

**IMPLEMENTASI COMPONENT-BASED SOFTWARE ENGINEERING
PADA APLIKASI E-COMMERCE
(STUDI KASUS MUSTIKA JAYA ELEKTRONIK)**

Muhammad Iqbal Rizqon Bahaudin¹, Helmie Arif Wibawa, S.Si., M.Cs², Satriyo Adhy, S.Si., MT³

Jurusan Ilmu Komputer/Informatika FSM Universitas Diponegoro
Email: iqbal.rizqon@gmail.com¹, helmie.arif@gmail.com², satriyoadhy@gmail.com³

ABSTRAK

Perusahaan-perusahaan bisnis saling bersaing untuk mempromosikan produk-produknya dengan media *internet* pada era globalisasi saat ini. *E-Commerce* merupakan cara baru dalam melakukan bisnis dengan elektronik melalui media *internet*. Mustika Jaya Elektronik (MJ Elektronik) merupakan salah satu perusahaan bisnis swasta di bidang elektronik yang sedang meningkatkan promosinya dalam dunia maya, sehingga perusahaan ini membutuhkan aplikasi *E-Commerce* untuk mempromosikan produk-produknya dan dapat tetap bersaing dengan perusahaan bisnis serupa. *Component-based Software Engineering (CBSE)* merupakan model proses pengembangan perangkat lunak dengan menggunakan komponen perangkat lunak yang tersedia. Model proses ini memungkinkan suatu sistem dapat dilakukan generalisasi sehingga pada saat pengembangan *software* yang sejenis hanya membutuhkan sedikit modifikasi. Penggunaan model proses *CBSE* mempermudah dalam pengembangan Aplikasi *E-commerce* Toko MJ Elektronik dengan adanya komponen-komponen yang sudah tersedia dan setiap komponen mempunyai sifat *reusable* yang tinggi, sehingga dapat langsung dipakai dan mudah dalam penggunaannya. Hasil akhir dari pengembangan perangkat lunak adalah dihasilkan sebuah perangkat lunak *E-commerce* Toko MJ Elektronik berbasis komponen (*URL*: www.mjelektronik.com).

Kata Kunci : *Component-based Software Engineering, Internet, E-Commerce, Mustika Jaya Elektronik*

ABSTRACT

Business Companies competes promoting their products using internet in this global era. *E-Commerce* is a new way to electronically run business through the internet. Mustika Jaya Elektronik (MJ Elektronik) is an electronic private company is increasing its promotion in cyberspace, so the company is in need of *E-commerce* to promote its products and to remain competitive with similar company. *Component-based Software Engineering (CBSE)* is a model of software engineering which using existing software component. This model brings through the capability of generalization in a system, so in the developing phase similar software only needed a bit modification. Use *CBSE* can be simplify in developing MJ Elektronik Store *E-Commerce* application with available components that have has a high reusable properties, so components can be simplify used. The final outcome of software development is generated an component-based *E-commerce* software Mustika Jaya Elektronik Store (*URL*: www.mjelektronik.com).

Keywords : *Component-Based Software Engineering, Internet, E-Commerce, Mustika Jaya Elektronik*

1. PENDAHULUAN

Electronic Commerce (E-Commerce) merupakan cara baru dalam melakukan transaksi bisnis dengan elektronik melalui media *internet* antar beberapa organisasi atau

perorangan. *E-Commerce* akan mengubah semua kegiatan dalam pemasaran dan transaksi agar dapat dilakukan dengan lebih efektif. Sekarang ini *E-Commerce* sudah banyak digunakan di dalam perdagangan di Indonesia [6].

E-Commerce yang digunakan oleh beberapa perusahaan bisnis pada umumnya memiliki fitur yang hampir semua sama. Hal ini dapat dilihat dari aplikasi *E-Commerce* sekarang ini seperti tokobagus.com, berniaga.com dan lainnya. Pengembang membutuhkan referensi arsitektur untuk membangun sebuah aplikasi *E-Commerce* dan memerlukan metode untuk menganalisis sifat dari sistem untuk meningkatkan desain sistem *E-Commerce*.

Toko Mustika Jaya Elektronik (MJ Elektronik) merupakan salah satu perusahaan bisnis swasta di bidang elektronik yang ada di Jepara. Sebagai perusahaan yang bergerak maju, perusahaan ini terus berusaha memperluas gerakannya. MJ Elektronik sedang meningkatkan promosinya dengan media *Internet*, sehingga perusahaan ini membutuhkan aplikasi *E-Commerce* untuk mempromosikan produk-produknya dan dapat tetap bersaing dengan perusahaan bisnis yang serupa pada era globalisasi ini.

Menurut Pressman, *Component-based Software Engineering (CBSE)* adalah model proses pengembangan *software* dengan menggunakan komponen *software* yang sudah ada. Model ini memungkinkan suatu sistem dapat dilakukan generalisasi sehingga pada saat pengembangan *software* yang sejenis hanya membutuhkan sedikit modifikasi sistem pada *user*. Sebagai contoh "*System Analysis and Programming*" (SAP) merupakan bentuk perangkat lunak *Component-based* pada dunia *Enterprise* [10].

Menurut Ivica, Stig dan Michel, pengembangan perangkat lunak dengan model proses *CBSE* pada beberapa akhir tahun ini sangat menjanjikan. Pengembangan perangkat lunak berbasis komponen yang berhubungan dengan *lifecycle* dari produk berbasis komponen berfokus pada teknologi yang berkaitan dengan perancangan dan implementasi komponen perangkat lunak dan sistem yang dibangun dari komponen yang tersedia [3].

Suatu perangkat lunak berbasis komponen dapat mengambil komponennya dari berbagai macam sumber independen, internal maupun eksternal. Komponen bersifat *plug & play*, sehingga penggabungan dapat dilakukan dengan mudah dan dapat dengan mudah berinteraksi dengan bagian lain dari keseluruhan perangkat lunak [8].

Masalah diatas diangkat menjadi topik dalam tugas akhir ini karena dengan mengimplementasikan model proses *Component-based Software Engineering* dan memanfaatkan komponen-komponen yang tersedia dapat mempermudah dalam pengembangan. Selain itu, dengan implementasi *Component-based Software Engineering* dihasilkan sebuah aplikasi *E-Commerce* yang dapat dijadikan media penjualan dan media mempromosikan produk pada Toko MJ Elektronik.

2. TINJAUAN PUSTAKA

2.1. *E-Commerce*

Pesatnya perkembangan teknologi *internet* sekarang ini semakin terasa dengan munculnya berbagai jenis informasi di *internet*, salah satu sektor yang berkembang besar di *internet* adalah *E-Commerce*. Menurut Laudon dan Guercio yang dijelaskan pada buku berjudul "*E-Commerce bussiness, technology, sosity*", *E-Commerce* merupakan suatu transaksi bersifat komersial antara perorangan atau organisasi secara digital. Transaksi yang dimaksud secara digital yaitu semua transaksi melalui media teknologi digital seperti *internet* dan *web*. Melalui media tersebut transaksi dapat dilakukan dari tempat dan waktu yang berbeda. Kegiatan yang dilakukan dalam transaksi juga menjadi lebih mudah dan tidak memerlukan usaha yang banyak [4].

E-Commerce dapat meningkatkan efisiensi, menekan biaya serta memberikan akses yang lebih luas kepada partner bisnis dan pelanggan perusahaan. *E-Commerce* juga sebagai komunikasi elektronik dan informasi digital yang memroses teknologi dalam transaksi bisnis untuk menciptakan, mengubah, dan mempererat hubungan antar organisasi maupun organisasi dengan perorangan [4].

2.2. *Component-based Software Engineering*

Menurut Pressman, *Component-based Software Engineering (CBSE)* adalah proses yang menekankan perancangan dan pembangunan *software* dengan menggunakan komponen *software* yang sudah ada. Seperti

yang dijelaskan oleh Szyperski C. (1998): “A software component is a unit of composition with contractually specified interface and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third parts”. Komponen perangkat lunak mempunyai komposisi dengan *interface* sebagai sebuah kontrak untuk komponen, yaitu mengenai syarat input yang diperlukan, seperti atribut dan format data, untuk menjaga integritas sistem di dalam sebuah komponen dan juga output yang dihasilkan. *Interface* dari komponen hanya membutuhkan data atribut yang sederhana [10].

Menurut Clemens Szyperski, pengembangan perangkat lunak berbasis komponen merupakan sebuah model pengembangan yang didasari oleh model *Object-Oriented* yang mengalami pergeseran paradigma dalam dunia pengembangan perangkat lunak. Pergeseran tersebut adalah perubahan dari sebuah sistem yang dibangun secara monolitik, *single-platform*, dan sistem yang selalu dibuat dari awal (*built-from-scratch*) menjadi sebuah komponen yang siap pakai, *platform-independent* yang memungkinkan adanya komunikasi antar *object*, dan tidak perlu selalu dibuat dari awal. Secara singkat, pengembangan perangkat lunak berbasis komponen ini sangat memperhatikan aspek *reusability* dari sebuah komponen. Tingkat *reusability* yang tinggi menjadi syarat mutlak untuk membuat sebuah komponen [13].

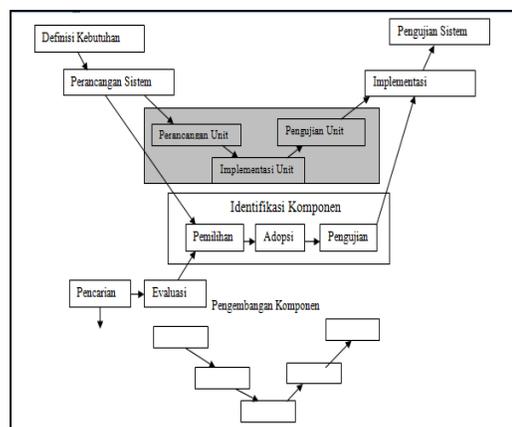
Komponen adalah sebuah bagian dari sistem yang dapat tergantikan, selama masih merealisasikan *interface-interface* yang sama. *Interface* pada komponen hanya menjelaskan layanan yang disediakan oleh komponen tersebut, bukan implementasinya. Implementasi fisik sebuah komponen tersembunyi bagi para penggunanya, karena komponen hanya diakses melalui *interface*-nya, karenanya tingkah laku yang dipresentasikan oleh *interface-interface* sebuah komponen harus didefinisikan dengan jelas [13].

Tujuan dari pengembangan perangkat lunak berbasis komponen ini adalah untuk mengembangkan sebuah sistem yang besar, dengan menggabungkan komponen-komponen yang telah dikembangkan sebelumnya, sehingga dapat mengurangi

waktu dan biaya dalam pengembangan suatu perangkat lunak.

Menurut Sommerville, Salah satu faktor pendukung pendekatan *CBSE* ini adalah telah banyaknya teknologi yang mampu memfasilitasi pemakaian komponen dalam pengembangan perangkat lunak. Visual Basic telah dikenal komponen seperti *VBControl* dan *ActiveX*, dan java juga menyediakan *JavaBeans* dan *EJB* untuk versi enterprise atau sekala besarnya. Selain itu teknologi objek terdistribusi seperti diperkenalkan oleh *CORBA* juga telah memungkinkan *remote method invocation* [12].

Menurut Ivica, Stig, dan Michael dalam jurnal yang berjudul “*Component-based Development Process and Component Life Cycle*”, alur pengembangan model proses *Component-based software engineering* dapat dilihat pada gambar 2.1 di bawah ini beserta penjelasannya.



Gambar 2.1. *Component-based development life cycle* [3].

Penjelasan tahapan model *CBSE* pada Gambar 2.1 sebagai berikut:

1) Definisi Kebutuhan sistem

Model proses *CBSE* ini dimulai dari tahap definisi dan spesifikasi kebutuhan sistem seperti model proses lainnya. Pada model pengembangan perangkat lunak *noncomponent-based* proses akan diteruskan pada tahap perancangan unit, implementasi

unit dan pengujian unit yang dijelaskan pada kotak dengan warna arsiran pada Gambar 2.1.

Pada tahap ini salah satu kegiatan yang penting adalah untuk menganalisis kemungkinan mewujudkan solusi yang memenuhi kebutuhan sistem. Pendekatan ini menyiratkan bahwa perlu untuk menganalisis apakah persyaratan ini dapat dipenuhi oleh komponen yang tersedia. Karena tidak mungkin bahwa komponen yang sesuai dapat selalu ditemukan.

2) Perancangan Sistem

Seperti dengan fase definisi kebutuhan sistem, perancangan sistem sangat terkait dengan ketersediaan komponen. Model komponen tertentu membutuhkan arsitektur *framework* tertentu dan aplikasi sistem tertentu harus menggunakan *framework* tersebut juga. Untuk alasan ini proses perancangan erat terhubung pada ketersediaan komponen.

Pada tahap pencarian komponen harus benar-benar dilihat dari perancangan sistem itu sendiri, karena dalam suatu *repository* terdapat banyak komponen yang tidak perlu untuk dipilih. Maka dalam pencarian komponen harus dilakukan proses evaluasi supaya komponen yang dipilih sesuai dengan kebutuhan sistem. Untuk mengetahui apakah sebuah komponen sesuai dengan definisi kebutuhan dan perancangan sistem dapat dilihat dari *interface* masing-masing komponen. Suatu komponen yang tersedia dalam *repository* biasanya sudah teruji dan dapat diadopsi sebelum dapat diintegrasikan dengan sistem yang lain.

3) Implementasi dan Pengujian Sistem

Pada implementasi sistem meliputi proses integrasi antara infrastruktur komponen standar dalam suatu *framework* dan aplikasi berbasis komponen itu sendiri.

Untuk dapat mengintegrasikan antara komponen dari standar *framework* dengan aplikasi berbasis komponen yaitu dengan cara

melakukan konfigurasi sesuai dengan peraturan yang ada pada *framework* tersebut.

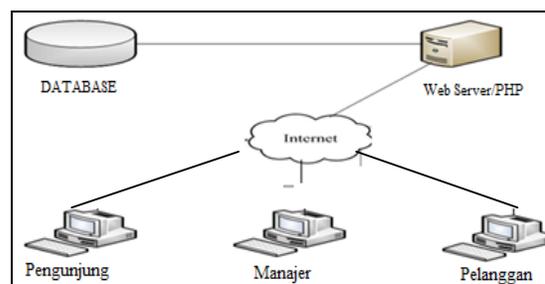
Tahap pengujian aplikasi *E-Commerce* Toko Mustika Jaya dilakukan dengan metode *black box* untuk menguji fungsionalitas setiap *service* yang disediakan oleh setiap komponen. Sebuah komponen yang diuji menggunakan metode *black-box* dikatakan berhasil jika *service-service* yang ada telah memenuhi spesifikasi *requirements* yang telah dibuat sebelumnya, sehingga komponen dapat dikatakan teruji dengan baik apabila sudah dapat berjalan pada sistem yang baru.

3. PEMBAHASAN

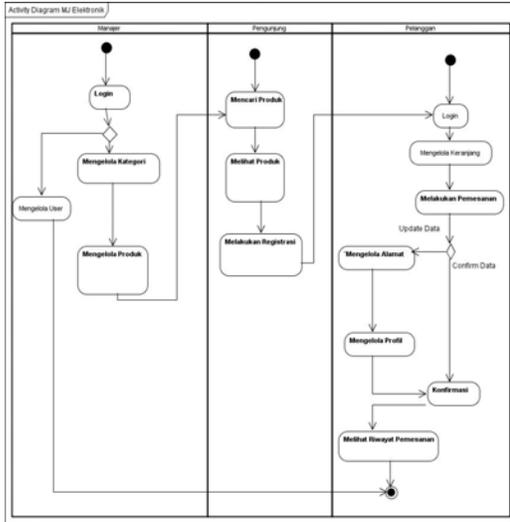
3.1. Analisis Sistem

Sistem yang dibangun adalah sistem *E-Commerce* Toko Mustika Jaya Elektronik berbasis komponen. Aplikasi ini digunakan untuk mendapatkan informasi mengenai produk-produk yang dijual di Mustika Jaya Elektronik. Selain itu aplikasi ini digunakan untuk mengelola katalog, mengelola *user* dan mengelola tarif pengiriman.

Arsitektur perangkat lunak *E-Commerce* yang menjelaskan interaksi antara *user* pengunjung, manajer dan pelanggan yang terhubung melalui media *internet* dapat dilihat pada Gambar 3.1. Diagram aktifitas yang memberikan penjelasan mengenai aktifitas utama yang terjadi antar *user* dapat dilihat pada Gambar 3.2.



Gambar 3.1. Arsitektur *E-commerce* Toko Mustika Jaya Elektronik



Penjelasan 3 jenis *user* pada Arsitektur *E-commerce* dan *Activity Diagram* yang menggambarkan kerja perangkat lunak secara umum ditunjukkan pada Gambar 3.1 dan Gambar 3.2 dideskripsikan sebagai berikut :

1. Manajer

Manajer merupakan *user* yang memiliki hak akses untuk mengelola data produk yang dijual, mengelola kategori produk, mengelola provinsi, mengelola kota, mengelola tarif pengiriman, mengelola pemesanan dan mengelola *user*.

2. Pengunjung

Pengunjung merupakan *user* yang memiliki hak akses melihat informasi produk dan melakukan registrasi untuk menjadi pelanggan.

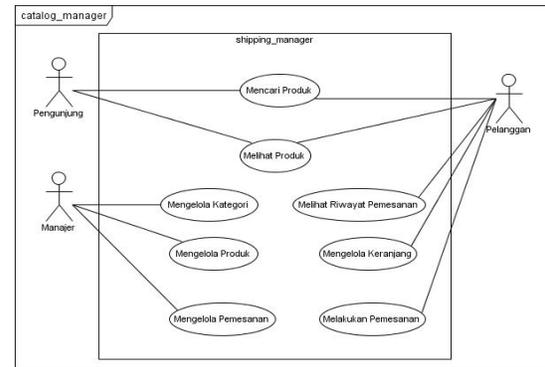
3. Pelanggan

Pelanggan merupakan *user* yang memiliki hak akses mendapatkan informasi produk, memesan produk, mengelola keranjang, melakukan *checkout*, melihat riwayat pemesanan, status pemesanan, mengelola profil dan mengelola alamat pengiriman pelanggan

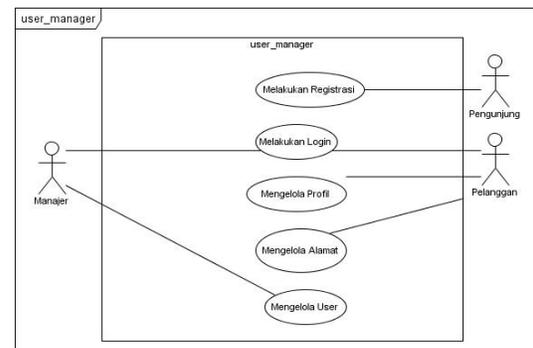
Model *Use Case* digunakan untuk mendeskripsikan fungsi sebuah sistem yang terdiri dari definisi *Actor* dan definisi *Use Case* yang digambarkan ke dalam suatu *Use Case Diagram*. Fungsi dari *Use Case Diagram* digunakan untuk menunjukkan hubungan antara *Actor* dan *Use Case*.

Use Case Diagram disusun berdasarkan daftar *Actor* dan daftar *Use Case* sistem.

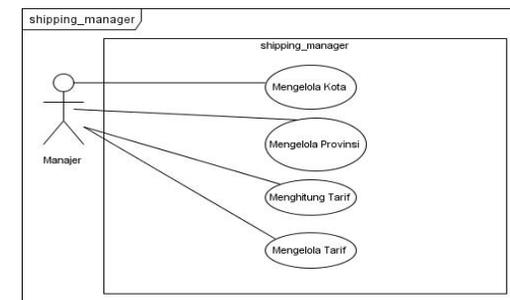
Penyusunan *Use Case Diagram* dilakukan berdasarkan hubungan keduanya. *Use Case Diagram* aplikasi *E-Commerce* dibagi menjadi 3 diagram yaitu *module catalog_manager*, *module user_manager*, dan *module shipping_manager*. *Use Case Diagram* setiap *module* dapat di lihat pada gambar 3.3, gambar 3.4 dan gambar 3.5.



Gambar 3.2. *Use Case Diagram* module *catalog_manager* (U-C)



Gambar 3.3. *Use Case Diagram* module *user_manager* (U-U)



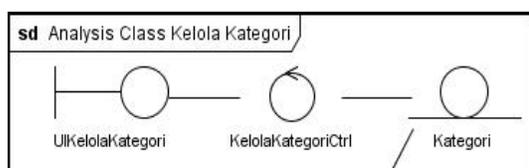
Gambar 3.4. *Use Case Diagram* module *shipping_manager* (U-S)

Use Case Realization menunjukkan interaksi antara kelas analisis dengan fungsional perangkat lunak. Setiap *Use Case* direalisasikan dengan menggunakan *Analysis Class Diagram* dan *Communication Diagram*.

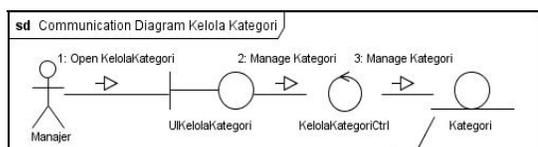
Berikut realisasi analisis untuk masing-masing *Use Case*:

Realisasi *Use Case* “Mengelola Kategori” ditunjukkan dengan *Analysis Class Diagram* pada Gambar 3.23 dan *Communication Diagram* pada Gambar 3.24. Kelas-kelas yang terkait dengan *Use Case* “Mengelola Kategori” antara lain:

- a. *Class boundary* : UIkelolaKategori
- b. *Class control* : KelolaKategoriCtrl
- c. *Class entity* : Kategori



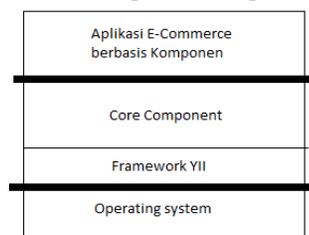
Gambar 3.5. *Analysis Class Diagram* Mengelola Kategori



Gambar 3.6. *Communication Diagram* Mengelola Kategori

3.2. Perancangan Sistem

Sesuai dengan pendekatan model proses *CBSE*, pengembangan aplikasi *E-Commerce* berbasis komponen berada pada *tier* pertama. Arsitektur sistem dapat dilihat pada gambar.



Gambar 3.7. Arsitektur Aplikasi *E-Commerce* berbasis komponen

Tier pertama adalah aplikasi *E-Commerce* Mustika Jaya Elektronik berbasis komponen, yaitu aplikasi utama yang dibangun dengan memanfaatkan komponen yang tersedia. *Tier* kedua adalah komponen-komponen yang sudah dievaluasi dan dipilih berdasarkan definisi kebutuhan dan perancangan sistem.

Tier ketiga adalah *Framework Yii* yang menyediakan komponen-komponen dan menangani aturan dalam interaksi antar komponen. Sedangkan *tier* terakhir adalah sistem operasi yang digunakan.

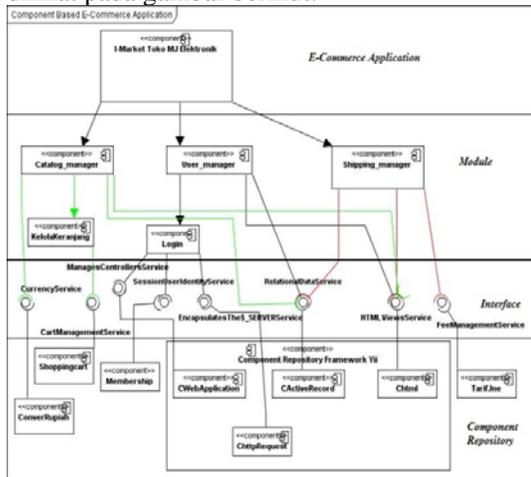
Perancangan sistem aplikasi *E-Commerce* yang berada pada *tier* pertama ini meliputi perancangan basis data, *Use Case realization* yang menghasilkan *Class Diagram*, Identifikasi daftar komponen yang sesuai dengan kebutuhan dan spesifikasi sistem dan *Design Class*.

No	Entity class	Nama tabel
1	Kategori	emk_kategori (10)
2	Produk	emk_produk
3	RiwayatPemesanan	emk_riwayat_pesana
4	Pemesanan	emk_pesanan
5	StatusPemesanan	emk_status_pesanan
6	Keranjang	emk_keranjang
7	User	emk_users
8	Pelanggan	emk_pelanggan
9	Provinsi	emk_provinsi
10	Kota	emk_kota
11	Tarif	emk_tarif

Proses indentifikasi komponen yang pertama adalah memilih komponen yang sudah dievaluasi berdasarkan analisis dan perancangan sistem. Evaluasi komponen meliputi proses pemilihan komponen yang memungkinkan untuk dipilih dengan melihat *interface* masing-masing komponen yang sesuai dengan kebutuhan sistem. aplikasi *E-Commerce* MJ Elekonik telah dibagi menjadi tiga *module* berdasarkan fiturnya, yaitu *module catalog_manager*, *module*

user_manager dan *shipping_manager*. Setiap module merupakan kumpulan *Use Case* yang menggambarkan kebutuhan setiap modulnya.

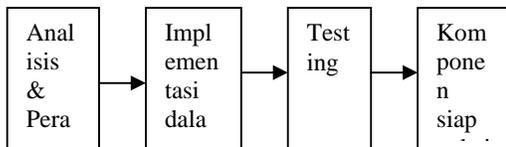
Proses pemilihan komponen dapat dilakukan dengan cara melihat kebutuhan sistem dari setiap *Use Case* yang ada pada setiap *Module*. Proses selanjutnya yaitu dengan mencocokkan kebutuhan sistem dengan *service-service* yang disediakan setiap komponen melalui *interface* yang ada. Komponen dan *interface* masing-masing komponen yang terpilih melalui proses pencarian dan evaluasi komponen dapat dilihat pada gambar berikut:



Gambar 3.8. Diagram Component Aplikasi *E-Commerce* MJ Elektronik

3.3. Implementasi dan Pengujian

Langkah-langkah pembuatan komponen pada *framework yii* dijelaskan pada Gambar 4.1.

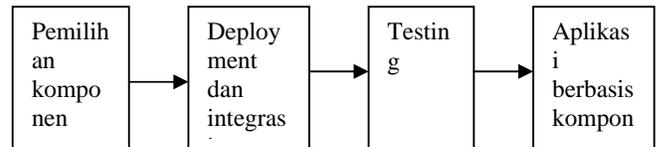


Gambar 4.1. Tahapan Pembuatan Komponen

Langkah-langkah analisis dan desain telah dilakukan dan dibahas pada bab III, maka langkah selanjutnya adalah membuat komponen itu sendiri. Pada arsitektur komponen dalam *Framework Yii*, sebuah komponen adalah sebuah file berekstensi *.php*

yang berisi *class-class* yang dibutuhkan. Dalam *Framework yii* komponen dimasukkan ke dalam folder bernama *components*.

Pemakai komponen mempunyai tugas utama untuk memilih komponen-komponen yang diperlukannya untuk membangun sebuah sistem atau aplikasi yang utuh, kemudian mengintegrasikan agar saling terhubung dan aplikasi dapat dijalankan. Langkah-langkah pemakaian komponen pada dijelaskan pada Gambar 4.2.



Gambar 4.2. Tahapan Pemakaian Komponen

Pemilihan komponen yang sesuai dengan kebutuhan sistem dapat dilakukan dengan cara melihat *interface* yang ada. Setiap komponen memiliki *interface* yang menampilkan *service* yang disediakan. Setelah menemukan komponen yang sesuai dengan sistem maka dapat memilih *service* apa yang ingin dipakai. Jadi tidak semua *service* yang ada pada komponen dibutuhkan.

Untuk dapat menggunakan komponen yang tersedia harus dilakukan konfigurasi untuk mengintegrasikan agar komponen terhubung dengan sistem. Tahap selanjutnya adalah memanggil *service* yang dibutuhkan oleh sistem. Setelah komponen terintegrasi dengan sistem, tahap terakhir dilakukan pengujian pada *service* komponen tersebut.

Pengujian aplikasi *E-Commerce* Toko Mustika Jaya dilakukan dengan metode *black box*, yaitu menguji fungsionalitas dari perangkat lunak, tanpa harus mengetahui struktur internal program. Sebuah perangkat lunak yang diuji menggunakan metode *black-box* dikatakan berhasil jika fitur-fitur yang ada telah memenuhi spesifikasi *requirements (Use Case)* yang telah dibuat sebelumnya.

Lingkungan pengujian meliputi perangkat keras dan perangkat lunak yang digunakan dalam proses pengujian. Perangkat keras yang digunakan dalam pengujian aplikasi *E-Commerce* Toko Mustika Jaya ini adalah :

- 1) CPU : AMD Turion(tm) II Dual-Core Mobile M520 (2CPUs) 2.3Ghz

- 2) Memory : 3072 MB RAM
- 3) Harddisk 500 GB

Software atau perangkat lunak yang digunakan pada proses pengujian perangkat lunak *E-Commerce* adalah sebagai berikut :

- 1) Operating System : Windows 7 Ultimate 32-bit
- 2) Google Chrome 27.0.1453.116
- 3) Web Server : Apache 2.2.21
- 4) DBMS : MySQL 5.5.16

4. KESIMPULAN DAN SARAN

4.1. Kesimpulan

Kesimpulan yang dapat diambil dari pembuatan tugas akhir ini yaitu:

1. Dihilangkan sebuah aplikasi berupa perangkat lunak *E-Commerce* yang dikembangkan dengan menggunakan model proses *Component-based Software Engineering*(*CBSE*).
2. Langkah-langkah dalam model proses *CBSE* diawali dengan tahap definisi kebutuhan sistem yang meliputi spesifikasi sistem dan analisa kebutuhan sistem dilanjutkan dengan perancangan sistem untuk menentukan komponen-komponen yang dibutuhkan sesuai kebutuhan sistem. Langkah selanjutnya yaitu implementasi basis data, *Class*, pembuatan komponen, pemakaian komponen dan antarmuka sistem. Langkah terakhir adalah pengujian sistem berdasarkan *requirements* yang sudah ditetapkan.
3. Cara memanfaatkan komponen yang sudah ada dalam pengembangan aplikasi *E-Commerce* ini yaitu melakukan konfigurasi

pada file *main.php* dengan menambahkan *source code* komponen-komponen yang dipilih. Cara pemanggilan *service-service* yang disediakan oleh komponen yaitu dengan menambahkan *source code service* yang dibutuhkan pada *Class-Class MVC* sesuai *requirements* yang sesuai dengan *service* yang disediakan oleh komponen.

4. Penggunaan model proses *CBSE* dapat memudahkan dalam pengembangan aplikasi *E-Commerce* Toko Mustika Jaya Elektronik dengan adanya komponen-komponen yang sudah tersedia dan setiap komponen yang ada mempunyai sifat *reusable* yang tinggi, sehingga dapat langsung dipakai dan mudah dalam penggunaannya.

4.2. Saran

Pengembangan aplikasi *E-Commerce* Toko Mustika Jaya Elektronik dengan mengimplementasikan model proses *Component-based Software Engineering* ini masih mempunyai kekurangan. Diantaranya dalam pemakaian komponen yang disediakan *framework Yii* masih perlu melakukan konfigurasi dengan menambahkan *source code* pada file *main* aplikasi utama dan belum mampu secara *drag-and-drop*. Diharapkan untuk kedepannya pengembang *framework Yii* ini mampu meningkatkan tingkat pemakaian komponen yang semakin mudah bagi pengembang dengan menggunakan bahasa pemrograman PHP.

Pembuatan beberapa komponen baru pada aplikasi *E-Commerce* MJ Elektronik ini hanya mencakup pada tipe aplikasi *I-Market* saja. Diharapkan bagi pengembang aplikasi *E-Commerce* berbasis komponen dapat menambahkan komponen-komponen pada tipe aplikasi *Customer Care*, *Management Vendor* dan *Extended Supply Chain*, sehingga dapat dijadikan sebagai referensi bagi

pengembang aplikasi *E-Commerce* secara keseluruhan.

5. DAFTAR PUSTAKA

- [1] Afriyudi. 2008. *Pemrograman Web Dinamis dengan Kolaborasi PHP dan Java*. Andi.
- [2] Booch, Grady, et all. 2005. *The Unified Modeling Language User Guide SECOND EDITION*. Addison Wesley Professional.
- [3] Crnkovic, Ivica, Michel Chaudron, and Stig Larsson. "Component-based development process and component lifecycle." *Software Engineering Advances, International Conference on*. IEEE, 2006.
- [4] Fingar, Peter. 2000. *Component-Based Framework For E-Commerce*. Wasington: Communications of the ACM.
- [5] Kadir, Abdul. 2009. *From Zero to A Pro: Membuat Aplikasi Web dengan PHP dan Database MySQL*. Yogyakarta: Andi Ofsset.
- [6] Laudon, Kenneth C., and Carol Guercio Traver. 2009. *E-Commerce bussiness, technology, society*. Prentice Hall; 5 edition.
- [7] Lupiyadi, Rambat. 2001. *Manajemen Pemasaran Jasa Teori dan Praktik*. Jakarta : Salemba Empat.
- [8] Nabi, Faisal, and Roisin Mullins. "Moving From Traditional Software Engineering to Componentware." *JSEA* 4.5 (2011): 283-292.
- [9] Nandiwardhana, Yoga. 2003. *Penggunaan Pendekatan Component-Based Software Enineering (CBSE) untuk Mengembangkan E-Catalogue sebagai Komponen Perangkat Lunak*. Depok, Universitas UI.
- [10] Pressman, Roger S. 2001. *Software Engineering : A Practitioner's Approach Fifth Edition*. New York: McGraw-Hill Companies, Inc.
- [11] Rumbaugh, James, et all. 2004. *The Unified Modelling Language Reference Manual Second Edition*. Boston: Addison Wesley.
- [12] Sommerville, Ian. 2004. *Software Engineering, 7th edition*. Addison-Wesley.
- [13] Szyperski, Clemens. 1999. *Component Software Beyond Object-Oriented Programming*. Addison-Wesley.
- [14] Winesett, Jeffrey. 2010. *Agile Web Application Development with Yii 1.1 and PHP 5*. Birmingham: Packt Publishing.