

Aplikasi *Web Service* Perbankan untuk Pembayaran Formulir Pendaftaran Mahasiswa Baru

Puspaningtyas Sanjoyo Adi

Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Sanata Dharma,
Kampus III Universitas Sanata Dharma, Paingan, Maguwoharjo, Depok, Yogyakarta 55282,
Telp: 0274-883037, 883968, ext.: 52341, 52320
e-mail: puspasa@usd.ac.id

Abstract

The new student registration process begins with the process of filling out the registration form where to get it, the prospective student pays to get the form. Sanata Dharma University cooperates with the bank to develop an application to pay registration form with web service technology. The advantage of this technology is electronic service and can work 24 hours. Web service technology is a service that allows communication between machines performed through a computer network with HTTP protocol web server. The web service application was developed and implemented by Sanata Dharma University and CIMB Niaga bank in October 2017. From the observation of January 2018 compared to January 2017, the January 2018 transaction increased by about 12% compared to January 2017. The proportion of web service transactions is around 30% of total transactions where 80% of them are conducted outside office hours. This data indicates that the knowledge of electronics services of prospective students and their parents is good enough.

Keywords: *web service, registration form, new students*

1. Pendahuluan

Proses penerimaan mahasiswa baru diawali dengan proses pendaftaran calon mahasiswa. Pada umumnya, proses pendaftaran dilakukan dengan cara mengisi formulir pendaftaran yang biasanya pemerolehan formulir dilakukan dengan cara membeli formulir tersebut. Dana pembelian formulir tersebut umumnya dialokasikan untuk proses administrasi berkas pendaftaran calon mahasiswa tersebut hingga ada keputusan diterima atau ditolaknya calon mahasiswa tersebut.

Proses pembelian formulir pendaftaran bisa berbeda antara satu universitas dengan universitas lainnya. Tulisan ini mengambil studi kasus Universitas Sanata Dharma Yogyakarta, di mana proses pembelian formulir di bank dilakukan di awal proses, lalu mendapat kode bayar dan selanjutnya dengan kode bayar yang diterimanya, calon mahasiswa meneruskan proses pengisian formulir pendaftaran. Beberapa universitas ada yang menerapkan proses pemerolehan akun terlebih dahulu lalu mendapat kode bayar dan membayar di bank lalu mendapat kode verifikasi. Calon dengan menggunakan kode verifikasi tersebut melanjutkan proses pengisian formulir. Secara sederhana, proses ada 2 yaitu pertama, "bayar dulu baru isi formulir", kedua, "isi dulu formulir baru bayar". Kelebihan proses pertama dibanding proses kedua adalah pengurangan calon mahasiswa yang iseng mendaftar, karena diharuskan membayar terlebih dahulu.

Kebijakan "bayar dulu baru isi formulir" ditetapkan oleh Universitas Sanata Dharma bekerjasama dengan Bank BRI yang memfasilitasi dengan aplikasi *host-to-host* pada teller bank BRI. Prosedur dapat dilihat di website <http://pmb.usd.ac.id>. Aplikasi ini akan memberikan kode bayar kepada setiap calon mahasiswa setiap transaksi. Kendala yang dihadapi pada aplikasi pembelian formulir di layanan teller bank adalah waktu layanan di luar jam sekolah. Hal ini

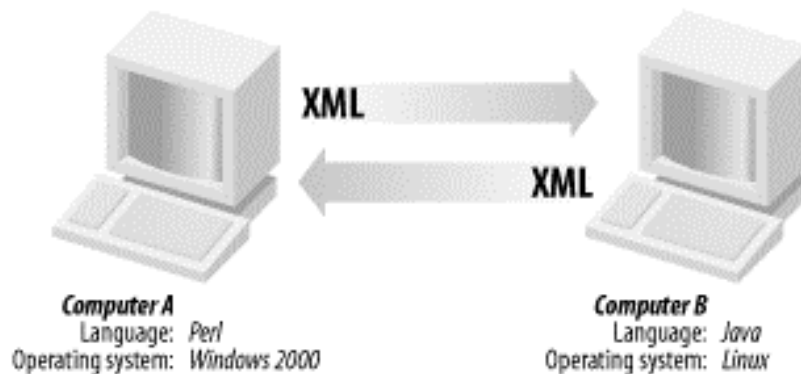
terjadi karena calon mahasiswa adalah siswa SMU yang jam belajarnya bersamaan dengan jam layanan bank BRI sehingga tidak bisa mengakses layanan teller bank BRI di luar jam belajar sekolah. Di sisi lain, bank mempunyai berbagai solusi untuk bertransaksi selain teller seperti mesin ATM (*automated teller machine*), *internet banking*, *sms banking*, dan *mobile banking*. Melihat kesempatan ini, Universitas Sanata Dharma bekerja sama dengan bank CIMB Niaga mengembangkan layanan web services sehingga layanan elektronik perbankan seperti mesin ATM, *internet banking*, *sms banking*, dan *mobile banking* dapat digunakan untuk pembelian formulir pendaftaran calon mahasiswa.

Tulisan ini akan membahas proses pembelian formulir dengan teknologi *web service*. Tujuan utama aplikasi ini adalah untuk memudahkan calon mahasiswa dalam membeli formulir pendaftaran di luar jam kerja teller bank dengan memanfaatkan semua *channel* layanan elektronik dari sebuah bank, dalam hal ini bank CIMB Niaga. Tulisan ini terdiri atas penjelasan tentang web service, metodologi penelitian yang merupakan metode pengembangan aplikasi, hasil pengujian serta pembahasan dan diakhiri dengan kesimpulan.

2. Web Service

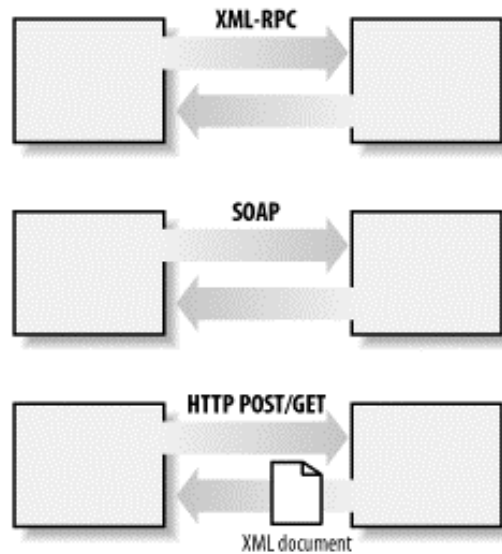
Tujuan utama aplikasi ini adalah bagaimana memanfaatkan semua layanan elektronik perbankan seperti mesin ATM, *internet banking*, *sms banking*, dan *mobile banking* untuk pembelian formulir pendaftaran calon mahasiswa sehingga calon mahasiswa dapat melakukan pembelian formulir di luar jam kerja teller bank. Solusinya adalah pemanfaatan web service.

Web service adalah sistem perangkat lunak yang dirancang untuk mendukung operasi mesin antar-mesin melalui interaksi jaringan komputer dengan aplikasi *web server*, *world wide web* (WWW) *HyperText Transfer Protocol* (HTTP) [1]. Web service memungkinkan komunikasi antar berbagai aplikasi dengan menggunakan standar terbuka seperti HTML, XML, WSDL, dan SOAP. Gambar 1 menunjukkan *web service* dapat memberikan layanan dengan sistem operasi dan bahasa pemrograman yang berbeda. Web service menggunakan XML sebagai standar pertukaran pesan dan data.



Gambar 1. *Web service* dasar [2]

Ada beberapa cara untuk pertukaran data XML. Kita bisa menggunakan XML Remote Procedure Calls (XML-RPC) atau SOAP, atau HTTP GET / POST seperti Gambar 2.

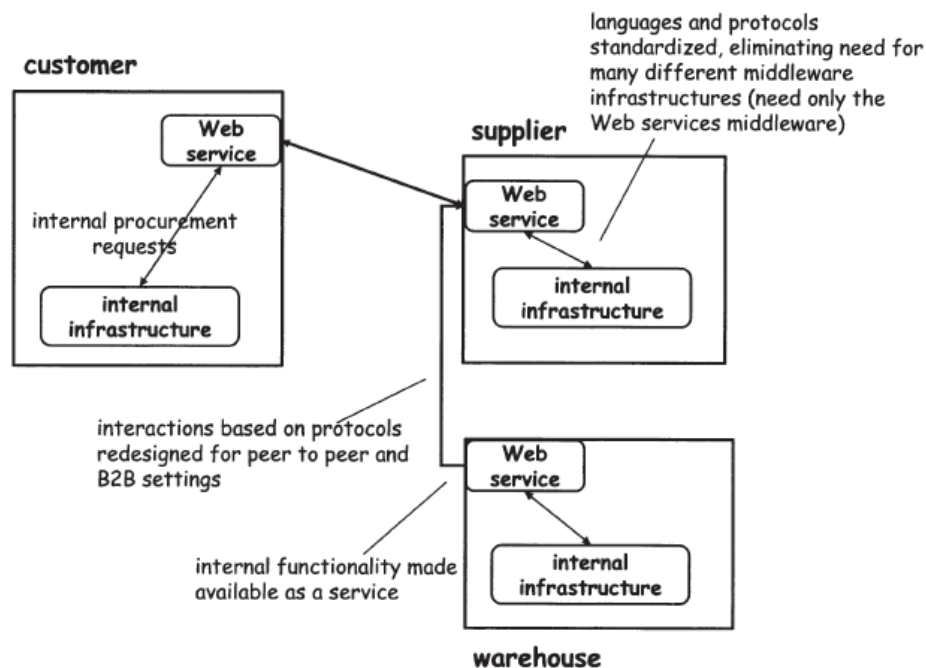


Gambar 2. Pertukaran pesan XML pada *web service*

Web service dilengkapi dengan 2 fitur/property:

1. *Web service* harus mampu menjelaskan dirinya sendiri (*self describing*). Saat sebuah *web service* diterbitkan, pengembang harus menyediakan *public interface* terkait dengan *web service* tersebut.
2. *Web service* harus mudah ditemukan (*discoverable*) oleh aplikasi pihak lain. Saat sebuah *web service* diterbitkan, *web service* tersebut harus mudah diterbitkan dan juga mudah ditemukan oleh pihak lain yang membutuhkan.

Dalam konteks dunia bisnis, *web service* digunakan untuk melakukan sinkronisasi antar entitas bisnis, misalnya kasus retail/perdagangan, untuk sinkronisasi data barang penjualan, data uang penjualan dan stok barang. Ini akan menghubungkan entitas toko, bank, dan produsen barang.



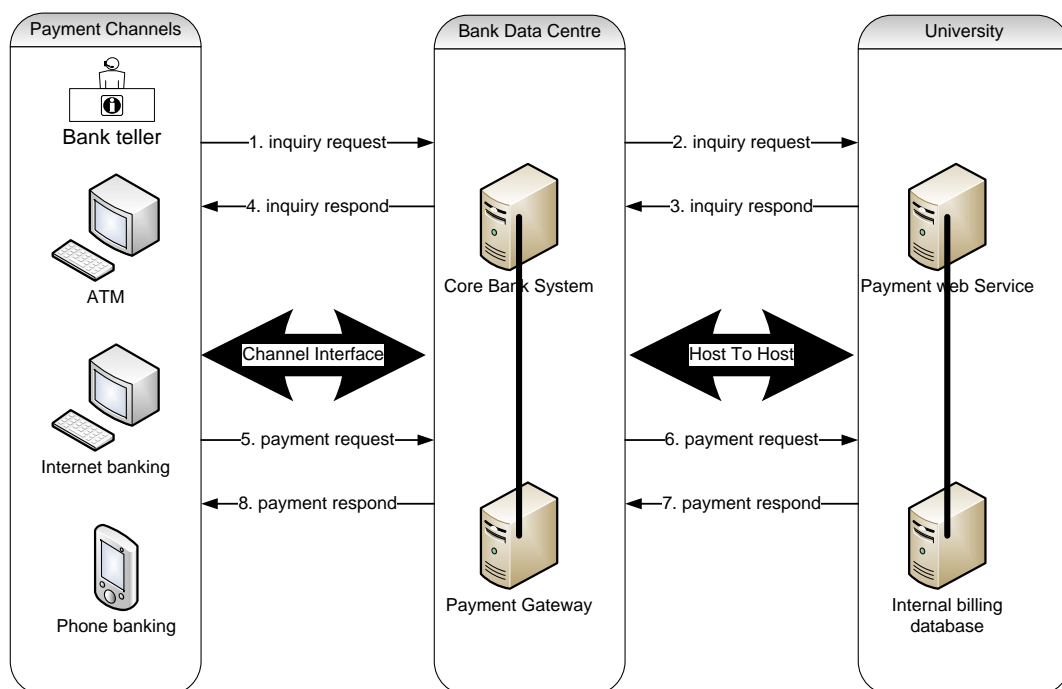
Gambar 3. *Service oriented, peer to peer protocols*, dan *standarisasi* merupakan komponen utama dari *web service*[3]

Gambar 3 menunjukkan bahwa masing-masing pihak yang terlibat menggunakan aplikasi berbasis Web sebagai titik masuk ke sistem informasi masing-masing pihak dan bersifat aplikasi internal. Interaksi antar pihak (*customer ke supplier dan supplier ke warehouse*) terjadi secara *peer-to-peer* menggunakan protocol komunikasi yang standar. Hal ini akan menyembunyikan kompleksitas pemrograman dari masing-masing pihak dengan memanfaatkan kesamaan protocol atau “bahasa” dalam pengiriman pesan dan data. Dengan kesamaan protocol, maka masing-masing pihak akan berkonsentrasi dalam membuat aplikasi internalnya masing-masing.

3. Metode Penelitian

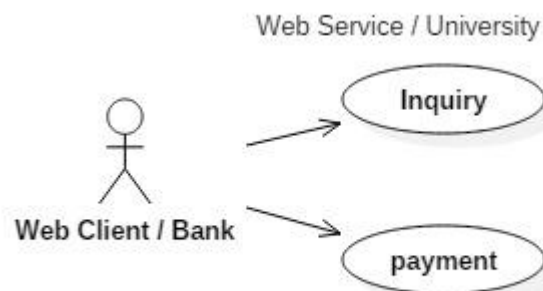
Penelitian ini dilakukan dengan membuat sebuah aplikasi yang nantinya akan diuji unjuk kerjanya berdasarkan efektifitas aplikasi tersebut terhadap proses pendaftaran mahasiswa baru di Universitas Sanata Dharma. Pengembangan aplikasi menggunakan pendekatan *Rational unified process* (RUP). Berdasarkan [4], ada 6 pekerjaan inti yaitu *Business modeling workflow, Requirements workflow, Analysis & Design workflow, Implementation workflow, Test workflow, dan Deployment workflow*. Kesemua pekerjaan ini akan didokumentasikan sesuai standar *unified modelling language* (UML) [5][6].

Berdasarkan proses kerja RUP diatas, langkah kerja awal adalah penentuan kebutuhan dari aplikasi. Berdasarkan masalah yang dihadapi oleh berbagai pihak khususnya calon mahasiswa Universitas Sanata Dharma adalah kesulitan mengakses kantor bank BRI di luar jam kantor. Hal ini disebabkan oleh karena calon mahasiswa masih berstatus pelajar yang jam sekolahnya sama dengan jam kantor bank BRI. Di sisi lain ada peluang penggunaan layanan berbasis elektronis seperti ATM, *internet banking*, dan lainnya, namun layanan tersebut masih belum bisa digunakan untuk keperluan pendaftaran calon mahasiswa baru. Berdasarkan masalah dan peluang tersebut, kebutuhan fungsional dari aplikasi yang dikembangkan adalah sebuah aplikasi yang mampu menangani transaksi pembayaran formulir calon mahasiswa baru menggunakan layanan elektronis bank (ATM, *internet banking, sms banking*). Sesuai kebutuhan fungsional di atas, teknologi *web service* dipilih untuk memenuhi kebutuhan fungsional tersebut.

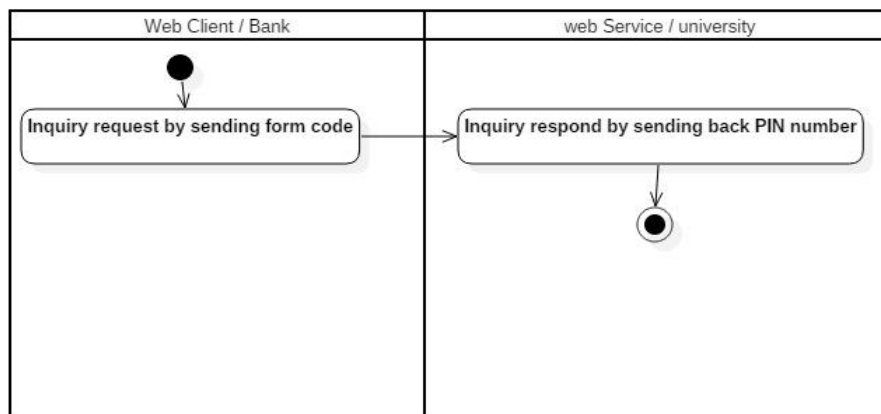


Gambar 4. Desain arsitektur host to host

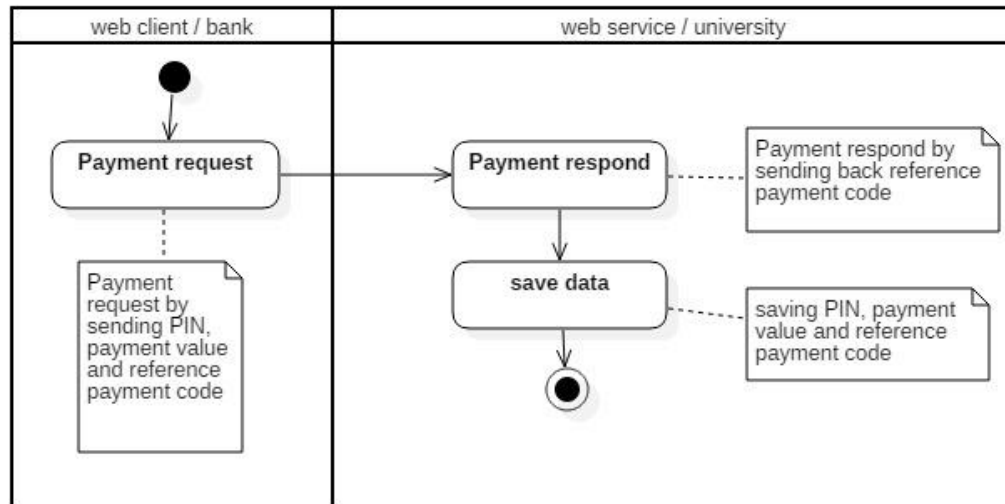
Aplikasi web service dikembangkan dalam desain besar seperti Gambar 4 yang merupakan gambaran umum aplikasi *web service* dalam konteks sistem perbankan. Web service yang dikembangkan berada di pihak penyedia ketiga dalam hal ini pihak universitas. Web service akan menyediakan data keuangan/tagihan yang diminta web client system bank. Ada 2 service penting yaitu *inquiry* dan *payment*. *Inquiry* merupakan service untuk memberikan informasi tagihan sedangkan *payment* merupakan service untuk pembayaran. Aplikasi yang dikembangkan merupakan aplikasi interaksi antara bank dengan universitas. Khusus untuk proses penjualan formulir pendaftaran mahasiswa, dijelaskan sesuai Gambar 5, Gambar 6, Gambar 7 yang merupakan diagram use case, diagram aktivitas antara web client bank dengan web service universitas. Untuk kasus penjualan formulir, ada sedikit perbedaan dengan proses pembayaran biasa. Hal ini mengakomodasi kebutuhan kebijakan “bayar dulu baru isi formulir” berakibat pada kebutuhan penyediaan kode pembayaran atau PIN pembayaran harus sudah disediakan pada saat transaksi. Hal ini dapat dilihat pada diagram aktivitas pada Gambar 6 dan Gambar 7. Gambar 6 menunjukkan proses *inquiry* dimana *web client/bank* mengirimkan permintaan *inquiry* berupa kode rekening ke *web service/universitas* yang nantinya akan dibalas/direspon. Pihak *web service/universitas* akan membalas data permintaan dengan data respon berupa kode pembayaran/PIN. Gambar 7 menunjukkan proses *payment* dimana *web client/bank* mengirimkan permintaan *payment* berupa kiriman ulang kode pembayaran/PIN beserta kode referensi pembayaran ke *web service/universitas*. Pihak *web service/universitas* akan membalas data permintaan dengan mengirim ulang kode pembayaran/PIN dan kode referensi pembayaran. Aplikasi *web service/universitas* selanjutnya akan menyimpan kode pembayaran dan kode referensi pembayaran untuk digunakan oleh aplikasi pengisian formulir pendaftaran mahasiswa baru.



Gambar 5. Diagram use case web service universitas

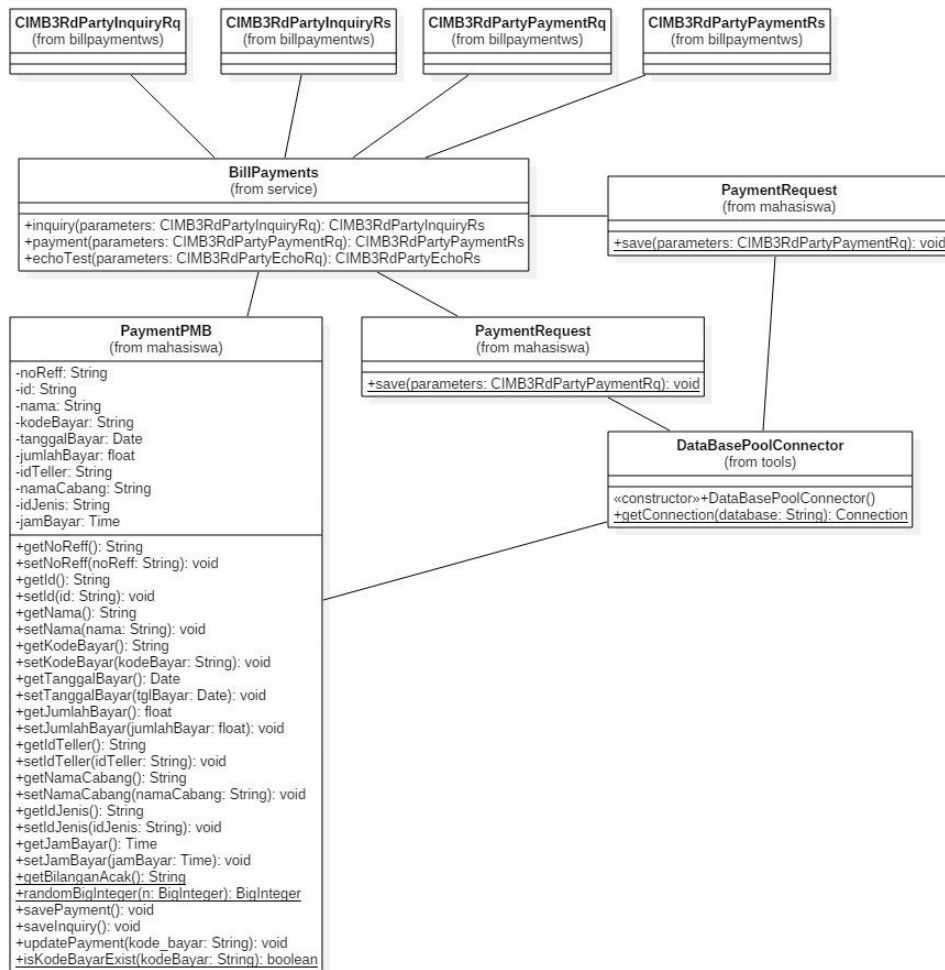


Gambar 6. Diagram aktivitas web service untuk kasus *Inquiry*



Gambar 7. Diagram aktivitas web service untuk kasus *Payment*

Diagram kelas pada gambar 8 merupakan diagram kelas implementasi web service. Pengembangan aplikasi web service mengacu pada standar yang ditetapkan oleh bank CIMB Niaga [7]. Kelas yang berawalan CIMB merupakan kelas library dari bank CIMB. Kelas yang diimplementasikan adalah kelas Bill Payments, Payment PMB, Payment Request, Payment Inquiry, dan Database Pool Connector. Kelas Bill payments merupakan kelas utama yang berisi fungsi service *Inquiry* dan *Payment* sedangkan kelas lainnya merupakan kelas entitas yang mengatur data dari table Payment PMB, Payment Request, Payment Inquiry. Khusus kelas Database Pool Connector merupakan kelas yang memberikan objek koneksi ke database MySQL. Pada masing-masing kelas entitas terdapat fungsi-fungsi yang akan melakukan perintah SQL untuk memanipulasi data pada masing-masing table.



Gambar 8. Diagram aktivitas web service untuk kasus *Payment*

Tabel 1 dan tabel 2 merupakan data XML hasil program aplikasi. Aplikasi web service akan membangkitkan data XML sesuai permintaan/request dari web client/bank. Bagian yang penting pada *inquiry respond* adalah bagian *customer Name* yang berisi PIN/kode bayar berupa 10-12 digit angka acak. PIN/kode bayar ini juga harus diteruskan oleh web client/bank pada saat mengirimkan *payment request* (tabel 2). Saat payment request, aplikasi web client/bank selain mengirimkan PIN/kode bayar, kode referensi pembayaran (*reference number transaction*) juga disertakan sebagai penanda/bukti telah terjadi penerimaan pembayaran oleh bank.

Tabel 1. XML inquiry request dan respond

Inquiry request	Inquiry respond
<pre> <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:bil="http://CIMB3rdParty/BillPaymentWS"> <soapenv:Header/> <soapenv:Body> <bil:CIMB3rdParty_InquiryRq> </pre>	<pre> <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"> <S:Body> <CIMB3rdParty_InquiryRs xmlns="http://CIMB3rdParty/BillPaymentWS"> <InquiryRs> </pre>

<pre> <!--Optional:--> <bil:InquiryRq> <!--Optional:--> <bil:TransactionID>123456789</bil:TransactionID> <!--Optional:--> <bil:ChannelID>ATM</bil:ChannelID> <!--Optional:--> <bil:TerminalID>192.168.0.1</bil:TerminalID> <!--Optional:--> <bil:TransactionDate>2018012909190000</bil:TransactionDate> <!--Optional:--> <bil:CompanyCode>5779</bil:CompanyCode> <!--Optional:--> <bil:CustomerKey1>1007</bil:CustomerKey1> <!--Optional:--> <bil:CustomerKey2></bil:CustomerKey2> <!--Optional:--> <bil:CustomerKey3></bil:CustomerKey3> <!--Optional:--> <bil:AdditionalData1></bil:AdditionalData1> <!--Optional:--> <bil:AdditionalData2></bil:AdditionalData2> <!--Optional:--> <bil:AdditionalData3></bil:AdditionalData3> <!--Optional:--> <bil:AdditionalData4></bil:AdditionalData4> </bil:InquiryRq> </bil:CIMB3rdParty_InquiryRq> </soapenv:Body> </soapenv:Envelope> </pre>	<pre> <TransactionID>123456789</TransactionID> <ChannelID>ATM</ChannelID> <TerminalID>202.94.83.49</TerminalID> <TransactionDate>2018012904280055</TransactionDate> <CompanyCode>5779</CompanyCode> <CustomerKey1>1007</CustomerKey1> <CustomerKey2/> <BillDetailList> <BillDetail> <BillCurrency>IDR</BillCurrency> <BillCode>007</BillCode> <BillAmount>10000</BillAmount> <BillReference>0072018012904290000</BillReference> </BillDetail> </BillDetailList> <Currency>IDR</Currency> <Amount>10000</Amount> <Fee>0</Fee> <PaidAmount>10000</PaidAmount> <CustomerName>143953909814 TESTPMB</CustomerName> <AdditionalData1>143953909814 TESTPMB</AdditionalData1> <AdditionalData2>Formulir PMB</AdditionalData2> <FlagPayment>1</FlagPayment> <ResponseCode>00</ResponseCode> <ResponseDescription>Success</ResponseDescription> </InquiryRs> </CIMB3rdParty_InquiryRs> </S:Body> </S:Envelope> </pre>
---	---

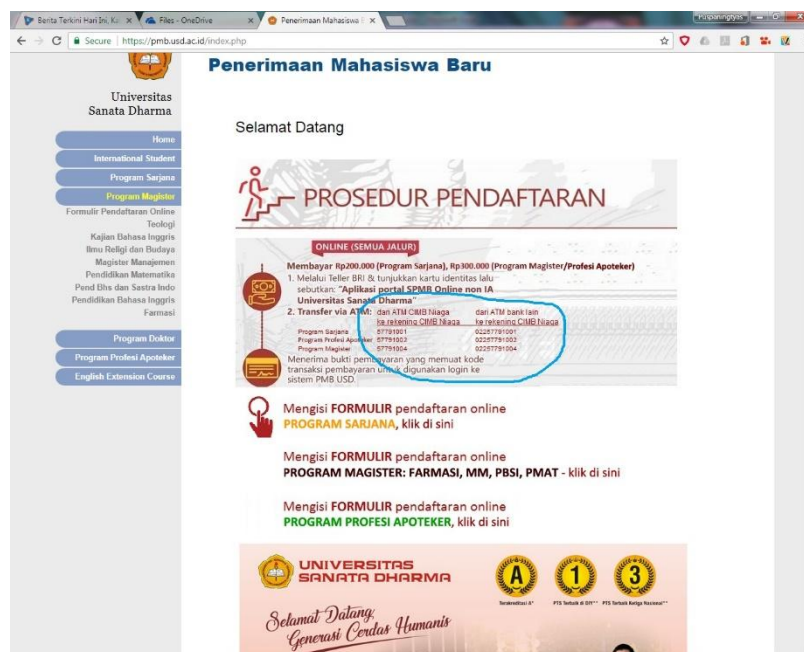
Tabel 2. XML payment request dan respond

Payment request	Payment respond
<pre> <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:bil="http://CIMB3rdParty/BillPaymentWS"> <soapenv:Header/> <soapenv:Body> <bil:CIMB3rdParty_PaymentRq> <!--Optional:--> </pre>	<pre> <S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"> <S:Body> <CIMB3rdParty_PaymentRs xmlns="http://CIMB3rdParty/BillPaymentWS"> <PaymentRs> <TransactionID>123456789</TransactionID> </pre>

<pre> <bil:PaymentRq> <!--Optional:--> <bil:TransactionID>123456789</bil:TransactionID> <!--Optional:--> <bil:ChannelID>ATM</bil:ChannelID> <!--Optional:--> <bil:TerminalID>192.168.0.1</bil:TerminalID> <!--Optional:--> <bil:TransactionDate>2018012909190000</bil:TransactionDate> <!--Optional:--> <bil:CompanyCode>5779</bil:CompanyCode> <!--Optional:--> <bil:CustomerKey1>1007</bil:CustomerKey1> <!--Optional:--> <bil:CustomerKey2></bil:CustomerKey2> <!--Optional:--> <bil:CustomerKey3></bil:CustomerKey3> <!--Optional:--> <bil:Language></bil:Language> <!--Optional:--> <bil:Currency>IDR</bil:Currency> <!--Optional:--> <bil:Amount>10000</bil:Amount> <!--Optional:--> <bil:Fee>0</bil:Fee> <!--Optional:--> <bil:PaidAmount>10000</bil:PaidAmount> <!--Optional:--> <bil:ReferenceNumberTransaction>B12345678xx</bil:ReferenceNumberTransaction> <!--Optional:--> <bil:FlagPaymentList>100000</bil:FlagPaymentList> <!--Optional:--> <bil:CustomerName>143953909814 TESTPMB</bil:CustomerName> <!--Optional:--> <bil:AdditionalData1>143953909814 TESTPMB</bil:AdditionalData1> <!--Optional:--> <bil:AdditionalData2></bil:AdditionalData2> <!--Optional:--> <bil:AdditionalData3></bil:AdditionalData3> <!--Optional:--> <bil:AdditionalData4>?</bil:AdditionalData4> </pre>	<pre> <ChannelID>ATM</ChannelID> <TerminalID>192.168.0.1</TerminalID> <TransactionDate>2018012904330029</TransactionDate> <CompanyCode>5779</CompanyCode> <CustomerKey1>1007</CustomerKey1> <PaymentFlag>1</PaymentFlag> <CustomerName>143953909814 TESTPMB</CustomerName> <Currency>IDR</Currency> <Amount>10000</Amount> <Fee>0</Fee> <PaidAmount>10000</PaidAmount> <ReferenceNumberTransaction>B12345678xx</ReferenceNumberTransaction> <AdditionalData1>Universitas Sanata Dharma</AdditionalData1> <AdditionalData2/> <ResponseCode>00</ResponseCode> <ResponseDescription>Transaction Success</ResponseDescription> </PaymentRs> </CIMB3rdParty_PaymentRs> </S:Body> </S:Envelope> </pre>
---	--

<pre> </bil:PaymentRq> </bil:CIMB3rdParty_PaymentRq> </soapenv:Body> </soapenv:Envelope> </pre>	
---	--

Implementasi *web service* untuk pembelian formulir pendaftaran mahasiswa baru Universitas Sanata Dharma dapat dilihat pada gambar 9, dimana halaman ini bisa diakses di <http://pmb.usd.ac.id>. Calon mahasiswa dapat membeli formulir PMB dengan melakukan transaksi di ATM bank CIMB Niaga atau ATM bank lainnya. Layanan *web service* memungkinkan semua model transaksi elektronik dari bank CIMB Niaga.



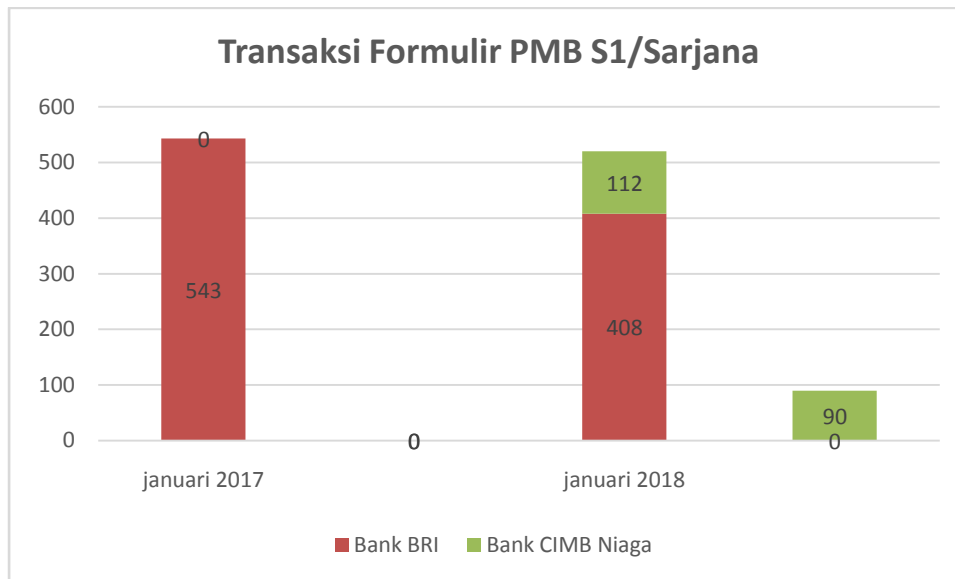
Gambar 9. Tampilan informasi pembayaran via ATM dengan rekening penampung di bank CIMB Niaga.

4. Hasil dan Pembahasan

Aplikasi pembayaran formulir berbasis *web service* dijalankan mulai Oktober 2017. Efektivitas aplikasi ini akan diukur berdasarkan jumlah transaksi yang terjadi sebelum dan sesudah aplikasi ini dijalankan. Penggunaan data transaksi formulir mahasiswa baru dapat menunjukkan perubahan perilaku calon mahasiswa dalam cara mendapatkan formulir.

Tabel 3. Jumlah transaksi formulir PMB program S1/Sarjana

Nama Bank	Januari 2017			Januari 2018		
	jam 08-16	non jam 08-16	sub total	jam 08-16	non jam 08-16	sub total
Bank BRI	543	0	543	408	0	408
Bank CIMB Niaga	0	0	0	112	90	202
Total	543	0	543	520	90	610



Gambar 10. Transaksi formulir PMB bulan Januari 2017 dan 2018.

Aplikasi web service mulai diimplementasikan oleh Universitas Sanata Dharma mulai bulan Oktober 2017 untuk pendaftaran mahasiswa baru tahun 2018. Data transaksi formulir PMB dapat dilihat pada tabel 3 dan gambar 10. Dari data tersebut terlihat bahwa terjadi kenaikan jumlah transaksi tahun 2018 sebesar 67 formulir atau meningkat sebesar 12% dibanding tahun 2017. Cukup sulit dipastikan bahwa kenaikan ini akibat penerapan web service dan ini perlu ada penelitian lebih mendalam kepada calon mahasiswa. Kenaikan pembelian formulir bisa saja terjadi karena penguatan branding dari universitas dan mungkin juga karena kemudahan dalam pembelian formulir. Hal ini perlu penelitian lebih lanjut. Dari data tersebut juga terlihat bahwa sekarang porsi transaksi dari bank BRI sebesar 408/610 atau sekitar 69% sedangkan sisanya sebesar 31% merupakan transaksi dari bank CIMB Niaga. Hal ini menunjukkan bahwa sekitar 31% transaksi sudah menggunakan layanan elektronis, bukan dari teller. Hal ini menunjukkan bahwa calon mahasiswa USD sudah cukup baik pengetahuannya tentang layanan elektronis perbankan. Pengetahuan masyarakat tentang layanan elektronis perbankan sudah baik karena layanan ini masih sangat baru, masih sekitar 3 bulan berjalan, namun penggunaannya sudah mencapai 30% transaksi. Dari data juga terlihat dari 220 transaksi di bank CIMB Niaga, ada 90 transaksi atau sekitar 80% transaksi dilakukan di luar jam 08-16, atau di luar jam kerja. Hal ini menunjukkan bahwa layanan web service sangat membantu calon mahasiswa yang kesulitan untuk bertransaksi di jam kerja dan juga di kantor cabang bank.

5. Kesimpulan

Dari data di atas dapat ditunjukkan manfaat yang sangat baik oleh layanan web service perbankan di luar jam kerja. Harapan manfaat besar juga terlihat dari proporsi layanan web service yang bersifat elektronis dibanding dengan layanan teller. Hal ini juga menunjukkan bahwa calon mahasiswa atau lulusan sekolah menengah atas/kejuruan sudah tinggi pengetahuannya tentang layanan perbankan. Sudah saatnya pihak-pihak yang berhubungan dengan perbankan dapat memanfaatkan tingkat pengetahuan masyarakat yang tinggi akan layanan elektronis perbankan untuk meningkatkan layanan mereka, termasuk universitas.

Daftar Pustaka

- [1] D. Booth, H. Haas, F. McCabe, E. Newcomer, M. Champion, C. Ferris, and D. Orchard, "Web Services Architecture," *W3C Note*, vol. 22, no. February, pp. 1–98, 2004.
- [2] E. Cerami, *Web services essentials*. 2002.

-
- [3] G. Alonso, F. Casati, H. Kuno, and V. Machiraju, "Web Services," *Web Serv. SE - 5*, no. Chapter 1, pp. 123–149, 2004.
 - [4] Paper Rational Software White, "Rational Unified Process Best Practices for Software," *Development*, pp. 1–21, 2004.
 - [5] J. Rumbaugh, I. Jacobson, and G. Booch, *The UML reference manual*, vol. 1. 1999.
 - [6] T. C. Lethbridge and R. Laganieri, "Object-Oriented Software Engineering : Practical Software Development Using Uml and Java," *McGraw-Hill Publ. Co.*, p. 561, 2004.
 - [7] CIMB Niaga Bank, "CIMB HOST TO HOST WSDL STANDARD," 2016.