat Allah SWT yang telah memberikan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul ” Perancangan Perangkat Lunak Sistem *E-Commerce (*Studi Kasus Toko A4 Sport Padang)”. Tugas Akhir ini ditulis sebagai salah satu syarat memperoleh gelar Sarjana Pendidikan Program Studi Pendidikan Teknik Informatika, Jurusan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Padang. Tugas Akhir ini dapat diselesaikan berkat bantuan dan bimbingan dari berbagai pihak. Untuk itu dalam kesempatan ini disampaikan penghargaan dan rasa terima kasih yang setulus-tulusnya kepada :

1. Bapak Drs. Denny Kurniadi, M.Kom selaku pembimbing I yang telah memberikan saran dan banyakbimbingan dalam penulisan Tugas Akhir ini.
2. Ibu Yeka Hendriyani, S.Kom, M.Kom selaku pembimbing II yang telah memberikan saran dan bimbingan dalam penulisan Tugas Akhir ini.
3. Bapak Oktoria, S.Pd, MT selaku dosen ketua penguji Tugas Akhir.
4. Bapak Asrul Huda, S.Kom, M.Kom selaku dosen penguji Tugas Akhir.
5. Bapak dan Ibu dewan dosen Program Studi Pendidikan Teknik Informatika Jurusan Teknik Elektronika, Fakultas Teknik Universitas Negeri Padang yang telah membimbing penulis selama menuntut ilmu.
6. Pihak Toko A4 Sport Padang, yang telah bersedia bekerja sama dan memberikan izin pengambilan data.

7. Rekan-rekan mahasiswa jurusan teknik elektronika Universitas Negeri Padang, khususnya program studi pendidikan teknik informatika angkatan 2010.

Semua pihak yang telah membantu penulis dalam menuntaskan Tugas Akhir ini dan menyelesaikan studi, yang tidak bisa penulis sebutkan satu per satu. Semoga bantuan dan bimbingan serta arahan yang diberikan menjadi amal saleh dan mendapat pahala dari Allah SWT, amin.

Semoga petunjuk, saran, bimbingan dan motivasi yang bapak, ibu, keluarga, dan teman-teman berikan menjadi amal kebaikan dan mendapatkan balasan yang lebih baik dari ALLAH SWT. Tak ada gading yang tak retak tidak ada manusia yang sempurna, untuk itu dengan segala kerendahan hati diharapkan saran dan kritik yang membangun dari semua pihak demi sempurnanya Tugas Akhir ini. Semoga Tugas Akhir ini bisa bermanfaat bagi penulis khususnya, dan semua pihak pada umumnya.

Padang, 10 Mei 2015

Penulis

DAFTAR ISI

Halaman

HALAMAN JUDUL i

ABSTRAK ii

KATA PENGANTAR iii

DAFTAR ISI v

DAFTAR GAMBAR viii

DAFTAR TABEL x

DAFTAR LAMPIRAN xii

BAB I PENDAHULUAN 1

A. Latar Belakang 1

B. Identifikasi Masalah 3

C. Batasan Masalah 4

D. Rumusan Masalah 4

E. Tujuan Tugas Akhir 5

F. Manfaat Tugas Akhir 6

BAB II LANDASAN TEORI 8

A. Ruang Lingkup *E-Commerce* 8

1. Pengertian *E-Commerce* 8

2. Keuntungan *E-commerce* 9

3. Kerugian Perdagangan Elektronik 12

4. Pemesanan 13

5. *Promotion* (Promosi) *E-Commerce* 13

B. Rekayasa Perangkat Lunak (RPL) 14

1. Pengertian RPL 14

2. Kriteria Rekayasa Perangkat Lunak 15

3. Tantangan Rekayasa Perangkat Lunak 16

5. Pengembangan RPL Model *Waterfall* 17

C. Perancangan *Unified Modelling Language (UML)* 18
1. Pengertian UML 18
2. Diagram-diagram Dalam UML 19
D. Konsep Dasar Pemrograman Berorientasi Objek 21
1. Pengertian Pemrograman Berorientasi Objek 21
2. Keuntungan Menggunakan Metodologi Berorientasi Objek ... 22
3. Konsep Dasar Pemrograman Berorientasi Objek 22
E. *Database (*Basis Data) 23
1. *Database* MySQL 23
2. DBMS (*Database Management System)* 24
3. Alur Hidup *Database* 25
4. Normalisasi 26
5. *Entity Relationship Diagram (ERD)* 27
F. *Tools* Pendukung 28
1. *Xammp* 28
2. *Framework* 28
3. *Codeigniter* 30
4. *Model, View, Controller* 33

BAB III ANALISIS DAN PERANCANGAN SISTEM 35

A. Analisis Sistem 35
1. Analisis Sistem Yang Berjalan 35
2. Analisis Fungsional 40
3. Analisis Non Fungsional 40
4. Analisis Kebutuhan Sistem 41
5. Analisis Masalah dan Solusi 41
6. Analisis Antar Muka 42
7. *Flowmap* yang Sedang Berjalan 43
B. Perancangan Sistem 44
1. Flowmap Sistem Yang Akan Dirancang 44

2. Perancangan *Context Diagram* 46
3. Use Case Diagram 47
4. Activity Diagram 48
C. Perancangan *Database* 53
1. Normalisasi 54
2. Struktur Tabel 57
3. *Entity Relationship Diagram* (ERD) 63
D. Perancangan Antar Muka (*Interface*) 64
1. Perancangan Antar Muka Halaman Utama 64
2. Perancangan *Input* 64
3. Perancangan *Output* 68

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM 72

A. Implementasi Sistem 72
1. Halaman *Home Costumer* 72
2. Halaman *Login Costumer* 74
3. Koneksi *Database* 75
4. Halaman *User* Admin 76
5. Halaman *User Owner* 90
6. Halaman *User Supplier* 95
B. Pengujian Sistem 99
1. Pengujian Halaman *Akses Costumer* 99
2. Pengujian Halaman Akses Admin 100
3. Pengujian Halaman Akses *Owner* 102
4. Pengujian Halaman Akses *Supplier* 103
5. Kelemahan dan Solusi Sistem 104
C. Pembahasan 104

BAB V PENUTUP 106

A. Kesimpulan 106

B. Saran 107

DAFTAR PUSTAKA 108

DAFTAR GAMBAR

Halaman

Gambar 1 : Ilustrasi Model *Waterfall* 17

Gambar 2 : Alur Hidup Basis Data 26

Gambar 3 : *Flowmap* yang Sedang Berjalan 43

Gambar 4 : *Flowmap* yang Akan Dibangun 45

Gambar 5 : *Context Diagram* 46

Gambar 6 : *Use Case* Diagram 48

Gambar 7 : *Activity Diagram* Menampilkan Menu *Home* 49

Gambar 8 : *Activity Diagram* Registrasi *User* 50

Gambar 9 : *Activity Diagram Login* 50

Gambar 10 : *Activity Diagram Order* 51

Gambar 11 : *Activity Diagram* Konfirmasi Pembayaran 52

Gambar 12 : *Activity Diagram* Cek Status *Order* 53

Gambar 13 : *Entity Relationship Diagram* 63

Gambar 14 : Rancangan Halaman Utama 64

Gambar 15 : *Form Input Admin* 65

Gambar 16 : Rancangan *Input* Halaman *Login Supplier* 66

Gambar 17 : Rancangan *Input* Halaman *Login Customer* 66

Gambar 18 : Rancangan *Input* Halaman Biodata *Costumer* 67

Gambar 19 : *Form Input* Data *Owner* 68

Gambar 21 : Halaman *Login Costumer* 74

Gambar 22 : Pesan *Error Login* 75

Gambar 23 : Halaman *Home* Admin 76

Gambar 24 : Halaman Manajemen Kategori Admin 77

Gambar 25 : Halaman Manajemen Produk Admin 78

Gambar 26 : Halaman Manajemen Merk Produk 79

Gambar 27 : Halaman Manajemen Ulasan Admin 80
Gambar 28 : Halaman Program Diskon Admin 81
Gambar 29 : Halaman Stok Produk Sedikit 82
Gambar 30 : Halaman Permintaan Produk Admin 82
Gambar 31 : Halaman Pesan Admin 84
Gambar 32 : Halaman Manajemen *Order* Admin 84
Gambar 33 : Halaman *Edit* Status *Order* 85
Gambar 34 : Halaman Pengaturan Admin 86
Gambar 35 : Halaman Menu *Supplier User* Admin 87
Gambar 36 : Halaman Menu Kurir *User* Admin 88
Gambar 37 : Halaman Menu Laporan Admin 89
Gambar 38 : Halaman Katalog Kategori *Owner* 90
Gambar 39 : Halaman Katalog Produk *Owner* 91
Gambar 40 : Halaman Katalog Merk Produk *Owner* 92
Gambar 41 : Halaman Lapora *Order Owner* 93
Gambar 42 : Halaman Laporan Penjualan *Owner* 94
Gambar 43 : Halaman *Login Supplier* 96
Gambar 44 : Halaman Permintaan Produk *User Supplier* 97
Gambar 45 : Halaman Pesan *User Supplier* 99

DAFTAR TABEL

Halaman

Tabel 1. Simbol-simbol Pada *Use Case Diagram* 20

Tabel 2. Simbol-simbol Pada *Activity Diagram* 22

Tabel 3. Perkembangan SQL 23

Tabel 4. Simbol-simbol ERD 28

Tabel 5. Analisis *User* 35

Tabel 6. Analisis Dokumen 36

Tabel 7. Analisis Masalah dan Solusi 42

Tabel 8. Bentuk *Unnormal* 54

Tabel 9. *First Normal Form (1NF)* 55

Tabel 10. *Second Normal Form (2NF1)* 56

Tabel 11. *Second Normal Form (2NF2)* 56

Tabel 12. *Second Normal Form (2NF3)* 56

Tabel 13. Normalisasi Admin 56

Tabel 14. Normalisasi *Supplier* 57

Tabel 15. Normalisasi *Costumer* 57

Tabel 16. Normalisasi *Order* Produk 57

Tabel 17. Tabel Admin 58

Tabel 18. Tabel *Costumer* 58

Tabel 19. Tabel *Supplier* 59

Tabel 20. Tabel Kategori 60

Tabel 21. Tabel Produk 60

Tabel 22. Tabel *Order* 61

Tabel 23. Tabel *Order* Produk 62

Tabel 24. Tabel *Confirm* 62

Tabel 25. Rancangan *Output Order Costumer* 69

Tabel 26. Laporan *Order* 70

Tabel 27. Laporan Penjualan 70

Tabel 28. Pengujian Halaman Akses *Costumer* 99

Tabel 29. Pengujian Halaman Akses Admin 100

Tabel 30. Pengujian Halaman Akses *Owner* 102

Tabel 31. Pengujian Halaman Akses *Supplier* 103

Tabel 32. Kelemahan Sistem..... 104

DAFTAR LAMPIRAN

Lampiran 1 : Lampiran *Script*

Lampiran 2 : Lampiran Surat Izin Pengambilan Data

Lampiran 3 : Lampiran Surat Balasan Izin pengampilan Data

Lampiran 4 : Lampiran Surat Keterangan pengambilan Data

BAB I

PENDAHULUAN

A. Latar Belakang

Perkembangan teknologi informasi dan komunikasi di seluruh dunia, membuka peluang bagi para pelaku usaha untuk mengembangkan kegiatan bisnis melalui media elektronik (*electronic business/e-business*), yang diprediksi akan menjadi model bisnis besar di masa depan. Salah satu bentuk *e-business* yang paling dikenal publik adalah kegiatan "perdagangan secara elektronik", atau yang lazim dikenal dengan istilah *electronic commerce* atau *e-commerce*.

Perkembangan yang begitu pesat, sekarang sangat menarik perhatian masyarakat dalam mengembangkan bisnis *online* melalui *website* maupun jejaring sosial. *Website e-commerce* sebagai salah satu tempat perdagangan *online* yang semakin banyak bermunculan di internet, mulai dari bidang usaha *fashion*, kosmetik, elektronik, buku, aksesoris dan properti olahraga.

Website termasuk salah satu tempat berbelanja *online* yang sangat mudah untuk melakukan transaksi pembelian. Dengan sistem belanja yang praktis membuat masyarakat lebih tertarik berbelanja langsung melalui *website* itu sendiri. *Website* menyediakan keterangan lengkap tentang produk dan dapat melakukan transaksi di dalam *website* itu sendiri, sehingga

kapan saja dan dimana saja kita dapat melakukan pembelian, dengan catatan kita terkoneksi dengan internet.

Demikian pula studi kasus yang dilakukan pada Toko A4 Sport Padang, yang beralamat di Jl. Sutomo, Simpang Aru, Padang. Toko ini menjual beberapa perlengkapan olah raga dengan *multibrand* atau banyak merk. Tapi dengan menggunakan *supplier* resmi dengan merk Adidas. Seperti sepatu olahraga, bola, seragam olahraga, dan sebagainya.

Pada toko A4 Sport jumlah karyawan hanya 2 orang. Keterbatasan jumlah karyawan dan beragamnya format dokumentasi, menambah waktu dalam penyusunan laporan manual stok produk maupun penjualan, serta tidak adanya keamanan data. Karyawan tersebut juga mengemban tugas untuk kebersihan dan kerapian toko. Tidak jarang ketika banyaknya pelanggan datang, karyawan tersebut harus meninggalkan pekerjaannya untuk melayani konsumen. Sehingga pelanggan tidak terlayani dengan maksimal. Banyaknya pelanggan membuat permintaan dan pesanan akan suatu produk menjadi meningkat, keadaan tersebut menjadi kendala dalam persediaan stok.

Berdasarkan hasil survei penulis, pengunjung rata-rata per hari adalah 30-35 orang, dengan jumlah pembeli rata-rata 25-30 orang, dan pengunjung yang berasal dari luar kota Padang rata-rata 20-25 orang per bulan. Di samping itu, tingkat rata-rata penjualan toko ini per hari adalah Rp. 2.500.000,00 sampai dengan Rp. 8.500.000,00.

Dengan kondisi seperti ini, perkembangan teknologi *e-commerce* dapat dimanfaatkan demi kelancaran sebuah penjualan. Selain dapat di

akses *online,* manajemen sistem *report* dan *order* produk dapat dilakukan dengan maksimal antar *user* terkait. Sehingga dalam penyusunan laporan stok produk dan penjualan menjadi lebih efektif dan efisien.

Untuk perancangan sebuah *website e-commerce* yang mempunyai sistem seperti dijelaskan di atas, penulis membutuhkan bahasa pemograman PHP *Framework Codeigniter*. Karena *codeigniter* dieksekusi dengan cepat dan memiliki *performance* yang handal. Selain itu, *framework codeigniter* juga menerapkan konsep *Model-View-Controller* (MVC). Pemodelan yang digunakan adalah *Unified Modelling Language* (UML), karena bahasa desainnya lebih mudah untuk diimplementasikan. Sedangkan konsep perancangan yang digunakan adalah *Object Oriented Programming* (OOP), karena konsep ini lebih cocok digunakan untuk pemograman PHP.

Berawal dari latar belakang masalah tersebut, penulis tertarik untuk merancang perangkat lunak sistem *e-commerce* untuk Toko A4 Sport sebagai Tugas Akhir (TA) dengan judul penulisan "**Perancangan Perangkat Lunak Sistem *E-Commerce* (Studi Toko A4 Sport Padang).**

B. Identifikasi Masalah

Berdasarkan latar belakang di atas, dapat diidentifikasi masalah sebagai berikut :

1. Keterbatasan karyawan yang menangani penjualan secara manual, mengurangi produktivitas penjualan.
2. Keterbatasan waktu menjadi kendala untuk pembuatan laporan stok produk dan laporan penjualan.

3. Beragamnya format dokumentasi menambah waktu dalam penyusunan laporan.
4. Banyaknya permintaan suatu produk, namun tidak bisa terlayani dengan baik.
5. Banyaknya pelanggan yang melakukan pesanan, sehingga mengalami kendala dalam persediaan stok.

C. Batasan Masalah

Agar tugas akhir ini lebih terarah dan sesuai dengan tujuan yang diharapkan, maka ditetapkan batasan masalah yaitu sebagai berikut :

1. Perangkat lunak yang akan dibangun adalah sistem *e-commerce* yang dapat diakses *user* secara *online.*
2. Perangkat ini dapat melayani sistem penjualan *online* dan sistem manajemen data produk dan laporan.
3. Perangkat yang diimplementasikan pada perancangan sistem *e-commerce* menggunakan konsep *Object Oriented Programming* (OOP).
4. Model perancangan sistem *e-commerce* yang digunakan adalah *Unified Modelling Language* (UML).
5. Bahasa pemrograman yang digunakan pada perancangan perangkat lunak sistem *e-commerce* adalah PHP *Framework Codeigniter, software development code generator atau* editornya menggunakan *Sublime Text* 3, *MySQL* sebagai DBMS untuk *database*, serta *Xampp-win* 32-1.6.8 sebagai servernya.

D. Rumusan Masalah

Berdasarkan identifikasi masalah dan batasan masalah di atas, maka rumusan masalah yang akan dikaji dalam tugas akhir ini yaitu :

1. Bagaimana membangun sebuah perangkat lunak sistem *e-commerce* yg dapat diakses *user* secara *online* ?
2. Bagaimana merancang sebuah sistem *e-commerce* yang dapat melayani sistem penjualan *online* dan sistem manajemen data produk dan laporan. ?
3. Bagaimana konsep perancangan sebuah sistem *e-commerce* menggunakan *Object Oriented Programming* (OOP) ?
4. Bagaimana merancang sistem *e-commerce* menggunakan pemodelan *Unified Modelling Language* (UML) ?
5. Bagaimana melakukan perancangan perangkat lunak sistem *e-commerce* menggunakan bahasa pemrograman PHP *Framework Codeigniter*, *software development code generator atau* editornya menggunakan *Sublime Text* 3, DBMS yang digunakan untuk *database* adalah *MySQL*, serta *Xampp-win* 32-1.6.8 sebagai servernya ?

E. Tujuan Tugas Akhir

Adapun tujuan yang ingin dicapai dari tugas akhir ini adalah :

1. Menghasilkan sebuah perangkat lunak sistem *e-commerce* yang dimanfaatkan secara *online.*

2. Menghasilkan sebuah perangkat lunak sistem *e-commerce* yang dapat melayani sistem penjualan *online* dan sistem manajemen data produk dan laporan.
3. Menghasilkan sebuah sistem *e-commerce* yang dirancang menggunakan konsep *Object Oriented Programming* (OOP).
4. Menciptakan sebuah sistem *e-commerce* yang dirancang menggunakan *Unified Modelling Language* (UML).
5. Menghasilkan sebuah perangkat lunak sistem *e-commerce* yang dirancang menggunakan bahasa pemrograman PHP *Framework Codeigniter*, *software development code generator atau* editornya menggunakan *Sublime Text* 3, DBMS yang digunakan untuk *database* adalah *MySQL*, serta *Xampp-win* 32-1.6.8 sebagai servernya.

F. Manfaat Tugas Akhir

Adapun manfaat yang diharapkan dari penelitian ini yaitu :

1. Bagi Toko A4 Sport Padang
   a. Dapat terpenuhinya informasi tentang tempat dan barang yang dijual oleh Toko A4 Sport Padang kepada masyarakat luas.
   b. Mempermudah pemilik Toko A4 Sport Padang untuk melakukan manajemen data produk dan laporan keuangan.
   c. Meningkatkan tingkat promosi kepada masyarakat luas.
2. Bagi Konsumen
   a. Mempermudah konsumen untuk menemukan informasi untuk analisa sebelum melakukan pembelian.

b. Mempermudah konsumen untuk melakukan pembelian, sehingga jika alamatnya jauh tidak perlu lagi datang ke toko tersebut.

3. Bagi Penulis
   a. Dengan melakukan perancangan perangkat lunak ini, penulis dapat meningkatkan daya kreativitas, dan bisa secara langsung menerapkan ilmu yang telah dipelajari selama ini.
   b. Dapat melakukan penelitian, perancangan dan ikut berpartisipasi dalam penggunaan dan pengembangan teknologi komputerisasi.
   c. Penulis dapat mengetahui lebih banyak tentang perancangan sebuah *website*.
4. Bagi Jurusan Elektronika
   a. Meningkatkan kontribusi sebuah lembaga pendidikan bagi kemajuan informasi untuk masyarakat.
   b. Agar dapat meningkatkan nama besar di masyarakat, sebagai alat bukti bahwa ilmu yang diajarkan kepada mahasiswa benar-benar bermanfaat

BAB II

LANDASAN TEORI

A. Ruang Lingkup *E-Commerce*

1. Pengertian *E-Commerce*

Dengan semakin matangnya teknologi internet dan web, telah mengubah cara berbagai perusahaan untuk mengendalikan bisnis mereka. Dengan semakin meluasnya pengaruh dan dampak internet, dan semakin banyak perusahaan menggunakan internet, kemungkinan untuk mengendalikan perdagangan antar bisnis di internet semakin bertambah, dan semakin menjadi bagian yang utama dari perdagangan barang dan jasa saat ini.

E-commerce sering disebut dengan perdagangan elektronik, merupakan prosedur berdagang atau mekanisme jual-beli di internet dimana pembeli dan penjual akan melakukan transaksi di dunia maya. Menurut Adi Nugroho (2006:5) "perdagangan elektronik (*e-commerce)* didefinisikan sebagai cara untuk menjual dan membeli barang-barang (dan jasa) lewat jaringan internet, tetapi hal itu (tentu saja) mencakup berbagai aspek".

Itu berarti *e-commerce* dapat melibatkan fasilitas elektronik, pertukaran data elektronik, sistem manajemen inventori otomatis, dan sistem pengumpulan data otomatis. Sehingga sistem jual beli *online* dapat dilakukan pada tempat yang berjauhan.

Dari definisi di atas, penulis menyimpulkan bahwa *e-commerce* merupakan suatu proses jual beli produk dan atau jasa yang dilakukan konsumen dari perusahaan dengan menggunakan media *online* sebagai perantara transaksi.

2. Keuntungan *E-Commerce* (Perdagangan Elektronik)

Perdagangan elektronik menawarkan kepada perusahaan keuntungan jangka pendek dan jangka panjang. Perdagangan elektronik tidak hanya membuka pasar baru bagi produk dan / jasa yang ditawarkan, mencapai konsumen baru, tetapi ia juga dapat mempermudah cara perusahaan melakukan bisnis. Disamping itu, perdagangan elektronik juga sangat bermanfaat bagi pelanggan / konsumen dan masyarakat umum. Menurut Adi Nugroho (2006:19) manfaat perdagangan elektronik adalah sebagai berikut :

a) Keuntungan Bagi Perusahaan

1) Memperpendek jarak**.**

   Perusahaan-perusahaan dapat lebih mendekatkan diri dengan konsumen. Dengan hanya mengklik *link-link* yang ada pada situs-situs, konsumen dapat menuju ke perusahaan dimana pun saat itu mereka berada.

2) Perluasan pasar**.**

   Jangkauan pemasaran menjadi semakin luas dan tidak terbatas oleh area geografis dimana perusahaan berada.

3) Perluasan jaringan mitra bisnis**.**

Pada perdagangan tradisional, sangat sulit bagi suatu perusahaan untuk mengetahui posisi georgrafis mitra kerjanya yang berada di negara-negara lain atau benua lain.

4) Efisien**.**

Seperti disinggung dibagian sebelumnya, perdagangan elektronik akan sangat memangkas biaya-biaya operasional.

b) Keuntungan Bagi Konsumen

1) Efektif.

Konsumen dapat memperoleh informasi tentang produk / jasa yang dibutuhkannya dan bertransaksi dengan cara yang cepat dan murah.

2) Aman secara fisik.

Konsumen tidak perlu mendatangi toko tempat perusahaan menjajakan barangnya dan ini memungkinkan konsumen dapat bertransaksi dengan aman sebab di daerah-daerah tertentu mungkin sangat berbahaya jika berkendaraan dan membawa uang tunai dalam jumlah yang besar.

3) Fleksibel.

Konsumen dapat melakukan transaksi dari berbagai lokasi, baik dari rumah, kantor, warnet, atau tempat-tempat lainnya. Konsumen juga tidak perlu berdandan rapi seperti pada perdagangan tradisional umumnya.

c) Keuntungan Bagi Masyarakat Umum

1) Mengurangi polusi dan pencemaran lingkungan.

Dengan adanya perdagangan elektronik yang dapat dilakukan dimana saja, konsumen tidak perlu melakukan perjalanan ke toko-toko, dimana hal ini akan mengurangi jumlah kendaraan yang berlalu-lalang di jalanan.

2) Membuka peluang kerja baru.

Era perdagangan elektronik akan membuka peluang-peluang kerja baru bagi mereka yang tidak 'buta' teknologi. Seperti pemrogram komputer, perancang web, ahli di bidang basis data, analisis sistem, ahli di bidang jaringan komputer, dan sebagainya.

3) Mengutungkan dunia akademis.

Berubahnya pola hidup masyarakat dengan hadirnya perdagangan elektronik, kalangan akademisi akan semakin diperkaya dengan kajian-kajian psikologis, antropologis, sosial-budaya, dan sebagainya.

4) Meningkatkan kualitas sumberdaya manusia.

Perdagangan elektronik, seperti juga teknologi komputer pada umumnya, hanya bisa dilakukan oleh orang-orang yang tidak ***gaptek*** ***(***gagap teknologi), sehingga merangsang orang-orang untuk mempelajari teknologi komputer.

Jadi, keuntungan lainnya bahwa *e-commerce* menawarkan pengurangan sejumlah biaya tambahan karena biaya tersebut tidak digunakan untuk gedung dan pelayanan pelanggan (*costumer service),* jika dibandingkan dengan jenis bisnis tradisional. Hal ini membantu perusahaan dalam meningkatkan keuntungannya. Salah satu jenis bisnis yang mengambil keuntungan dari *e-commerce* adalah perbankan.

3. Kerugian Perdagangan Elektronik

Sebagai media *online,* selain mempunyai keuntungan, perdagangan elektronik juga mempunyai kerugian. Karena sesuatu ciptaan manusia pasti memiliki kekurangan dan sisi negatif. Namun, dari sudut pandang manapun, perdangan elektronik memiliki sisi keuntungan yang lebih banyak dari pada sisi kerugiannya. Menurut pendapat Adi Nugroho (2006:22) segi negatif perdagangan elektronik sebagai berikut :

a) Meningkatkan Individualisme

Pada perdagangan elektronik, seseorang dapat bertransaksi dan mendapatkan barang/jasa yang diperlukannya tanpa perlu bertemu dengan siapa pun.

b) Terkadang Menimbulkan Kekecewaan

Apa yang dilihat di layar monitor komputer kadang berbeda dengan apa yang dilihat secara kasat mata. Seseorang yang membeli lukisan di internet mungkin suatu saat akan mendapati

lukisannya tidak memiliki warna yang sama dengan apa yang dilihatnya di layar monitor.

c) Tidak Manusiawi

Sering kali orang pergi ke toko-toko dan pusat-pusat perbelanjaan (*mall)* tidak sekedar ingin memuaskan kebutuhannya akan barang/jasa tertentu. Ia mungkin melakukannya untuk penyegaran (*refreshing).* Perdagangan elektronik gagal dipandang dari sudut pandang seperti ini.

4. Pemesanan

Pemesan merupakan hal yang rutin bahwa para konsumen secara elektronik mengirimkan pesanan-pesanan akan barang dan/atau jasa ke perusahaan. Menurut Adi Nugroho (2006:15) "*Form* elektronik merupakan *form* yang mirip dengan surat pesanan pada perdagangan tradisional".

Aplikasi-aplikasi klien/server sering dirancang untuk menanganinya. Web menuntut campur tangan yang relatif sedikit dari personal-personal perusahaan, namun karena kebanyakan sistem web mendukung *form-form* elektronik, kebanyakan perusahaan yang bergerak di bidang perdagangan elektronik saat ini menggunakan aplikasi web dan basis data untuk tujuan yang sama.

5. *Promotion* (Promosi) *E-Commerce*

Fungsi penjualan dan pemasaran merupakan entitas yang terpisah dalam rantai nilai berorientasi konsumen. Menurut Adi

Nugroho (2006:210), "*Promotion* (promosi) dalam ilmu pemasaran tradisional, merupakan suatu upaya perusahaan agar semua fungsi-fungsi pemasaran suatu saat nanti akan berakhir dengan tahapan penjualan".

Adi Nugroho (2006:210) juga menjelaskan pesan-pesan ini mungkin dikomunikasikan dengan berbagai cara, yaitu :

a) Saluran periklanan komersial.

b) Cerita-cerita dan berita-berita pers.

c) Kalimat-kalimat dari mulut ke mulut.

d) Pengalaman konsumen secara pribadi.

e) Pengemasan.

B. Rekayasa Perangkat Lunak (RPL)

1. Pengertian RPL

Perangkat lunak (*software)* adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan *(user manual).*

Menurut Rosa A. S. & M. Shalahuddin (2013:4) "Rekayasa perangkat lunak (*software engineering)* merupakan pengembangan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin".

Jadi rekayasa perangkat lunak adalah sebuah pengembangan dari konsep rekayasa yang sudah ada untuk menghasilkan sebuah perangkat lunak yang bernilai lebih dari yang sebelumnya.

2. Kriteria Rekayasa Perangkat Lunak

Rekasaya perangkat lunak lebih fokus pada praktik pengembangan perangkat lunak dan mengirimkan perangkat lunak yang bermanfaat kepada pelanggan (*costumer*). Rosa A. S. & M. Shalahuddin (2013:5) juga menjelaskan rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut :

a) Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*maintainability).*

b) Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (*dependability* dan *robust).*

c) Efisien dari segi sumber daya dan penggunaan.

d) Kemampuan untuk dipakai sesuai dengan kebutuhan (*usability).*

Dari kriteria di atas, maka perangkat lunak yang baik adalah perangkat lunak yang dapat memenuhi kebutuhan pelanggan atau *user* atau berorientasi pada pelanggan atau pemakai perangkat lunak, bukan berorientasi pada pembuat atau pengembang perangkat lunak.

3. Tantangan Rekayasa Perangkat Lunak

Dalam hal pengembangannya, rekayasa perangkat lunak juga memiliki tantangan yang dihadapi dari prosenya. Rosa A. S. & M. Shalahuddin (2013:7) juga menguraikan tantangan rekayasa perangkat lunak sebagai berikut :

a) Tantangan warisan dimana perangkat lunak dikembangkan selama bertahun-tahun oleh orang-orang yang berbeda, hal ini dapat menyebabkan ketidakpahaman atau perubahan tujuan pembuatan perangkat lunak

b) Tantangan heterogenetis dimana perangkat lunak harus dapat beradaptasi dengan teknologi yang terus berkembang dengan semakin luasnya lingkungan distribusi perangkat lunak

c) Tantangan pengiriman bahwa perangkat lunak dengan sekala besar dan kompleks sekalipun dapat sampai ke tangan pelanggan (*costumer)* atau *user* dengan cepat dan kualitas tetap terjaga.

4. Proses Rekayasa Perangkat Lunak

Proses rekayasa perangkat lunak dilakukan selama pembangunan perangkat lunak. Rosa A. S. & M. Shalahuddin (2013:11) juga menjelaskan proses perangkat lunak sebagai berikut :

a) Pengumpulan spesifikasi (*spesification).* Mengetahui apa saja yang harus dapat dikerjakan sistem perangkat lunak dan batasan pengembangan perangkat lunak.

b) Pengembangan (*development).* Pengembangan perangkat lunak untuk menghasilkan sistem perangkat lunak.

c) Validasi (*validation).* Memeriksa apakah perangkat lunak sudah memenuhi keinginan pelanggan (*costumer).*

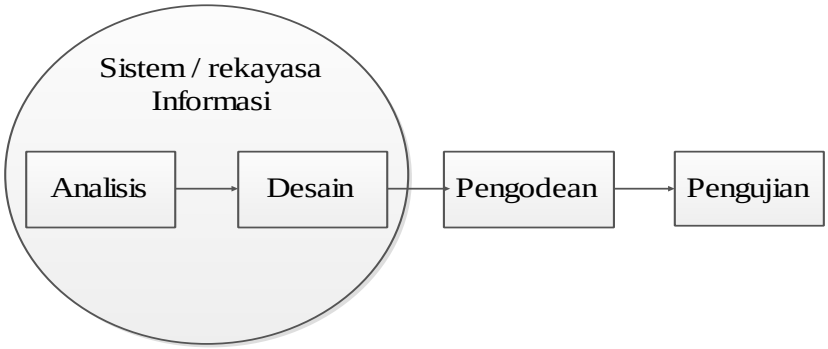
d) Evolusi (*evolution).* Mengubah perangkat lunak untuk memenuhi perubahan kebutuhan pelanggan (*costumer).*

5. Pengembangan RPL Model *Waterfall*

Model SLDC air terjun (*waterfall)* sering juga disebut model sekuensial linier (*sequential linear)* atau alur hidup klasik (*classic life cycle).* Menurut Martin Fowler (2005:28) "Gaya *waterfall* adalah memecah sebuah proyek berdasarkan aktivitas. Adapun kegiatan untuk membuat perangkat lunak yaitu analisis persyaratan, desain, *coding*, dan *testing*".

Sedangkan menurut Rosa A. S. & M. Shalahuddin (2013:28) model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support).*

Berikut adalah gambar model airt terjun :



Gambar 1. Ilustrasi Model *Waterfall*

Dari gambar di atas dijelaskan urutan pemodelan *waterfall* yang dimulai dari analisis, desain, pengodean dan pengujian.

a) Analisis kebutuhan perangkat lunak. Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak.

b) Desain. Desain perangkat lunak adalah prosesmulti langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.

c) Pembuatan kode program. Desain harus ditranslasikan ke dalam program perangkat lunak.

d) Pengujian. Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji.

e) Pendukung (*support)* atau pemeliharaan (*maintenance).* Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user.*

C. Perancangan *Unified Modelling Language* (*UML)*

1. Pengertian UML

Dalam buku yang berjudul UML *Distilled* Panduan Singkat Bahasa Pemodelan Objek Standar Martin Fowler (2005:1) menjelaskan bahwa "*Unified Modelling Language* (*UML)* adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang

membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO)".

Sedangkan menurut Rosa A. S. & M. Shalahuddin (2013:137) "UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung."

Jadi UML adalah bahasa yang digunakan untuk pemodelan dan desain rancangan perangkat lunak dalam bentuk diagram dan teks untuk menggambarkan rancangan.

2. Diagram-diagram dalam UML

Pada saat membuat desain, diperlukan teknik untuk mendokumentasikan perangkat lunak sejak pertama kali ditulis. Dokumentasi ini dapat menolong orang-orang untuk memahami perangkat lunak tersebut saat mereka harus menelaahnya tapi tidak menguasai kode-kodenya. Rosa A. S. & M. Shalahuddin (2013:141) menjelaskan diagram UML sebagai berikut :

a) ***Use-case Diagram***

Use case diagram merupakan pemodelan untuk melakukan *(behavior)* sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Syarat penamaan pada *use case* adalah didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

1) Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

2) *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

b) ***Activity Diagram***

Activity diagram menggambarkan *work flow* (aliran kerja) atau aktivitas sari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

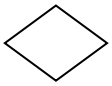
1) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.

2) Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki antarmuka tampilan.

3) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Berikut ini tabel simbol pada *activity diagram* :

Tabel 1. Simbol-simbol Pada *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
2		Percabangan / *Decision*	Asosiasi percabangan dimana jika ada piliha aktivitas biasanya diawali dengan kata kerja.
3		Status Awal	Status awal yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status awal.

Tabel di atas menjelaskan jumlah simbol yang terdapat pada *activity diagram,* untuk menjelaskan aktivitas yang dilakukan oleh sistem.

D. Konsep Dasar Pemrograman Berorientasi Objek

1. Pengertian Pemrograman Berorientasi Objek

Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat

lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya.

Menurut Rosa A. S. & M. Shalahuddin (2013:100) “Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas”. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek.

2. Keuntungan Menggunakan Metodologi Berorientasi Objek

Dalam buku yang berjudul Rekayasa perangkat Lunak, Rosa A. S. & M. Shalahuddin (2013:100) menjelaskan keuntungan menggunakan metodologi berorientasi objek sebagai berikut :

a) Meningkatkan produktivitas

b) Kecepatan pengembangan

c) Kemudahan pemeliharaan

d) Adanya konsistensi

e) Meningkatkan kualitas perangkat lunak

3. Konsep Dasar Pemrograman Berorientasi Objek

Rosa A. S. & M. Shalahuddin (2013:104) juga menjelaskan konsep dasar yang harus dipahami tentang metodologi berorientasi objek :

a) Kelas (*class*)

b) Objek (*object)*

c) Metode (*method)*

d) Atribut (*attribute)*

e) Abstraksi (*abstraction)*

f) Antarmuka (*interface)*

g) Komunikasi Antar Objek

h) *Package*

E. *Database* (Basis Data)

1. *Database* MySQL

Database MySql merupakan sistem manajemen basis data SQL yang sangat terkenal dan bersifat *open source*. MySql dibangun, didistribusikan, dan didukung oleh MySql AB. MySql AB merupakan perusahaan komersial yang dibiayai oleh pengembang MySql.

Menurut Wahana Komputer (2006:2) fitur utama dari MySql adalah :

a) Ditulis dalam bahasa C dan C++

b) Bekerja dalam berbagai *platfrom*

c) Menyediakan mesin penyimpanan transaksi dan non transaksi

d) Server tersedia sebagai program yang terpisah untuk digunakan pada jaringan *client/server*

e) MySql mempunyai *library* yang ditempel pada aplikasi

f) Mempunyai sistem *password* yang fleksibel dan aman

g) Dapat menangani basis data dalam skala besar

h) Klien dapat terkoneksi ke MySql server menggunakan *socket* TCP/IP pada platform manapun.

i) Server dapat mengirim pesan error ke klien dalam berbagai bahasa.

2. DBMS (*Database Management System*)

DBMS atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data. Menurut Rosa A. S. & M. Shalahuddin (2013:45) suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut:

a) Menyediakan fasilitas untuk mengelola akses data

b) Mampu menangani integritas data

c) Mampu menangani akses data yang dilakukan secara

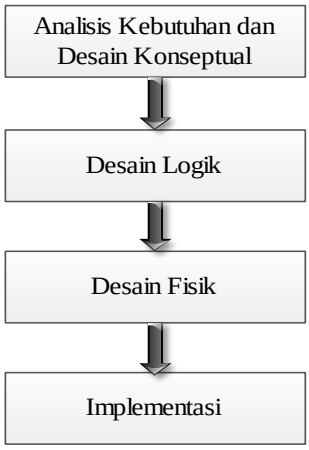
d) Mampu menangani *backup* data

Karena pentingnya data bagi suatu organisasi/perusahaan, maka hampir sebagian besar perusahaan memanfaatkan DBMS dalam mengelola data yang mereka miliki. Pengelolaan DBMS sendiri biasanya ditangani oleh tenaga ahli yang spesial menangani DBMS yang disebut sebagai DBA (*Database Administrator*).

Pada buku karangan Wahana Komputer (2006:3) dijelaskan bahwa “MySql termasuk jenis RDBMS (*Relational Database Management System*)”.

3. Alur Hidup *Database*

Tidak hanya perangkat lunak yang memiliki alur hidup, dalam membuat perencanaan basis data juga memiliki alur hidup atau *Database Life Cycle* (DBLC). Alur hidup basis data dapat dilihat pada gambar berikut :



Gambar 2. Alur Hidup Basis Data

Gambar di atas menjelaskan tentang alur hidup basis data. Dimana sebelum melakukan desain lojik, harus dilakukan analisis kebutuhan dan desain konseptual. Barulah setelah itu melakukam desain fisik dan melakukan implementasin.

Menurut Rosa A. S. & M. Shalahuddin (2013:49), fase-fase DBLC antara lain :

a) Analisis Kebutuhan / *Requirement Analysis*

Hal-hal yang harus dilakukan pada tahap ini adalah:

1) Didefinisikan dengan mewawancarai produsen dan pemakai data, data apa sajakah yang butuh untuk disimpan dan terkait dengan aplikasi komputer yang akan dikembangkan.
2) Membuat kontrak spesifikasi basis data
3) *Entity Relationship Diagram* (ERD) sebagai bagian dari desain konseptual

b) Desain Lojik Basis Data / *Logic Database Design*

Pada tahap ini harus dibuat rancangan lojik basis data. Bisanya pada tahap ini dibuat *Conceptual Data Model* (CDM).

c) Desain Fisik Basis Data / *Physical Database Design*

Pada tahap ini harus dibuat rancangan fisik basis data. Biasanya pada tahap ini dibuat *Physical Data Model* (PDM).

d) Implementasi

- Membuat *Query* SQL
- Aplikasi ke DBMS atau *file*

4. Normalisasi

Normalisasi adalah langkah-langkah sistematis untuk menjamin bahwa struktur *database* memungkinkan untuk *general purpose query* dan bebas dari *insertion, update* dan *deleting anomalies* yang dapat menyebabkan hilangnya integritas data. Normalisasi dilakukan untuk memperbaiki desain tabel yang kurang baik, sehingga penyimpanan data menjadi lebih efisien dan bebas anomali data. Normalisasi

dilakukan dengan cara menerapkan bentuk-bentuk normal secara bertahap dari level terendah sampai level yang dikehendaki.

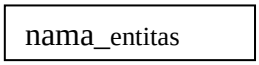
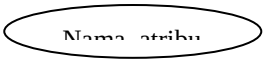
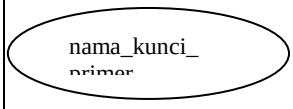
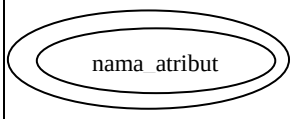
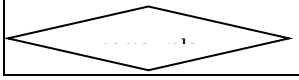
5. *Entity Relationship Diagram* (ERD)

Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan ERD. ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.

Menurut Rosa A. S. & M. Shalahuddin (2013:50) ERD memiliki beberapa aliran notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crows Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen.

Berikut adalah simbol-simbol pada ERD dengan notasi Chen :

Tabel 2. Simbol-Simbol ERD

Simbol	Nama	Deskripsi
	Entitas / *Entity*	Entitas merupakan data inti yang akan disimpan
	Atribut	*Field* atau kolom data yang butuh disimpan dalam suatu entitas
	Atribut Kunci primer	*Field* atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses *record*; biasanya berupa id
	Atribut Multinilai	*Field* atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
	Relasi	Relasi yang menghubungkan antar

Tabel di atas merupakan tabel yang berisikan simbol-simbol yang digunakan pada perancangan ERD. Setiap simbol mempunyai nama dan kategori yang berbeda. Dimana pada sebuah ERD terdiri dari nama entitas, nama atribut, nama kunci primer, dan nama relasi.

F. *Tools* Pendukung

1. Xampp

Perangkat lain yang digunakan untuk perancangan aplikasi adalah *xampp. Xampp* mempunyai beberapa aplikasi yang dibutuhkan untuk proses perancangan. Menurut Bunafit Nugroho (2012:1) "*xampp* adalah *software triad* yang berisi *software Apache,* PHP, dan *MySql* dalam sekali *installasi*".

Ini berarti *xampp* mempunyai kelebihan kelengkapan tersendiri. Seperti *user* dapat mengaplikasi *MySql* untuk perancangan *database*, dan *apache* untuk *web* server aplikasi yang dirancang. Hanya dengan melakukan 1 kali installasi, *user* dapat memanfaatkan beberapa aplikasi yang tersedia di dalam *xampp* tersebut.

2. *Framework*

a) Pengertian *Framework*

Framework merupakan sekumpulan perintah dengan aturan-aturan tertentu yang saling berkaitan untuk pembuatan sebuah *website*. Hal ini berkaitan dengan pendapat Awan Pribadi Basuki (2014:13) bahwa "*Framework* dapat diartikan sebagai kumpulan potongan-potongan program (kelas dan fungsi) yang disusun dan

diorganisasikan sedemikian rupa, sehingga dapat digunakan kembali untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal".

Framework memungkinkan untuk membangun sebuah aplikasi yang lebih cepat karena *developer* hanya akan memfokuskan pada pokok permasalahan yang diminta untuk sebuah aplikasi, sedangkan hal-hal penunjang lainnya seperti koneksi *database*, *form validation*, GUI dan *security*, umumnya telah disediakan oleh *framework*.

b) Alasan Memilih *Framework*

Menurut Wahana Komputer (2011:2) alasan mengapa menggunakan *framework*, sebagai berikut:

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi web.
2. Memudahkan dalam proses *maintenance* karena sudah ada pola tertentu dalam sebuah *framework* (dengan syarat *programmer* mengikuti pola standar yang ada).
3. Umumnya *framework* menyediakan fasilitas-fasilitas yang umum dipakai sehingga Anda tidak perlu membangun dari awal.
4. Lebih bebas dalam pengembangan jika dibandingkan CMS.

3. *Codeigniter*

a) Pengertian *Codeigniter*

Codeigniter merupakan sebuah bahasa pemrograman *web* yang dikembangkan dari bahasa pemroraman PHP yang berbasis *OOP* (*Object Oriented Programming*) yang memiliki *class* dan *function*. Sesuai dengan pendapat Wahana Komputer (2011:2) bahwa "*Codeigniter* adalah aplikasi *open source* yang berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *web*site dinamis dengan menggunakan PHP".

Sedangkan menurut Riyanto (2011:1) "*Codeigniter* (selanjutnya disebut CI) adalah sebuah *framework* yang digunakan untuk membuat sebuah aplikasi berbasis web yang disusun dengan menggunakan bahasa PHP".

Pendapat lain menurut Awan Pribadi Basuki (2014:23) "*Codeigniter* adalah *framework* PHP yang dibuat berdasarkan "*design pattern" Model-View-Controller* atau biasa disingkat MVC. *Design pattern* sendiri adalah sekumpulan penjelasan mengenai metode-metode bagaimana cara menyelesaikan suatu masalah yang umumditemui dalam proses perancangan perangkat lunak (*software design)".*

Jadi *codeigniter* merupakan sebuah aplikasi *framework* dari PHP yang mempunyai desain tersendiri yang digunakan untuk membangun sebuah *website. Codeigniter* adalah *framework* PHP

yang mudah untuk dikembangkan. Sehingga sebagai pemula, tidak akan kesulitan untuk menggunakan *framework* ini. Karena selain mudah digunakan, *framewrok codeigniter* juga cepat dalam perancangan sebuah sistem aplikasi.

b) Kelebihan *Codeigniter*

Di samping menerapkan konsep MVC, *codeigniter* juga mempunyai kelebihan. Wahana Komputer (2011:4) juga menjelaskan bahwa beberapa kelebihan *CodeIgniter* (CI) dibandingkan dengan *framework* PHP lain, sebagai berikut:

a) Performa sangat cepat. Salah satu alasan tidak menggunakan *framework* adalah karena eksekusinya yang lebih lambat dari pada PHP, *from the scratch.*

b) Konfigurasi yang sangat minim (*Nearly Zero Configuration).* Tentu saja untuk menyesuaikan dengan *database* dan keleluasaan *routing* tetap diizinkan melakukan konfigurasi dengan mengubah beberapa *file* konfigurasi seperti *database*, *php* atau *autoload.php.*

c) Banyak komunitas. Dengan banyaknya komunitas *codeigniter* ini, memudahkan Anda untuk berinteraksi dengan yang lain, baik itu bertanya atau mengetahui perkembangan terbaru.

d) Dokumentasi yang sangat lengkap. Setiap paket installasi *codeigniter* sudah disertai *user guide* yang sangat bagus dan

lengkap untuk dijadikan permulaan. Bahasa yang digunakan pun mudah dipahami.

Awan Pribadi Basuki (2014:14) menjelaskan beberapa alasan mengapa menggunakan *codeigniter* yaitu :

a) *Codeigniter* termasuk *framework* sederhana dengan ukuran kecil, namun dengan kemampuan besar.
b) *Codeigniter* dieksekusi dengan cepat dan memiliki *performance* yang handal.
c) Menerapkan konsep *Mode-View-Controller* (MVC).
d) *Codeigniter* menyediakan *Library* dan *Helper* yang cukup lengkap.
e) Dokumentasi *codeigniter* lengkap dan jelas.

c) Kelebihan *Framework Codeigniter*

Sedangkan menurut Riyanto (2011:3) kelebihan *framework codeigniter* sebagai berikut :

a) Berukuran kecil

Hanya dengan sekitar 4MB, seorang pengembang pemula sudah bisa menggunakan CI yang di dalamnya terdapat berbagai fitur andal yang bisa digunakan untuk membangun aplikasi, mulai dari skala kecil samapai skala *enterprise*.

b) Sangat cepat

Sangat cepat di sini dalam eksekusi *file*. Kecepatan ini memungkinkan CI membangun aplikasi berskala *enterprise*.

c) Sederhana (*simple*)

Anda harus mencobanya terlebih dahulu maka Anda akan sadar betapa sederhananya CI, baik dari sisi sintaks yang digunakan maupun dari sisi alur penyusunan program.

d) Bisa langsung digunakan

Setiap orang akan bisa menggunakan CI. Hanya tinggal mengunduh, mengekstrak *file,* dan meletakkan pada *folder* kerja *web* server Anda (**htdoc** atau **public html**) maka Anda akan bisa langsung mencobanya.

e) Bisa dikombinasikan dengan beberapa aplikasi dengan cara yang sangat mudah.

f) Kode program rapi

Kode program Anda akan menjadi sangat rapi, karena penggunaan sistem MVC yang mengelompokkan kode sesuai dengan fungsinya masing-masing.

4. *Model, View, Controller (MVC)*

MVC adalah sebuah pendekatan yang ditempuh untuk memisahkan aplikasi menjadi 3 bagian, yaitu *Model, View*, dan *Controller.* Awan Pribadi Basuki (2014:23) juga menjelaskan bahwa MVC memberikan struktur kepada aplikasi, sehingga dapat dicapai "*code reusability".*

a) ***Model,*** merepresentasikan data yang digunakan aplikasi. Contoh data misalnya *database,* RSS, atau data yang diperoleh

dari pemanggilan API, dan aksi yang melibatkan operasi *Create, Read, Update,* dan *Delete (CRUD)* data.

b) ***View,*** adalah informasi yang ditampilkan kepada *user* melalui *browser.* Biasanya berupa *file* HTML atau kode PHP yang menyusun *template* untuk sebuah website. Pada *codeigniter, view* dapat berupa bagian-bagian sebuah halaman, *template,* atau jenis lain dari halaman atau *template.*

c) ***Controller,*** adalah "*business logic"* bertugas sebagai jembatan antara *Model* dan *View. Controller* akan merespon HTTP *request* yang datang dari *user* (melalui *browser),* dari *request ini controller* akan menentukan apa yang harus dilakukan.

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil Perancangan Perangkat Lunak Sistem *E-Commerce* (Studi Kasus Toko A4 Sport Padang) dapat disimpulkan sebagai berikut :

1. Perangkat lunak sistem *e-commerce* yang dirancang, setelah diuji coba telah berhasil dengan baik.
2. Pada perangkat lunak yang telah diimplementasikan, sistem penjualan *online* dan sistem manajemen data produk dan laporan telah dapat digunakan dengan baik.
3. Konsep *Object Oriented Programming* (OOP) yang diimplementasikan pada perancangan sistem *e-commerce* telah berhasil dengan baik.
4. Pemodelan *Unified Modelling Language* (UML) yang diimplementasikan pada perancangan sistem *e-commerce* telah berhasil baik.
5. Bahasa pemrograman PHP *Framework Codeigniter*, *software development code generator atau* editornya menggunakan *Sublime Text* 3, *MySQL* sebagai DBMS untuk *database*, serta *Xampp-win* 32-1.6.8 sebagai servernya, yang digunakan pada perancangan perangkat lunak sistem *e-commerce* telah berhasil baik.

B. Saran

Adapun saran setelah merancang sistem *e-commerce* ini adalah:

1. Agar menghasilkan sebuah sistem *e-commerce* yang lebih komplit, sebaiknya sistem dapat bekerja sama dengan sistem *online* kurir yang dibutuhkan.
2. Agar konsumen lebih merasa puas, ada bagusnya pada sistem ditambahkan *form* untuk konsumen memilih dan meng*input*kan kurir yang diinginkan.
3. Pada pengembangan sistem *e-commerce* selanjutnya, sebaiknya perancangan untuk pesan admin dan *supplier* lebih baik lagi, seperti perancangan *chat* yang lebih *smart* dan 1 *form* merangkap *inbox* dan *outbox.*

DAFTAR PUSTAKA

Adi Nugroho. 2006. *E-Commerce Memahami Perdagangan Modern di Dunia Maya*. Bandung : Informatika Bandung

Awan Pribadi Basuki. 2014. *Proyek Membangun Website Berbasis PHP dengan CodeIgniter*. Yogyakarta : Lokomedia

Bunafit Nugroho. 2012. *Panduan Membuat Program Toko Dengan PHP, MySql dan Dreamweaver Point Of Sale (POS)*. Yogyakarta : Alif Media

Fowler, Martin. 2005. *UML DISTILLED Edisi 3.* Yogyakarta:Andi

Riyanto. 2011. *Membuat Sendiri Aplikasi E-Commerce dengan PHP & MySql menggunakan Codeigniter & JQuery.* Yogyakarta: Andi

Rosa A.S. & M. Shalahuddin. 2013. *Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek).* Bandung : Informatika Bandung

Universitas Negeri Padang. 2010. *Buku Panduan Penulisan Tugas Akhir/Skripsi* Padang : Universitas Negeri Padang.

Wahana Komputer. 2006. *Pengolahan Database Dengan MySql*. Yogyakarta : Andi

Wahana Komputer. 2011. *Mudah dan Cepat Membuat Website Dengan CodeIgniter.* Yogyakarta : Andi