

MAKALAH SEMINAR TUGAS AKHIR
SIMULASI E-COMMERCE DENGAN JSP (JAVA SERVER PAGES)

Oleh :
Suhartoko
L2F301471

Abstraksi

Perkembangan teknologi elektronis terutama internet, telah mengubah pola pikir manusia untuk melakukan kegiatan yang menyangkut kebutuhan akan informasi dan jasa. Dengan jangkauan internet yang dapat diakses dari segala penjuru dunia, dapat dimanfaatkan sebagai media pemasaran yang sangat baik bagi produsen suatu produk.

Electronic Commerce (E-commerce) adalah suatu cara baru untuk melakukan perdagangan dengan memanfaatkan medium elektronis yang telah memiliki standar yang universal dan tersebar luas seperti halnya Internet. Pendekatan ini akan memberikan nilai tambah kepada semua pihak yang terlibat dalam mata rantai perdagangan tersebut. Kemudahan dalam memilih dan membeli barang adalah juga salah satu nilai tambah yang didapatkan pihak pembeli/pelanggan dalam mekanisme perdagangan elektronis tersebut.

Java Server Pages (JSP) merupakan bagian dari teknologi Java 2 Enterprise Edition (J2EE) yang berfungsi untuk membangun suatu aplikasi web.

E-Commerce dengan JSP ini dibuat untuk memberikan alternatif pada para programmer untuk membuat aplikasi E-Commerce menggunakan bahasa pemrograman Java dengan kelebihan yang dimilikinya seperti open source dan multi platform. Dari sistem E-Commerce yang dibuat dengan JSP ini pelanggan dapat melihat buku-buku dari daftar katalog yang ada, mengisi daftar belanja pada shopping cart. Pada saat pelanggan memutuskan untuk membeli buku yang ada pada keranjang belanjanya, maka pelanggan harus login terlebih dahulu atau mendaftarkan dirinya supaya bisa melakukan order pembelian. Setelah pelanggan melakukan order, order akan dimasukkan ke database, keranjang belanja pelanggan akan dikosongkan dan halaman terima kasih akan ditampilkan beserta nomor order pelanggan yang bersangkutan.

I. PENDAHULUAN

Pada generasi pertama, Internet hanya digunakan sebagai sarana pemasaran belaka. Hal ini ditandai dengan munculnya *homepage-homepage* perusahaan yang berisi informasi tentang perusahaan tersebut. Kemudian pada generasi kedua, Internet digunakan sebagai sarana transaksi bisnis jual-beli. Hal ini ditandai dengan munculnya situs-situs *e-commerce*. Selanjutnya pada generasi ketiga, Internet digunakan sebagai sarana utama dalam bisnis internal maupun eksternal pada suatu perusahaan^[3].

Keberhasilan implementasi *e-commerce* pada suatu perusahaan, salah satunya ditentukan oleh penggunaan teknologi yang digunakan untuk mengembangkannya. Pada saat ini terdapat banyak pilihan teknologi yang dapat digunakan untuk mengembangkan aplikasi *e-commerce*, baik yang gratis maupun yang membutuhkan biaya yang sangat mahal. Faktor-faktor lain yang harus diperhatikan selain biaya yaitu masalah kecepatan, keandalan, serta keamanan pada teknologi server-side yang digunakan.

Di antara beragamnya teknologi yang tersedia, penulis menganggap ada salah satu kekuatan besar yang dapat dipertimbangkan untuk mengembangkan aplikasi *e-commerce* yaitu teknologi Java (J2EE). *Java Server Pages* (JSP) merupakan bagian dari J2EE yang dibuat dan didefinisikan oleh kalangan industri tepatnya Java Community Process yang dipimpin oleh Sun Microsystems^[6].

Teknologi JSP memiliki berbagai sifat dan karakter yang membuat JSP memiliki banyak

kelebihan daripada bahasa pemrograman web yang lain. Salah satu kelebihan Java yang terkenal yaitu "Write Once, Run Anywhere" sehingga teknologi Java memiliki portabilitas yang tinggi^[1,2]. Para programmer Java tidak perlu khawatir dengan sistem operasi yang digunakan karena Java dapat berjalan di semua sistem operasi termasuk Linux, Windows dan Macintosh^[6].

Dengan berbagai kelebihan yang dimiliki Java, maka penulis mencoba mengimplementasikan teknologi Java dalam pembuatan situs *e-commerce*. Dengan dibuatnya situs *e-commerce* berbasis Java ini diharapkan, tidak ada lagi kekhawatiran mengenai kompatibilitas antara sistem operasi yang ada.

1.1. Maksud dan Tujuan

Maksud dan tujuan dibuatnya tugas akhir ini yaitu :

Merancang simulasi sistem *e-commerce* untuk aplikasi perdagangan elektronis pada suatu toko buku menggunakan Java Server Pages (JSP) yang berbasis bahasa pemrograman Java.

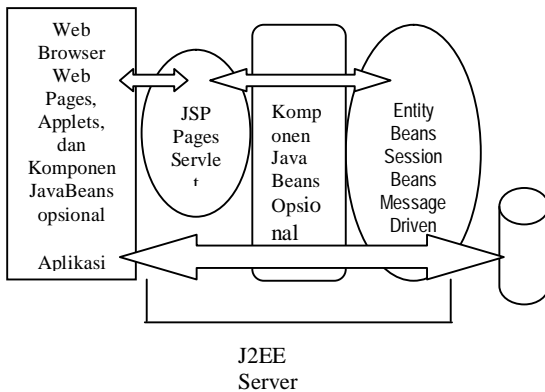
1.2 Batasan Masalah

Pembatasan masalah dari penyusunan tugas akhir ini adalah :

1. Sistem *e-commerce* yang dibuat hanya dengan teknologi Java tanpa adanya grafis dan multimedia yang terintegrasi secara penuh.
2. Sistem *e-commerce* yang dibuat tidak mengimplementasikan penggunaan

2.5. Komponen Bisnis

Business code atau *Business logic*, merupakan bagian *logic* yang digunakan untuk memecahkan suatu permasalahan yang sesuai dengan kebutuhan pada bisnis tertentu seperti perbankan, retail atau *finance*. Komponen bisnis di-handle oleh *Enterprise bean* yang berjalan pada *business tier*. Gambar 2.5^[3] menunjukkan bagaimana *Enterprise bean* menerima data dari aplikasi *client*, memprosesnya (jika diperlukan) dan mengirimkannya ke *enterprise information system tier* untuk disimpan. *Enterprise bean* juga menerima data dari penyimpanan data (*storage*), memprosesnya (jika diperlukan) dan mengirimkannya kembali ke aplikasi *client*.



Gambar 2. Komponen Bisnis

Terdapat tiga jenis *Enterprise beans* yaitu *session beans*, *entity beans* dan *message-driven beans*. *Session bean* mempresentasikan komunikasi sementara dengan *client*. Pada saat *client* mengakhiri eksekusi, *session bean* dan datanya akan lenyap. Sebaliknya, *entity bean* mempresentasikan *persistent* data yang disimpan pada sebuah baris di sebuah tabel database. Jika *client* mengakhiri atau jika server *shut down*, *service* tertentu menjamin bahwa data *entity bean* telah disimpan. *Message-driven bean* mengkombinasikan fitur pada *session bean* dengan *Java Message Service (JMS) message listener*^[2,3].

2.6. Java Server Pages (JSP)

Tujuan dari adanya spesifikasi JSP dalam J2EE API adalah untuk menyederhanakan pembuatan dan manajemen halaman web yang bersifat dinamis, dengan cara memisahkan content dan presentasi dari halaman web tersebut^[3,8]. Pada umumnya JSP merupakan halaman yang menggabungkan HTML atau XML dengan *scripting tag* dan program Java yang terdapat dalam tag tersebut. Pada saat JSP dipanggil oleh *client* melalui web-browser, JSP tersebut akan di-*compile* terlebih dahulu menjadi sebuah *servlet*, kemudian hasilnya dikembalikan kepada *client*.

Penjelasan berikut mengenai kelebihan JSP dibandingkan teknologi web lain, yang menjadi pertimbangan sebagai bahasa pemrograman untuk aplikasi web yang tangguh^[3,8].

1. Kelebihan JSP

a. Memisahkan presentasi statis dan isi yang dinamik

Untuk menghasilkan aplikasi web tentunya web developer harus berurusan dengan tag-tag HTML maupun XML untuk menghasilkan halaman web. Agar menghasilkan tampilan halaman web yang baik, tentunya web developer juga harus mendesainnya sehingga harus menentukan kode HTML yang ingin dihasilkan.

Dengan teknologi JSP hal ini dapat dilakukan, dimana dengan JSP maka web programmer dapat menisipkan tag atau skrip dengan data atau isi dinamik yang akan ditampilkan pada bagian-bagian dari halaman web yang telah didesain.

b. Menekankan komponen *reusable*

Teknologi JSP memerlukan komponen yang *reusable* untuk melakukan pemrosesan yang lebih kompleks. Dengan komponen (Java bean), developer dapat menggunakannya untuk operasi yang umum sehingga memungkinkan *sharing* dan distribusi komponen kepada publik atau komunitas di Internet. Penggunaan komponen dapat mempercepat pembuatan aplikasi web karena proses logik yang diperlukan sudah tersedia dan langsung dapat digunakan.

c. Berbasis bahasa pemrograman Java

Oleh karena JSP berbasis Java, maka aplikasi yang di buat dengan JSP juga memiliki manajemen memori dan sekuritas yang baik. Selain itu, JSP juga mudah dipelajari dan dapat memanfaatkan pemrograman berorientasi objek dari Java.

d. Bagian dari platform Java

Oleh karena merupakan bagian dari platform Java, maka JSP juga memiliki karakteristik “*Write Once, Run Anywhere*” yaitu portabilitas yang tinggi.

f. Terintegrasi dengan J2EE

Oleh karena JSP merupakan bagian integral J2EE, maka aplikasi JSP dapat dikembangkan ke aplikasi berskala *Enterprise*.

Dengan kelebihan-kelebihannya, maka JSP sudah banyak diterima dan diimplementasikan oleh berbagai industri secara online, di antaranya yaitu, *American Century*, *BMW Group Sverige*, *ClickRewards*, *First USA* dan masih banyak lagi^[3].

2.7 Penggunaan JSP

Halaman JSP biasanya digunakan untuk membuat struktur dari suatu data. JSP sangat cocok diterapkan dimana isi dari suatu data berubah setiap ada permintaan tetapi struktur data tidak mengalami perubahan atau berubah sangat sedikit. Berikut ini penggunaan JSP yang tepat :

1. Untuk Presentasi Data

Halaman JSP sangat cocok digunakan untuk data aplikasi dengan isi yang terstruktur seperti HTML. JSP juga dapat digunakan pada halaman web yang tetap

dengan beberapa elemen yang berubah secara dinamik pada saat *runtime*.

2. Untuk Menjalankan XML

Halaman JSP adalah teknologi yang sangat bagus untuk menjalankan XML dengan struktur yang tetap. JSP berguna untuk menjalankan pesan XML dalam format yang standar dimana tag pesan adalah tetap dan hanya nilai attribute atau karakter data yang berubah setiap halaman tersebut dipanggil.

3. Untuk Menjalankan Data Yang Tidak Terstruktur.

JSP juga dapat digunakan untuk menjalankan isi yang tidak terstruktur seperti kode ASCII dan POSTScript.

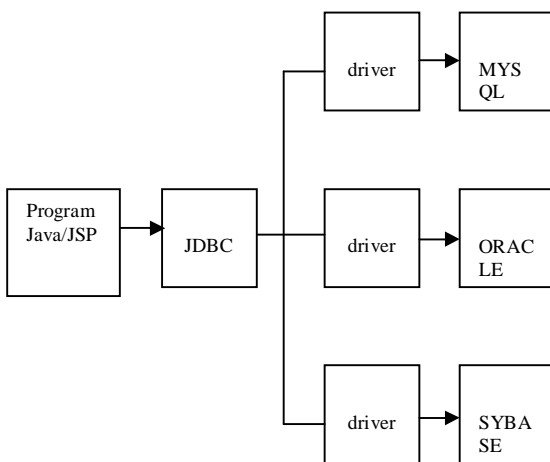
4. Untuk Template

JSP juga sangat tepat untuk menerima dan menyatukan data yang berasal dari berbagai macam sumber.

2.8. Sistem Database

Java Server Pages memiliki kemampuan menangani database dengan menggunakan JDBC (Java Database Connectivity) sebagai perantara antara program JSP dengan database server. JDBC merupakan teknologi Java untuk menyimpan informasi, mendapatkan informasi, maupun melakukan manajemen database dalam JSP^[2].

JDBC adalah API (*Application Programming Interface*) yang dikembangkan oleh Sun Microsystem untuk menyediakan akses data universal dalam bahasa pemrograman Java, yang merupakan bagian dari Java 2 SDK *Standard Edition*. Inti JDBC merupakan paket `java.sql` dan `javax.sql` yang terdapat dalam J2EE. JDBC menyediakan mekanisme untuk koneksi database, sintaks untuk mengirimkan query dan melakukan transaksi serta struktur data yang mewakili hasil query. JDBC tidak melakukan standarisasi sql, tetapi dapat mendukung sintaks sql sesuai data base vendor^[9].



Gambar 3. Diagram JDBC

JDBC merupakan koleksi API yang terdiri atas sekumpulan class dan interface yang ditulis dalam bahasa pemrograman Java sebagai standar API untuk developer maupun software database sehingga

memungkinkan pembuatan aplikasi database yang portabel dengan bahasa pemrograman Java^[2].

Dalam JDBC terdapat tujuh langkah standar untuk menangani database^[2]:

1. Memanggil / load driver JDBC

Driver adalah software yang mampu berkomunikasi dengan database server sehingga apabila menggunakan database server yang berbeda, memerlukan driver yang berbeda pula. Pada saat memanggil driver, sebenarnya yang dilakukan adalah memanggil file program class. Driver-driver yang sesuai dengan database server yang bermacam-macam ini, dapat diperoleh dari situs Sun Microsystem^[9].

2. Mendefinisikan URL untuk koneksi database

Setelah memanggil driver JDBC, perlu untuk mendefinisikan lokasi secara spesifik dari database server. URL yang mengacu ke database server menggunakan protokol `jdbc:` dan diikuti server host, port dan nama database. Data user dan password dapat diikutsertakan pada url secara langsung pada url koneksi.

3. Melakukan koneksi database

Dalam melakukan koneksi database, terlebih dulu harus mengirimkan informasi URL, username dan password untuk database ke metoda `getConnection` dari class `DriverManager`.

4. Membuat objek Statement

Untuk melakukan query, memerlukan objek `Statement` yang dapat dibuat dari objek `Connection`.

5. Melakukan query dan update

Setelah memiliki objek `Statement`, dapat digunakan untuk mengirimkan query dan mengeksekusi dengan metode `executeQuery`, yang mengembalikan objek bertipe `ResultSet`.

6. Memproses hasil

Untuk memproses hasil, digunakan objek `ResultSet` yang akan menampilkan semua baris hasil query pada browser .

7. Menutup koneksi

Sebelum menutup koneksi database, perlu untuk me-release atau melepaskan objek `ResultSet` yang ada, dengan pernyataan `statement.close`.

2.9. MySQL

Merupakan suatu software sistem manajemen database. Untuk membuat, mengedit, mengakses dan memproses data yang disimpan disimpan dikomputer, diperlukan sistem manajemen database seperti MySQL^[2].

Suatu sistem relasional database menyimpan data pada tabel berbeda dan tidak meletakkannya pada satu tabel saja. Hal ini dapat meningkatkan kecepatan dan fleksibilitas. Tabel-tabel tersebut dihubungkan dengan suatu relasi yang telah didefinisikan sehingga dapat mengkombinasikan data dari beberapa tabel pada suatu saat. MySQL menggunakan standar SQL (Structured Query Language), yaitu bahasa standar yang paling banyak digunakan untuk mengakses database.

Alasan penggunaan MySQL sebagai database server karena sangat cepat, tangguh dan mudah digunakan. MySQL didesain untuk menangani database yang besar, konektifitas, kecepatan dan sekuriti yang baik, sehingga sangat cocok digunakan diinternet. Database MySQL memilki kemampuan sebagai berikut^[2]:

1. Dapat digunakan dan dimanfaatkan oleh banyak CPU sekaligus
2. Memiliki konektifitas dengan bahasa C, C++, Java, Perl, PHP, Phytan dan Tcl.
3. Dapat berjalan dalam berbagai sistem operasi
4. Sangat cepat
5. Mendukung banyak fungsi SQL
6. Mendukung ODBC
7. Mampu menangani database yang besar

2.10. Sekilas Tentang E-commerce

Electronic Commerce (E-commerce) adalah suatu cara baru untuk melakukan perdagangan dengan memanfaatkan medium elektronis yang telah memiliki standar yang universal dan tersebar luas seperti halnya Internet. Secara umum dapat dikatakan bahwa konsep perdagangan elektronis bisa merupakan sebuah pemersatu mata rantai distribusi dalam perdagangan. Dalam konsep perdagangan konvensional, mata rantai ini terputus karena keterbatasan sarana dan fasilitas. Siklus terbuka terjadi pada kelompok-kelompok pemberi suplai dan penerima suplai secara terpisah. Dengan menggunakan jaringan aplikasi perdagangan elektronis yang tepat maka bisa didapatkan sebuah mekanisme mata rantai perdagangan yang tidak terputus antara penjual dengan pembeli atau pelanggan. Pendekatan ini akan memberikan nilai tambah kepada semua pihak yang terlibat dalam mata rantai perdagangan tersebut. Hal ini sangat membantu dalam mengatur penjadwalan distribusi dan penyimpanan barang. Efisiensi tersebut pada akhirnya akan menguntungkan bagi pelanggan atau pembeli karena biaya produksi dan distribusi dapat ditekan sehingga akan menurunkan harga jual barang. Kemudahan dalam memilih dan membeli barang adalah juga salah satu nilai tambah yang didapatkan pihak pembeli/pelanggan dalam mekanisme perdagangan elektronis tersebut. Hal ini akan berdampak pada perluasan ataupun penajaman segmen pasar sebagai target penjualan yang akan semakin meningkat. Oleh karenanya kemampuan perusahaan menanggapi *demand* atau permintaan pasar akan semakin menentukan keberhasilan perusahaan tersebut dalam persaingan.

III. ANALISA DAN PERANCANGAN SISTEM

Membahas mengenai analisa kebutuhan sistem perdagangan pada toko buku, sesuai dengan batasan sistem yang dibuat dalam tugas akhir ini. Menganalisa dan membangun tabel-tabel yang diperlukan dan menentukan relasi antar tabel. Merancang komponen logika presentasi dan logika bisnis yang diperlukan untuk aplikasi pemrograman *web*.

3.1. Analisa Kebutuhan Sistem

Pengembangan aplikasi *e-commerce* ini berdasarkan pada analisa kebutuhan dalam melakukan transaksi penjualan *online*.

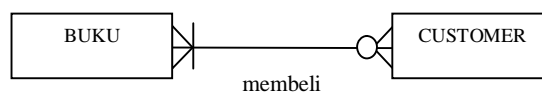
- a. Pada toko buku ini, pelanggan dapat melihat-lihat buku yang ada pada katalog, kemudian melakukan pencarian buku berdasarkan kategori, maupun mencari berdasarkan *keyword* (kata kunci) yang dapat dimasukkan melalui *search box* (kotak mencari) yang ada pada setiap halaman katalog. Pada halaman depan, pemilik buku bisa menampilkan buku-buku yang ingin dipromosikan secara khusus.
- b. Setelah menemukan buku yang tepat, pelanggan dapat memasukkan buku yang diinginkan ke keranjang belanjanya (*shopping cart*). Pelanggan bisa kembali melihat-lihat katalog dan menambahkan buku yang lain ke keranjang belanjanya. Pelanggan juga dapat melakukan *update* (pembaruan) terhadap jumlah buku yang ingin dibeli dan bisa membatalkan pembelian dengan cara mengeluarkan buku yang ada dari keranjang belanjanya.
- c. Pada saat pelanggan memutuskan untuk membeli buku yang ada pada keranjang belanjanya, maka pelanggan harus login terlebih dahulu atau mendaftarkan dirinya supaya bisa melakukan *order* pembelian.
- d. Setelah pelanggan melakukan *order*, *order* akan dimasukkan ke database, keranjang belanja pelanggan akan dikosongkan dan halaman terima kasih akan ditampilkan beserta nomor order pelanggan yang bersangkutan.

3.2. Tahapan Perancangan

Dalam Tugas Akhir ini akan dilakukan perancangan dengan menggunakan perangkat pemodelan ERD (*Entity Relationship Diagram*) dan DFD (*Data Flow Diagram*). ERD digunakan untuk menentukan hubungan antar entitas. Entitas merupakan suatu objek atau kejadian. DFD digunakan untuk menganalisa aliran data dan proses dalam sistem.

3.2.1 ERD (Entity Relationship Diagram)

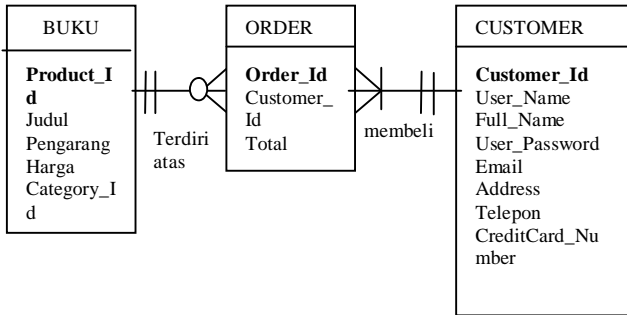
Langkah pertama yang dilakukan adalah menentukan entitas, kemudian terlebih dahulu dibuat model data konteks atau diagram struktur data dari entitas sistem yang telah ditentukan. Pada Tugas Akhir ini ditunjukkan oleh Gambar 4 berikut.



Gambar 4. Model data konteks sistem *e-commerce* Tokobuku

Selanjutnya dibuat model data yang menggambarkan entitas lengkap dengan seluruh atributnya yang akan menjadi acuan dalam

membangun tabel-tabel dalam basis data. Entitas akan diterjemahkan menjadi nama tabel dan atribut dari suatu entitas akan menjadi nama-nama kolom dari tabel tersebut. Sehingga dari model data terakhir sudah dapat ditemukan suatu tabel yang dilengkapi dengan atribut yang menjadi kunci sebagai pengenal unik suatu tabel tersebut. Model data tersebut ditunjukkan oleh Gambar 5 berikut.

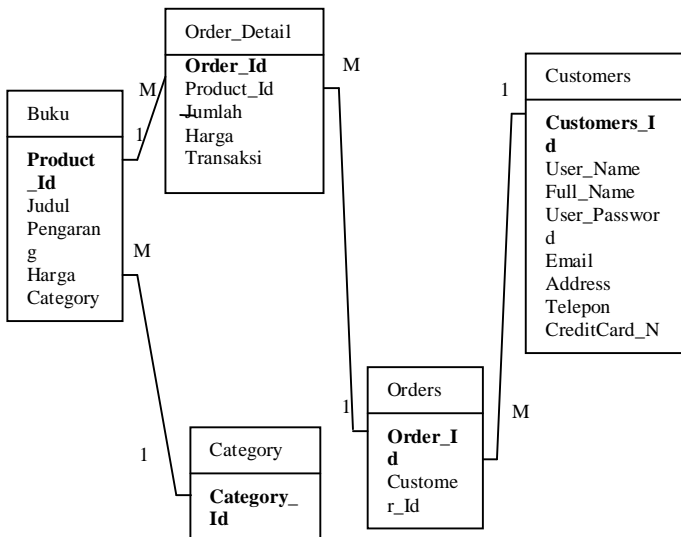


Gambar 5 Model data lengkap dengan atribut pada sistem e-commerce

3.2 Normalisasi Tabel

Untuk mencegah pembuatan tabel yang tidak fleksibel dan untuk mengurangi ketidakefisienan, perlu dilakukan normalisasi terhadap tabel atau atribut dari entitas-entitas yang sudah terbentuk. Aturan-aturan normalisasi dinyatakan dalam istilah bentuk normal.

Dari proses normalisasi basisdata diperoleh skema basis data untuk aplikasi penjualan buku *online* ini terdiri atas tujuh buah tabel yang saling berkaitan. Hubungan antar tabel-tabel dinyatakan dalam diagram E-R (*Entity-Relationship*). *Entity* merupakan suatu objek atau kejadian. *Relationship* merupakan hubungan atau kaitan antar entitas. Gambar 3.6 menunjukkan diagram E-R yang dinyatakan dalam notasi 1:1 yang berarti hubungan satu ke satu dan 1:M yang berarti hubungan satu ke banyak :



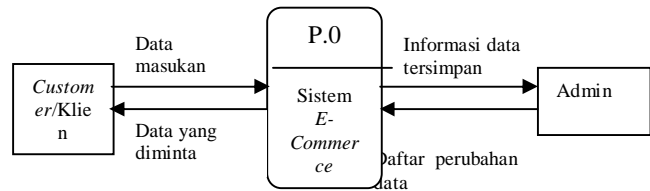
Gambar 6. Relasi antar tabel pada e-commerce

3.2.1 Desain Fisik

Pada proses desain fisik ini, merupakan tahap untuk menuangkan perancangan basis data yang telah dilakukan ke dalam basis data fisik yang tersimpan pada media penyimpan, yang pada tugas akhir ini menggunakan MySQL dengan nama basis data Jual Buku. Tabel – tabel yang terdapat dalam basis data JualBuku ada tujuh yaitu Tabel Buku, Tabel *Order*, Tabel *Order_Detail*, Tabel *Category*, dan Tabel *Customer*. Tabel-tabel ini merupakan hasil dari proses normalisasi yang telah dilakukan dan dari kebutuhan fasilitas tambahan, seperti Tabel *Promotion* yang digunakan untuk menampilkan buku-buku baru yang sedang dipromosikan.

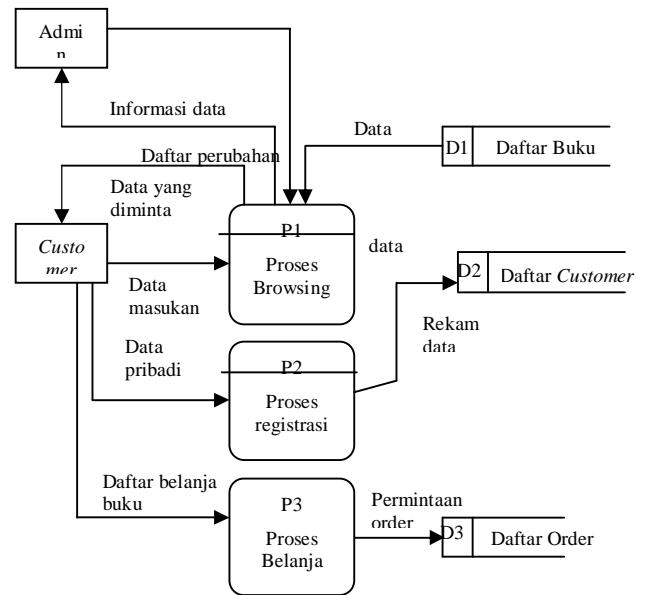
DFD (Data Flow Diagram) Sistem

Untuk menentukan ruang lingkup sistem yang akan dibangun, sesuai dengan kebutuhan sistem yang telah dianalisa sebelumnya, dijabarkan dalam bentuk diagram konteks seperti pada Gambar 3.7.



Gambar 3.7. Diagram konteks e-commerce toko buku

Selanjutnya dari diagram konteks, dijabarkan menjadi *Data Flow Diagram (DFD)* untuk mendapatkan gambaran sistem yang lebih detail. Berdasarkan analisa kebutuhan sistem yang dilakukan oleh penulis seperti yang telah dijelaskan sebelumnya, sesuai dengan batasan sistem yang diberikan.



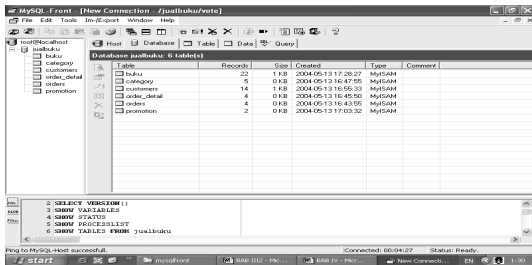
Gambar 3.8. DFD e-commerce toko buku

BAB IV IMPLEMENTASI DAN PENGUJIAN PROGRAM

Pada tahapan ini dilakukan pembuatan tabel-tabel pada database server yang digunakan yaitu MySQL sesuai dengan hasil analisa dan proses desain database yang telah dilakukan. Dan juga dilakukan pengujian terhadap program yang dibuat.

4.1. Implementasi Database

Database pada sistem *e-commerce* toko buku ini, dibuat dengan database server MySQL. Dalam proses pembuatan database dan tabel-tabel digunakan program pendukung yaitu MySQL-Front. Database sistem *e-commerce* disimpan dengan nama jualbuku.



Gambar 4.1 Tampilan struktur database jualbuku pada MySQL-Front

4.2. Pengujian Program

Setelah penulisan kode program selesai, kemudian program dipasang pada Apache Tomcat 4.0 dengan diletakkan pada direktori webapps. Keseluruhan program disatukan dalam folder dengan nama Tokobuku. Program-program java yang terangkum dalam logika bisnis dipackage dalam paket myEJB. Untuk menyimpan paket ini diperlukan subfolder baru pada folder Tokobuku yaitu WEB-INF. Sedangkan file-file untuk logika presentasi diletakkan langsung dalam folder Tokobuku dengan ekstensi .jsp untuk file-file JSP dan untuk file-file HTML disimpan dengan ekstensi .html.

4.2.1. Halaman Index Bagian User

Halaman index merupakan halaman utama dari *e-commerce* toko buku ini. Halaman utama berfungsi untuk menampilkan buku yang dipromosikan. Di halaman ini, pelanggan dapat memilih buku yang di sukai menggunakan katalog yang berada di samping kiri ataupun dengan menulis buku tersebut pada *Search Books*.



Gambar 4.2. Halaman index

Dari halaman ini pelanggan yang sudah terdaftar bisa melakukan login pada form yang tersedia dengan memasukkan *user_name* dan *password* sesuai yang dimilikinya. Bila data yang dimasukkan benar maka akan ditampilkan halaman selamat datang yang menyatakan bahwa pelanggan tersebut telah berhasil melakukan login.



Gambar 4.3 Halaman Login berhasil

Apabila salah satu atau kedua data yang dimasukkan salah / tidak sesuai dengan data yang ada dalam tabel database, maka akan ditampilkan halaman pesan *error* yang menerangkan kesalahan data yang dimasukkan. Seperti yang terlihat pada Gambar 4.4 dan Gambar 4.5 berikut:



Gambar 4.4 Halaman pesan kesalahan *user_name*



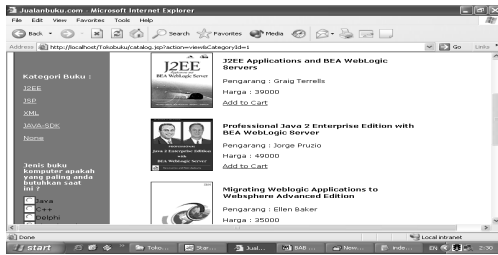
Gambar 4.5 Halaman pesan kesalahan password

Dengan menekan tombol OK maka akan kembali ke halaman utama, dan dapat memasukkan kembali data yang benar.

4.2.2. Halaman Katalog

Halaman katalog berfungsi untuk menampilkan katalog yang berisi buku-buku tertentu sesuai kategori maupun berdasarkan hasil pencarian. Pada situs *e-commerce* ini terdapat empat kategori buku yaitu J2EE, JSP, XML, JAVA-SDK dan None.

Misalnya seorang pelanggan ingin melihat daftar buku-buku yang terdapat dalm kategori J2EE, maka dia tinggal mengarahkan *pointer mouse* pada tulisan J2EE yang ada disisi kiri halaman dibawah tulisan kategori buku. Kemudian akan ditampilkan buku-buku dalam database yang termasuk kategori J2EE seperti ditunjukkan oleh Gambar 4.6.



Gambar 4.6. Halaman katalog

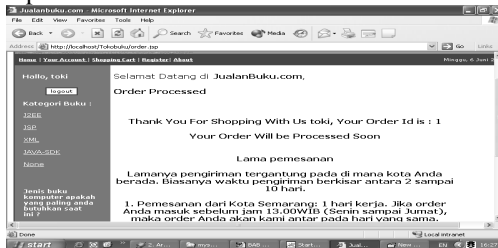
4.2.3. Halaman Shopcart

Halaman shopcart ini berfungsi menampilkan daftar belanja pelanggan. Pada halaman ini pelanggan juga bisa menghapus dan menambah jumlah buku yang ingin dibeli. Pada halaman ini ditampilkan judul buku, jumlah, sub total, delete dan *update* dan Total. *Update* akan digunakan apabila pelanggan ingin menambah atau mengurangi jumlah buku yang akan dibeli. Untuk melakukan penambahan atau pengurangan jumlah buku, pelanggan dapat mengganti nilai pada kolom jumlah dengan nilai yang diinginkan kemudian menekan tombol *update*. Kolom sub total akan mengalikan jumlah buku dengan harga tiap bukunya, kemudian akan dijumlahkan dengan sub total buku yang lain menjadi Total. Pada halaman shopcart ini, pelanggan diwajibkan untuk login terlebih dahulu apabila ingin melalui proses *order*.



Gambar 4.7. Halaman shopcart

Untuk memasukkan daftar buku yang dibeli, agar segera diproses maka pelanggan harus melakukan order dengan menekan tombol order. Setelah itu keranjang belanja akan dikosongkan lagi dan daftar belanja telah tersimpan di database admin untuk diproses.



Gambar 4.8 Halaman proses order

Setelah tombol order ditekan maka akan ditampilkan Gambar 4.8 yang berisi keterangan tentang proses pengiriman buku yang dipesan.



Gambar 4.9 Isi dari Tabel Order

Daftar order yang dimasukkan oleh pelanggan akan tersimpan dalam Tabel Order pada database yang hanya bisa dilihat oleh bagian Admin untuk proses pengiriman. Untuk memperjelas daftar order, Admin bisa menekan tombol detail dan akan ditampilkan halaman order detail seperti pada Gambar 4.10.



Gambar 4.10 Isi dari Tabel order_detail

Dari sini, admin dapat melihat nomor pembelian, buku- buku apa saja yang dibeli oleh pelanggan dan jumlah buku yang dibeli.

4.2.4. Halaman Register

Komponen berikut ini merupakan komponen yang berkaitan dengan data pelanggan. Pada suatu e-commerce pada umumnya, data pelanggan disimpan oleh toko yang berkaitan pada database sehingga saat melakukan pembelian seorang pelanggan tidak perlu memasukkan data dirinya secara berulang-ulang. Oleh karena itu terdapat komponen registrasi yang berfungsi untuk menangani pendaftaran pelanggan.

Pada komponen ini juga ditambahkan fasilitas untuk login dan logout.



Gambar 4.11. Halaman register

Setelah semua data diisikan ke form yang tersedia, kemudian tombol register ditekan untuk memasukkan data-data tersebut ke dalam database. Untuk melihat hasilnya dapat dilakukan oleh bagian Admin dengan melihat daftar customer yang tersimpan, ataupun dapat dilihat langsung pada database MySQL. Hasilnya ditunjukkan oleh Gambar 4.12 yang menampilkan data-data pelanggan yang telah terdaftar dalam database jualbuku.

8	Yuda	Yuda	Yuda	Uda@Bandung.com	Bandung	0221234	987654321
9	Paux	paux	paux	paux@Baskoro.com	Tembalang	1234567	123456789
10	Jjo	Jjo	Jjo	Jjo@Baskoro.com	Jakarta	1234567	123456789
11	Inda	Indapupo	popo	Inda@ngaran.com	ngaran	5467654	123456787
12	Susi	Susilo	Susilo	Susi@Kudus.com	Kudus	0292123	567456543
13	Deco	Chery	Chery	Deco@garasi.com	Jl Baskoro	456456	654654654
14	adi	Adi Utomo	adi	adi@jakarta.com	Jakarta	1231231	123123123

Gambar 4.12 Isi dari Tabel customer dalam database jualbuku

Namun bila data yang diisikan kedalam form registrasi tidak terisi semua maka proses registrasi akan gagal dan ditampilkan pesan error, seperti terlihat pada Gambar 4.10. Menyatakan pesan kesalahan karena ada data yang kosong misal pada full_name, email dan nomor telepon.

Gambar 4.13 Halaman registrasi gagal

V. PENUTUP

Dari seluruh uraian yang dijelaskan pada bab-bab sebelumnya dan dari hasil analisis dan pengujian, dapat diambil kesimpulan dan saran-saran untuk pengembangan dan perbaikan lebih lanjut.

5.1. Kesimpulan

1. Dari proses desain basisdata dihasilkan enam buah tabel yang telah mengalami proses normalisasi sampai bentuk normal ke dua.
2. Sistem *e-commerce* ini berhubungan dengan database server dan dapat melakukan penambahan, perubahan/penggantian dan penghapusan terhadap isi data dari tabel-tabel dalam database dan menampilkannya pada *browser*.
3. Server Tomcat tidak dapat mendeteksi secara otomatis perubahan yang terjadi pada file yang di-include-kan secara langsung (*direktif include*). Agar perubahan dapat dideteksi otomatis dan mampu memberikan parameter pada file yang di-include-kan, maka digunakan *action include*.
4. Proses pemanggilan/eksekusi file-file “.jsp” memiliki waktu lebih lama dibandingkan file-file “.html”, dikarenakan pada file-file “.jsp” mengalami proses kompilasi menjadi servlet.

5.2. Saran

Sistem yang dibangun pada Tugas Akhir ini sebatas pada proses jual beli, untuk itu dapat dikembangkan dengan membangun sistem pada toko secara lebih mendalam seperti laporan administrasi dan keuangan, hubungan dengan pihak penerbit dan mengenai segi keamanan, serta pembuatan halaman/tampilan yang lebih dinamis disertai animasi yang menarik.

VI. DAFTAR PUSTAKA

1. Agus Setyabudi, *Aplikasi E-commerce dengan Java Servlet dan JSP*, Elex Media Komputindo, 2002
2. Didik Dwi Prasetyo, *Administrasi Database Server MySQL*, Elex Media Komputindo, 2003
3. Isak Rickyanto, *Belajar Sendiri Java Server Pages*, Elex Media Komputindo, 2002
4. Marty Hall, *Core Servlets and Java Server Pages*, Prentice Hall, 2000
5. Patrick Naughton, *The Java Handbook*, Mc-Grow Hill, Inc, 1996
6. Ridwan Sanjaya, Onno W Purbo, *Membangun Web dengan JSP*, Elex Media Komputindo, 2002
7., java.sun.com/docs/books/tutorial
8., java.sun.com/products/jsp/

Tentang Penulis:

Suhartoko, seorang mahasiswa Teknik Komputer Informatika Fakultas Teknik UNDIP Semarang tingkat akhir, sedang dalam proses menyelesaikan studi. Dapat dihubungi di email dik_toko@yahoo.com

Menyetujui:

Pembimbing,

Agung Budi P. ST. MIT
NIP. 132 137 932