

PERANCANGAN SISTEM TRANSAKSI *ONLINE* PADA *E-COMMERCE* MENGGUNAKAN TEKNOLOGI *WEB SERVICE* (STUDI KASUS DESA MLATIHARJO KABUPATEN DEMAK)

Rendra Bagus Setiawan¹
672005050@student.uksw.edu

Rudy Latuperissa²
rudi.latuperissa@staff.uksw.edu

Supriyadi³
supriyadi@staff.uksw.edu

Abstract

Economy development policy in Indonesia focuses on increasing local or regional economy, especially the real sector including trade sector. One of the problems today is how to improve and strengthen the region potential, for example cooperatives, SMEs and cluster in using information technology such as e-Commerce. Therefore, this research designs online transaction system on e-Commerce using web service technology. The case study used in the process of design is Mlatiharjo village in Demak because it has the predicate of innovative village of agriculture. This system is developed using prototyping method and implemented using PHP 5 technology, MySQL database and supported by web service SOAP technology. Thus, online transaction system in the e-Commerce is able to manage and control authority for the village potential. By facilitating this village with the developed system, every people in the village could contribute to the commerce transaction of their products.

Keywords: *e-Commerce, Web Service, Mlatiharjo*

1. Permasalahan

Kebijakan peningkatan ekonomi Indonesia sekarang ini mengacu pada kebijakan ekonomi real yang berfokus pada peningkatan kegiatan ekonomi lokal. Untuk mendukung kebijakan tersebut, salah satu yang dapat dilakukan dalam bidang teknologi informasi yaitu dengan membangun suatu model *e-Commerce* sebagai indikator pencapaian ekonomi lokal suatu daerah. Selain itu, *e-Commerce* juga dapat memberdayakan potensi-potensi yang ada pada suatu unit pemerintah daerah yang paling kecil yaitu tingkat desa, contoh koperasi. Sedangkan proses bisnis pada *e-Commerce* yang ada saat ini belum dapat memberdayakan potensi-potensi yang dimiliki pada sebuah desa.

Pada penelitian ini, studi kasus adalah Desa Mlatiharjo Kecamatan Gajah Kabupaten Demak yang mewakili desa-desa yang ada Indonesia, yang memiliki produk-produk unggulan yang mampu bersaing di pasaran dan layak untuk dijadikan pasar digital, serta masyarakat usaha yang ada di desa telah banyak tergabung dalam sebuah koperasi bernama Koperasi Serba Usaha Citra Kinaraya. Rumusan masalah penelitian adalah bagaimana merancang suatu transaksi *online* yang didalam *e-Commerce* yang dapat mengintegrasikan kewenangan beberapa lembaga potensi desa, bagaimana memanfaatkan teknologi *web service* untuk mendukung sistem transaksi *onlinetersebut* dan bagaimana membangun

¹Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Salatiga

²Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Salatiga

³Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Salatiga

aplikasi sistem transaksi *online* dengan menerapkan teknologi *web service* menggunakan teknologi PHP dan MySQL.

2. Usulan Solusi

Untuk dapat memberdayakan potensi-potensi desa antara lain kelompok usaha masyarakat desa dan badan usaha koperasi, maka pada penelitian ini bertujuan untuk membangun aplikasi sistem transaksi pembayaran *online* untuk *e-Commerce* Mlatiharjo dengan menerapkan teknologi *web service* menggunakan bahasa pemrograman PHP dan MySQL yang memiliki fasilitas untuk pembagian tugas atau wewenang antara potensi-potensi desa dalam hal pembayaran *online* dalam *e-Commerce*. Manfaat penelitian ini bagi Desa Mlatiharjo yaitu bahwa desa memiliki media untuk pemasaran produk dengan teknologi *web*, memberikan layanan transaksi pembayaran *online* bagi *member* tetap, bagi masyarakat umum dapat melihat produk Desa Mlatiharjo dan bagi akademik dapat menjadi bahan pembelajaran tentang transaksi *online* menggunakan *web service* pada dua buah *web*.

Batasan masalah dari penelitian ini adalah transaksi pembayaran *online* yang dimaksud adalah 1) transaksi pembayaran tanpa melibatkan pihak dari luar, 2) tidak membahas keamanan *content web*, 3) perancangan aplikasi *web* mengacu pada *website* Desa Mlatiharjo yaitu www.mlatiharjo.com, 4) transaksi *online* yang ada pada aplikasi menggunakan *web service* hanya bersifat *administrative*, 5) *web e-Bank* hanya sebatas nama aplikasi yang dipergunakan untuk mewakili koperasi KSU Citra Kinaraya, 6) tidak menganalisis dan membandingkan konsep *web service* tapi hanya mengimplementasikan konsep teknologi *web service* menggunakan SOAP pada suatu *e-Commerce*, yang bertujuan untuk mengelola bagian kewenangan pada kelembagaan di desa sehingga dapat memberdayakan potensi kelembagaan pada suatu desa tersebut, dan 7) tidak membahas dampak dari hasil *e-Commerce* seperti peningkatan penjualan dan perilaku pasar.

Sistem transaksi adalah pertukaran komoditas atau layanan melalui pasar atau instansi pengatur lainnya dan juga dapat berarti sebagai unit terbesar dari kegiatan ekonomi yang tidak dapat dibagi untuk tujuan transaksi antara 2 atau lebih pelaku ekonomi (Rao, 2003).

Dalam teknologi komputer, *online* merujuk kepada keadaan sistem (contohnya *personal computer*) terhubung ke jaringan komunikasi seperti internet. Istilah ini sudah berkembang untuk merujuk pada sebuah aktivitas (seperti kegiatan edukasi melalui *computer conferencing* atau forum) (Harasim, 2011).

Bagi masyarakat pada umumnya, istilah "*electronic commerce*" berarti belanja pada dari internet yang disebut *world wide web (www)*. Namun, *e-Commerce* juga mencakup banyak kegiatan lain, seperti pertukaran pebisnis dengan pebisnis yang lain dan proses internal yang digunakan perusahaan untuk mendukung pembelian, penjualan, perekrutan dan perencanaan. Kelebihan *e-Commerce* yaitu dapat meningkatkan penjualan dan mengurangi biaya produksi (Schneider, 2010).

Pembangunan *web service* pada PHP menggunakan *library SOAP extension*. Dokumen WSDL digunakan untuk mendeskripsikan layanan yang disediakan aplikasi (Ahmadi, 2007). *Web Service* adalah aplikasi jaringan *interface* yang dapat diakses, dibangun dengan menggunakan teknologi internet standar. Dengan kata lain, jika sebuah aplikasi dapat diakses melalui jaringan menggunakan kombinasi protokol seperti HTTP, XML, SMTP, atau Jabber, maka itu adalah *web service* (Snell & Tidwell, 2002).

Web service memiliki tiga entitas dalam arsitekturnya yaitu a) *Service Provider*: Berfungsi untuk menyediakan layanan/ *service*. b) *Service Registry*: Berfungsi sebagai lokasi central yang mendeskripsikan semua layanan/ *service* yang telah di-*register*. c) *Service Consumer*: Peminta layanan yang mencari dan menemukan layanan yang dibutuhkan serta menggunakan layanan tersebut (Short, 2002). Dalam *Building Block* ada beberapa tahapan yaitu:

- 1) *Discovery: Universal Description Discovery and Integration (UDDI)*, adalah sebuah *service registry* bagi pengalokasian *web service*. UDDI mengkombinasikan SOAP

- (Simple Object Access Protocol) dan WSDL (*Web Service Definition Language*) untuk pembentukan sebuah *registry* API bagi pendaftaran dan pengenalan *service*.
- 2) *Description* :WSDL, merupakan suatu standar bahasa dalam format XML yang berfungsi untuk mendeskripsikan seluruh layanan yang tersedia.
 - 3) *Message Format* dipakai untuk pertukaran data antara klien dan *server*. Sebuah cara standar pengkodean data memastikan bahwa data yang dikodekan oleh klien akan benar ditafsirkan oleh *server*.
 - 4) *Encoding Data*, data yang ditransmisikan antara klien dan *server* harus dikodekan kedalam tubuh pesan.
 - 5) *Transport*. Setelah pesan telah diformat dan data telah serial ke dalam tubuh pesan, maka pesan tersebut harus ditransfer antara klien dan *server* melalui beberapa *protocol transport* (Short, 2002).

PHP berjalan pada semua sistem operasi utama, dari varian Unix termasuk Linux, FreeBSD, dan Solaris ke Windows dan Mac OS X. Hal ini dapat digunakan dengan semua *server web* terkemuka, termasuk Apache, Microsoft IIS, dan Netscape/iPlanet *server*. Bahasa PHP sangat fleksibel, sebagai contoh PHP tidak terbatas hanya *output* format HTML atau teks dokumen saja yang dapat dihasilkan. PHP memiliki kemampuan mendukung untuk menghasilkan *file* PDF, GIF, JPG, dan gambar PNG, dan film. Salah satu fitur PHP yang paling signifikan adalah dukungan luas untuk basis data. PHP mendukung semua *database* utama (termasuk MySQL, PostgreSQL, Oracle, Sybase, dan ODBC). Oleh sebab itu, dengan PHP membuat halaman *web* dengan konten dinamis dari basis data menjadi lebih mudah (Lerdorf& Tatroe, 2006).

Metode dalam pembuatan sistem ini adalah menggunakan *prototyping* model yang membantu *customer* dan *developer* dalam memahami *requirements* yang dibutuhkan dalam suatu sistem. *Prototyping* dianggap sebagai kegiatan pengurangan resiko (Sommerville, 2006). *Prototyping* dipilih karena seringkali *customer* mendefinisikan satu tujuan umum untuk perangkat lunak tetapi tidak mengidentifikasi *input* secara detil, pemrosesan, atau persyaratan *output*. Sedangkan *developer* sering tidak memiliki kepastian terhadap efisiensi dari algoritma, kemampuan adaptasi dari sistem, atau bentuk interaksi manusia/mesin yang harus diambil. Dalam hal ini, paradigma *prototyping* menawarkan pendekatan yang terbaik (Pressman, 2010). Tahapan *prototyping*:

1. Pengumpulan kebutuhan data Desa Mlatiharjo dan produk yang dihasilkan dengan mengambil dari *website* Desa Mlatiharjo yaitu www.mlatiharjo.com dan dengan melakukan survei langsung.
2. Perancangan *prototype* dimulai dengan merancang tampilan antarmuka aplikasi dan perancangan proses aplikasi.
3. Evaluasi. Tahap ini dilakukan pengujian aplikasi dengan seorang penduduk Desa Mlatiharjo selaku admin dan dosen pembimbing. Aplikasi yang dirasa kurang sesuai, maka diadakan perbaikan aplikasi mulai dari pengumpulan data. Aplikasi yang sudah sesuai dengan yang diinginkan, maka proses selesai.

3. Proses Integrasi Data pada 2 Aplikasi yaitu *Web* Mlatiharjo dan *Web e-Bank*

Mekanisme *member* mendapatkan saldo di *webe-Bank* yaitu *member* (pembeli) mentransfer uang ke nomor rekening admin di bank dan dari transfer tersebut *member* akan mendapatkan bukti transfer. Dengan bukti transfer tersebut, *member* memberitahukan kepada admin telah men-transfer sejumlah uang ke nomor rekening admin. Kemudian admin mengecek kebenaran transfer tersebut di bank dan setelah itu admin meng-input saldo sesuai dengan jumlah yang telah ditransfer oleh *member* dan saldo terakhir *member* bertambah.



Gambar 1. Mekanisme Pembelian dan Pembayaran Member Pembeli

Gambar 1 menjelaskan bahwa pembeli melakukan pembelian dan pembayaran dimulai dengan pembeli melakukan pemesanan di *web* Mlatiharjo, dan dari pemesanan tersebut *member* akan mendapatkan ID pesanan. Dari ID pesanan, pembeli dapat melakukan pembayaran pada *webe-Bank*. Dari *web e-Bank* akan dikirimkan data pembayaran oleh pembeli ke *web* Mlatiharjo melalui *web service*. Kemudian status pada pemesanan pembeli dari pesan akan berubah menjadi bayar.



Gambar 2. Mekanisme Konfirmasi Pelunasan Pembayaran Pembeli

Gambar 2 menjelaskan mekanisme konfirmasi pelunasan pembayaran pembeli oleh admin yaitu data pembayaran oleh pembeli pada *webe-Bank* dikirimkan ke *web* Mlatiharjo melalui *web service*. Status pembayaran pembeli di menu admin akan berubah dari pesan ke bayar. Setelah itu, admin akan mengkonfirmasi pembayaran tersebut dan mengirimkan data kepada penjual dengan perubahan status dari bayar menjadi kirim untuk memberitahukan kepada penjual untuk mengirimkan barang yang sudah dibayar oleh pembeli.

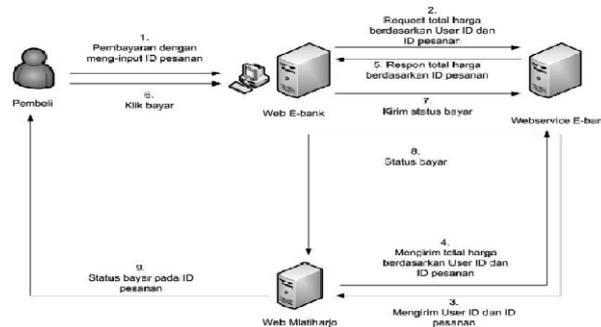
Mekanisme perancangan konfirmasi pengiriman barang oleh penjual yaitu pengiriman data pembelian yang sudah lunas oleh admin kepada penjual di *web* Mlatiharjo yang merupakan pemberitahuan bagi penjual untuk pengiriman barang. Setelah penjual mengirim barang, maka penjual akan mengkonfirmasi pengiriman barang yang kemudian data konfirmasi ini akan masuk ke halaman pembeli sebagai pemberitahuan bahwa barang yang dibeli telah dikirim oleh penjual dan perubahan status pada tanggal kirim dari “belum dikirim” berubah menjadi tanggal pengiriman. Pada saat pengiriman barang, maka penjual akan mendapat nomor resi yang akan dikirimkan kepada admin dan admin akan mengirimkan kepada pembeli melalui *email* sebagai bukti bahwa barang yang dibeli telah benar-benar dikirim.

3.1. Perancangan *web service*

Web Service merupakan alat penghubung antara *web* Mlatiharjo dan *webe-Bank*. Dalam penelitian ini *web Service* digunakan dalam proses pembayaran di *webe-Bank* dari pembelian di *web* Mlatiharjo. Teknologi *web service* dipergunakan interoperabilitas pada kedua aplikasi *web* tersebut yaitu pada transaksi pembelian dan pembayaran dapat terwujud. Interoperabilitas dan komunikasi antar kedua aplikasi *web*, yaitu *web* Mlatiharjo milik masyarakat usaha (penjual) dan *web e-Bank* milik koperasi KSU Citra Kinaraya terjadi karena proses bisnis yang disepakati oleh pihak desa Mlatiharjo yaitu penjual dan pembeli

harus menjadi member *web e-Bank*. Hal ini dipergunakan sebagai jaminan transaksi jual beli pada aplikasi *e-Commerce* milik desa Mlatiharjo.

Teknologi *web service* juga memberi manfaat bagi terjaminnya kondisi keamanan integritas masing-masing aplikasi, sekaligus memberi kesempatan pengembangan jika dibutuhkan integrasi dengan aplikasi yang lain yang dimiliki oleh lembaga lain atau lembaga desa sebagai potensi desa.



Gambar 3. Mekanisme Web Service Member Melakukan Pembayaran

Gambar 3 adalah mekanisme *member* melakukan pembayaran. *Member* melakukan *input* ID pesanan di *webe-Bank* yang didapat dari pembelian di *web* Mlatiharjo. *Webe-Bank* meminta total harga pembelian berdasarkan User ID dan ID pesanan kepada *web* Mlatiharjo melalui *web service*. *Web* Mlatiharjo mengirim total harga berdasarkan user ID dan ID pesanan melalui *web service* dan *web service* merespon dengan mengirim total harga ke *webe-Bank*. Total harga muncul dan *member* mengklik bayar, *Web e-Bank* meminta *web service* untuk mengirim status bayar ke *Web* Mlatiharjo. Status ID pesanan pada *Web* Mlatiharjo dari pesan berubah menjadi bayar

3.2. Perancangan Aplikasi

Aplikasi yang dibangun menggunakan teknik pemrograman terstruktur supaya aplikasi dapat diimplementasikan pada *web server* yang memiliki berbagai versi PHP. Bahasa pemodelan yang digunakan dalam perancangan aplikasi adalah *Unified Modelling Language*.

a. Use Case Diagram

Use case yang ada pada aplikasi ini dibagi dalam dua bagian, yaitu *use case* Mlatiharjo dan *use case e-Bank*. Aktor yang ada pada kedua *use case* diagram tersebut adalah admin, pembeli, penjual dan *guest*. Aktor admin dapat melakukan pengolahan keseluruhan data yang ada pada sistem seperti melihat data pembeli, penjual, barang, pemesanan, pembayaran dan histori pembelian *member* serta melakukan proses delete jika tidak sesuai dengan ketentuan yang berlaku di *web* Mlatiharjo. Aktor pembeli dapat melakukan perencanaan pemesanan yang disimpan di keranjang belanja, pemesanan barang pada *browse* produk Mlatiharjo, lihat pemesanan jika pembeli sudah yakin memesan serta dapat mengedit data pribadi. Aktor penjual dapat melakukan pengolahan barang, melihat barang penjual lain di *browse* barang penjual, dan mengedit data pribadi. Sedangkan aktor *guest* hanya dapat melihat halaman utama, selang pandang, geliat ekonomi desa dan produk-produk yang dijual di *web* Mlatiharjo di pasar desa *online* dan melakukan pendaftaran menjadi *member*.

Pada *use case e-Bank*, Admin dalam *web* Mlatiharjo juga akan menjadi admin *e-Bank* dan penjual dan pembeli di Mlatiharjo, dalam *e-Bank* akan tergabung menjadi satu yaitu aktor *member*. Dalam hal ini, penjual atau pembeli yang tergabung dalam Mlatiharjo akan otomatis terdaftar dalam *e-Bank*. Di *e-Bank*, aktor admin dapat melakukan pengolahan saldo *member* di transaksi serta dapat melihat laporan transaksi yang dilakukan oleh *member*. Sedangkan aktor *member* dapat melihat saldo akhir, transfer rekening sesama

member, melakukan pembayaran dari pemesanan di Mlatiharjo serta melihat laporan transfer rekening yang dilakukan oleh member yang bersangkutan.

b. Activity Diagram

Activity Diagram yang akan ditampilkan adalah pembelian barang dan pembayaran dimana pembeli melakukan pencarian barang pada browse produk Mlatiharjo dan melakukan perencanaan pemesanan yang disimpan di keranjang belanja. Dalam keranjang belanja ini, pembeli dapat menghapus daftar barang yang tidak diinginkan atau membatalkan secara keseluruhan dan jika sudah yakin maka pembeli dapat melakukan proses pemesanan yang akan masuk pada lihat pemesanan. Pada lihat pemesanan, pembeli dapat melihat detail pembelian dan mencetaknya dan aktivitas pembelian berakhir.

Activity diagram pembelian dapat dijelaskan bahwa member melakukan login, jika user name dan password benar maka akan masuk pada halaman utama yang berisi informasi saldo, mutasi rekening, transfer rekening dan pembayaran e-Commerce.

4. Pengujian

Prototype ini masih bersifat localhost karena sistem ini masih merupakan “model” e-Commerce untuk Desa Mlatiharjo yang diharapkan mampu mewakili potensi-potensi yang ada di Desa Mlatiharjo seperti Koperasi Serba Usaha (KSU) Citra Kinaraya yang di dalam sistem ini diwakilkan dengan e-Bank dan produk-produk dari masyarakat desa Mlatiharjo yang di-inputkan didalam aplikasi e-Commerciweb Mlatiharjo.

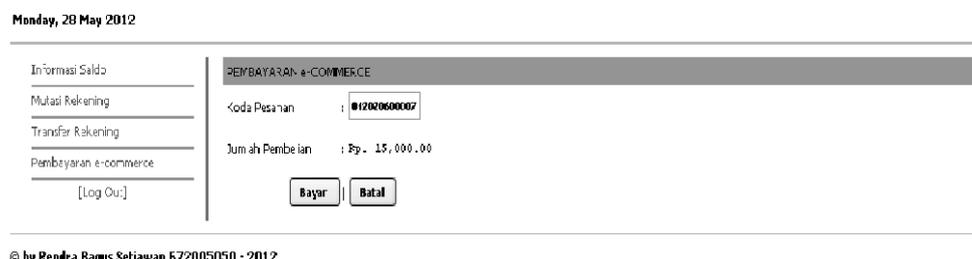
Pengujian aplikasi dijalankan pada PHP versi 5.3.10, MySQL versi 5.5.20, dan Apache 2.2.21. Pengujian ini difokuskan pada pengujian kinerja web service yaitu dengan caradijelaskan dalam bentuk gambar yang berisi langkah-langkah dalam aplikasi yang menggunakan web service.



No	ID Pesanan	Tanggal	Total (Rp)	Status	Lihat Data	Invoice
1	2012020600007	28-05-2012	15,000	Pesan	view	Cetak

Gambar 4.Data Pemesanan Pembeli

Pada saat pembelian, pembeli akan mendapatkan ID pesanan yang akan digunakan sebagai acuan proses pembayaran pada web e-Bank.Gambar 4 merupakan daftar pesanan pembeli pada web Mlatiharjo.



Monday, 28 May 2012

Informasi Saldo
Mutasi Rekening
Transfer Rekening
Pembayaran e-commerce
[Log Out]

PEMBAYARAN e-COMMERCE

Koda Pesanan : 2012020600007

Jumlah Pembelian : Rp. 15.000.00

Bayar Batal

© by Rendra Bagus Setiawan 672005050 - 2012

Gambar 5.Pembayaran e-Commerce pada web e-Bank

Gambar 5 memperlihatkan bahwa pada halaman pembayaran e-Commerce, pembeli menginput ID pesanyang didapat dari pembelian di web Mlatiharjo sebagai request dan akan mendapatkan response total harga melalui web service.

No	ID Pesanan	Tanggal	Tanggal Kirim	Total(Rp)	Status	Lihat Data
1	2012020600001	06-02-2012	Belum dikirim!	15,000	Bayar	view
2	2012020600003	06-02-2012	Belum dikirim!	5,000	Kirim	view
3	2012020600005	21-05-2012	21-05-2012	5,000	Kirim	view
4	2012020600006	22-05-2012	22-05-2012	15,000	Kirim	view
5	2012020600007	28-05-2012	Belum dikirim!	15,000	Bayar	view

Gambar 6. Lihat History Pembelian

Setelah pembeli meng-klik bayar, maka terjadi perubahan status dari pesan menjadi bayar pada halaman Lihat History Pembelian seperti pada Gambar 6.

Melihat dari pengujian *web service* di atas, maka penggunaan *web service* dapat dikembangkan yaitu misalnya jika ada *web* lembaga desa yang akan bergabung dalam *e-Commerce*. Contoh adalah lembaga yang akan mengurus pengiriman barang dari pembelian barang di *e-Commerce*, dapat dilakukan langkah-langkah seperti menambahkan alamat *website* departemen baru pada WSDL dengan cara mengganti/*update* pada *mlatiharjo.wsdl*, dan pada *server.php* ditambah layanan *response* jika ada *request* dari *web* lembaga baru tersebut. Dengan demikian hanya merubah *source code* yang mengelola *web service* pada aplikasi yaitu pada *mlatiharjo.wsdl* dan *server.php*.

Kode Program 1. Mlatiharjo.wsdl

```

1 <message name='getQuoteRequest'>
2 <part name='symbol' type='xsd:string' />
3 <part name='kode' type='xsd:string' />
4 </message>
5 <message name='getQuoteResponse'>
6 <part name='Result' type='xsd:float' />
7 </message>
8
9 <portType name='StockQuotePortType'>
10 <operation name='getQuote'>
11 <input message='tns:getQuoteRequest' />
12 <output message='tns:getQuoteResponse' />
13 </operation>
14 </portType>

```

WSDL adalah definisi method yang terdapat di *web service* yang digunakan untuk berkomunikasi.

getQuoteRequest adalah *method* mengambil *request id* yang ingin dibayar.

getQuoteResponse adalah nilai kembali dari *method request* pembayaran

Kode Program 2. Server.php

```

1 $server = new
2 SoapServer("http://localhost/mlatiharjo/mlatiharjo.wsdl");
3 $server->setClass("QuoteService");
4 $server->handle();

```

SoapServer adalah *class* untuk menggunakan *SOAP protocol*

Lokasi dari *SoapServer* adalah letak file *wsdl* dari *web service*

setClass mendefinisikan *class* yang akan digunakan didalam *web service*

Kode Program 3. Perintah mengambil mengambil kode dan total transaksi melalui web service

```
1 Class QuoteService {
2 private $quotes = array("ckdbeli" => 0, "pay" => 0);
3 function getQuote($symbol, $kode) {
4 require_once("function/koneksi.php");
5
6 if ($kode!="pay"){
7     $query = "select * from tbhbeli where ckdbeli='$symbol' and
8 cstatus=0";
9     $result = mysql_query($query) or die(mysql_error());
10    $row = mysql_fetch_array($result);
11
12    $quotes["ckdbeli"] = $row['ntotal'];
13
14    if (isset($this->quotes["ckdbeli"])) {
15        return $quotes["ckdbeli"];
16    } else {
17        throw new SoapFault("Server","Unknown Symbol '$symbol'.");
18    }
19 }
20 }
21 }
```

Class QuoteService yang didefinisikan merupakan class yang mengambil kode transaksi yang dilakukan oleh pembeli untuk melakukan pembayaran. Selanjutnya akan dipanggil *method getQuote* untuk kode transaksi oleh pembeli dan mengembalikan total pembayaran. Kemudian pembayaran dilakukan untuk melunasi transaksi yang telah dilakukan oleh pembeli. Database dilakukan modifikasi didalam *server.php*.

5. Kesimpulan

Kesimpulan penelitian ini antara lain, yaitu: a) Teknologi *web service* dipergunakan sebagai sistem pendukung transaksi pembayaran *online* pada suatu aplikasi *e-Commerce*. b) *Web service* dapat digunakan sebagai manajemen pengendalian proses atau layanan antar badan usaha di suatu desa dalam suatu *e-Commerce*. c) Teknologi *web service* membantu mempermudah pihak *developer* dalam hal *maintenance* dan operasional terutama jika ada perubahan pengembangan layanan transmisi data antar badan usaha yang ada di suatu desa pada suatu aplikasi. Sistem transaksi *online* ini dapat dikembangkan lebih baik dengan ditambahkan layanan *sms gateway* pada *web e-Bank* yang dihubungkan dengan *sms banking* perbankan sehingga memberikan fasilitas otomatis konfirmasi pembayaran dari perbankan oleh pembeli kepada pengelola koperasi, pada bagian implementasi. Pemrograman dengan bahasa PHP dapat dikembangkan dengan menerapkan konsep *object oriented programming* secara optimal dan dapat ditambahkan *web service security* untuk menjamin keamanan transmisi data.

Daftar Pustaka

- Ahmadi, I. (2007). *Implementasi web service pada content management system*. (Undergraduate's thesis, Institut Teknologi Bandung, Bandung, Indonesia) Diakses dari <http://digilib.itb.ac.id/gdl.php?mod=browse&op=read&id=jbptitbpp-gdl-imamahmadi-28956>.
- Harasim, L. M. (2011). *Learning theory and online technology*. New York, NY: Routledge.
- Lerdorf, R., & Tatroe, K. (2006). *Programming PHP* (2nd ed.). Sebastopol, Calif.: O'Reilly.
- Pressman, R. S. (2010). *Software engineering: a practitioner's approach* (7th ed.). New York: McGraw-Hill Higher Education.
- Rao, P. K. (2003). *The Economics of Transaction Costs: Theory, Methods and Applications*. New York: Palgrave Macmillan.
- Schneider, G. (2010). *Electronic Commerce* (9 ed.). Boston: Couse Technology, Cengage Learning™.
- Short, S. (2002). *Building XML web services for the Microsoft .NET platform*. Redmond, Wash.: Microsoft Press.
- Snell, J., & Tidwell, D. (2002). *Programming Web services with SOAP*. Sebastopol, CA: O'Reilly & Associates.
- Sommerville, I. (2000). *Software engineering* (6th ed.). Harlow, England: Addison-Wesley.