

---

## HOST TO HOST PAYMENT SYSTEM PADA POLITEKNIK ATI PADANG

**Erien Nada Azandra**

STMIK Jaya Nusa

E-mail : erien\_jn@yahoo.co.id

Submitted: 16-08-2017, Reviewed: 17-08- 2017, Accepted 22-08-2017

<http://doi.org/10.22216/jsi.v3i1.2363>

---

### **Abstract**

*Polytechnic ATI Padang is a public university in the city of Padang, formerly known as Padang Industrial Technology Academy. In 2015 renamed the Polytechnic ATI Padang. Currently Polytechnic ATI Padang has established cooperation with Bank BNI for payment of student tuition. The tuition fee data will be a reference for student re-enrollment so students can fill out the KRS (Study Plan Card) online through their respective portals accounts. The existing constraints have not been integrated in the payment of student money at Bank BNI with academic information system so that the student registration process must still be done manually by the finance department by bringing proof of payment provided by the Bank. One solution that can be used is to use a web service to integrate school fee payment system with academic information system so that student registration process can be done easily. With this system the student enrollment process is done automatically by the system without having to report to the finance department.*

**Keywords :** *registrasion, student, system informasi, web service*

### **Abstrak**

*Politeknik ATI Padang adalah universitas negeri di kota Padang yang sebelumnya dikenal dengan Akademi Teknologi Industri Padang. Pada tahun 2015 berganti nama menjadi Politeknik ATI Padang. Saat ini Politeknik ATI Padang telah menjalin kerja sama dengan Bank BNI untuk pembayaran uang kuliahmahasiswa. Data pembayaran uang kuliah ini akan menjadi rujukan untuk pendaftaran kembali mahasiswa sehingga mahasiswa dapat mengisi KRS (Kartu Rencana Studi) secara online melalui portal masing-masing. Kendala yang ada saat ini belum terintegrasi pembayaran uang kuliahmahasiswa di Bank BNI dengan sistem informasi akademik sehingga proses registrasimahasiswa masih harus dilakukan secara manual oleh departemen keuangan dengan membawa bukti pembayaran yang diberikan oleh Bank. Salah satu solusi yang bisa digunakan adalah menggunakan layanan web untuk mengintegrasikan sistem pembayaran uang kuliah dengan sistem informasi akademik sehingga proses registrasimahasiswa dapat dilakukan dengan mudah. Dengan sistem ini proses registrasimahasiswa dilakukan secara otomatis oleh sistem tanpa harus melapor ke departemen keuangan.*

**Katakunci:** *mahasiswa, registrasi, sistem informasi, web service*

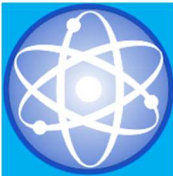
---

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Politeknik ATI Padang saat ini sudah menggunakan system informasi akademik berbasis web untuk menunjang kegiatan akademiknya. Salah satu sistem informasi yang ada adalah Portal Mahasiswa yang digunakan untuk pengisian Kartu Rencana

Studi (KRS). Salah satu syarat untuk melakukan pengisian KRS adalah mahasiswa harus melakukan registrasi ulang dengan membayar uang kuliah ke Bank yang telah ditunjuk oleh pihak kampus. Bukti pembayaran uang kuliah tersebutlah yang digunakan oleh bagian keuangan untuk registrasi ulang. Setelah registrasi ulang baru



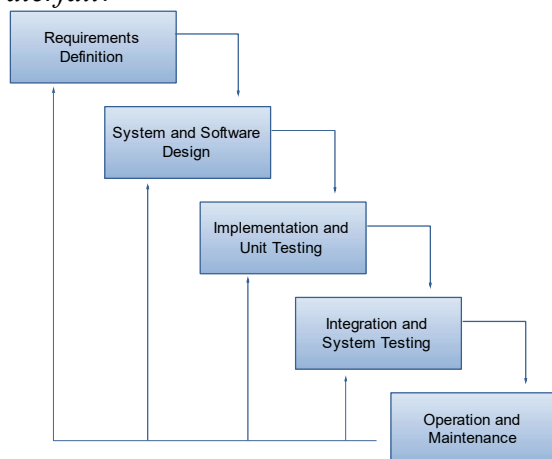
mahasiswa bisa melakukan pengisian KRS melalui halaman portal mahasiswa dengan menggunakan user dan password yang sudah dimiliki.

Salah satu kendala yang dihadapi saat ini adalah ketika mahasiswa harus menyerahkan bukti pembayaran untuk registrasi ulang ke bagian keuangan setelah melakukan pembayaran uang kuliah, bagi mahasiswa yang berdomisili dekat dengan kampus atau yang masih ada di kota Padang, mungkin tidak terlalu menjadi masalah. Tapi bagi mahasiswa yang berdomisili diluar kota Padang ataupun di luar Provinsi Sumatera Barat, maka harus datang ke kampus hanya untuk melakukan registrasi ulang, padahal pembayaran uang kuliah sudah mereka dilakukan di daerahnya masing-masing. Tentu saja hal ini memberatkan dan menyulitkan bagi mahasiswa yang berasal dari luar kota Padang baik dari segi biaya, tenaga maupun waktu yang digunakan.

## 1.2 Metode Penelitian

Model *waterfall* adalah proses pengembangan perangkat lunak tradisional yang umum digunakan dalam proyek-proyek perangkat lunak. Hal ini disebut *waterfall* karena proses mengalir secara sistematis dari satu tahap ke tahap lainnya dalam mode ke bawah. Membentuk kerangka kerja untuk pengembangan perangkat lunak.[1]

Gambar berikut adalah bagan dari model *waterfall*:



Gambar 1. *SDLCwaterfall model*

Gambar di atas adalah tahapan umum dari model proses ini. Penjelasan dari tahap-tahap *waterfall model* adalah sebagai berikut :[2]

### 1. *Requirements definition*

Pada tahap ini, proses pencarian kebutuhan diintensifkan dan difokuskan pada sistem *software*. Kebutuhan dari sistem dan *software* ditunjukkan kepada pelanggan dalam hal ini adalah pihak Poltek ATI.

### 2. *System And Software Design*

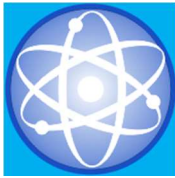
Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk *blueprint software* sebelum *coding* dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya.

### 3. *Implementation And Unit Testing.*

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap desain yang secara teknis nantinya dikerjakan oleh *programmer*. Pada tahap inilah proses pembuatan *web service* yang mengintegrasikan sistem pembayaran uang kuliah dengan system informasi akademik Poltek ATI. *Web service* yang dibangun adalah dari sisi servernya.

### 4. *Integration And Sytem Testing.*

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari *error* dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya. Pada tahap ini proses uji



coba dilakukan terlebih dahulu pada 1 komputer saja(*localhost*) sampai tidak ada lagi *error* atau kesalahan yang ditemukan pada sistem. Untuk menguji apakah *web service* servernya dapat berjalan dengan baik maka digunakan *software* SOAPUI.

#### 5. Operation And Maintenance

Pada tahap ini sistem langsung dioperasikan pada komputer server yang sudah disediakan dan kembali melakukan pengujian dengan data yang sebenarnya. Pada tahap ini, proses perawatan/pemeliharaan terhadap sistem tentu sangat dibutuhkan agar sistem dapat berjalan dengan baik dan sesuai dengan kebutuhan.

## LANDASAN TEORI

### 1. Web Service

Menurut Snel (2001) *web service* didefinisikan sebagai *interface* akses jaringan pada fungsi aplikasi, yang dibangun menggunakan teknologi standar yang digunakan di internet. Dengan kata lain jika suatu aplikasi diakses menggunakan protokol HTTP(*Hypertext Transfer Protocol*), XML(*Extensible Markup Language*), SMTP(*Simple Mail Transfer Protocol*) atau Jabber, maka bisa dikatakan itu adalah *web service*.

Menurut IBM (2001) *web service* adalah aplikasi modular, terdistribusi dan dinamis yang dapat dideskripsikan, di-*publish* dan diletakkan di jaringan untuk menciptakan produk, proses dan *supply chain*. Aplikasi *web service* dapat bersifat lokal, terdistribusi atau berbasis web

Menurut W3C (2009) *web service* adalah sebuah *software* aplikasi yang dapat teridentifikasi oleh URI(*Uniform Resource Identifier*) dan memiliki *interface* yang didefinisikan, dideskripsikan dan dimengerti oleh XML dan juga mendukung interaksi langsung dengan *software* aplikasi yang lain

dengan menggunakan *message* berbasis XML melalui protokol internet.

Menurut Suda (2003) kelebihan *web service* antara lain *interoperability*, *reuseable*, *platform independent*, *strandarized pttocol*, *XML based*, *plaint text* dan sebagainya.[3]

Web services dibangun atas 5 (lima) dasar standar :

1. XML adalah format data standar yang digunakan dalam proses pertukaran data.
2. HTTP(S) adalah protokol pertukaran data berbasis teks yang saat ini telah menjadi protokol standar pengaksesan internet melalui browser dan media lainnya.
3. WSDL: *Web Services Definition Language* adalah standar *web service* yang menyediakan aturan fungsi dan object.
4. SOAP: *Simple Object Application Protocol* yaitu protokol yang mengemas fungsi dan object pada saat proses pertukaran data.
5. UDDI: *Universal Description, Discovery and Integration* yaitu penampung *registry web services* yang telah dikembangkan agar dapat diakses dan diintegrasikan.[4]

Beberapa alasan mengapa digunakannya *web service* adalah sebagai berikut:

1. *Web service* dapat digunakan untuk mentransformasikan satu atau beberapa bisnis *logic* atau *class* dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
2. *Web service* memiliki kemudahan dalam proses *deployment*-nya, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. *Web service* cukup di-upload ke web server dan siap diakses oleh pihak-pihak yang telah diberikan otorisasi.
3. *Web service* berjalan di port 80 yang merupakan protokol standar HTTP,



dengan demikian web service tidak memerlukan konfigurasi khusus di sisi *firewall*. [5]

## 2. Host to Host

Host to Host atau sering dikenal H2H adalah suatu sistem antar server yang terhubung yang terhubung secara langsung satu dengan yang lainnya secara *real time*. Dengan kata lain suatu komunikasi atau hubungan dalam suatu jaringan yang terjadi antar host, yaitu komputer dengan komputer ataupun komputer dengan perangkat lain yang terhubung stau dengan yang lain. Host to Host berasal dari kata Host, kata Host dalam bahasa Inggris yang artinya tuan rumah. Kata Host dalam bidang teknologi dapat diartikan sebagai istilah dalam jaringan komputer yang digunakan untuk menunjuk sebuah komputer yang terhubung ke jaringan internet. Berdasarkan pengertian tersebut maka Host to Host adalah Host (sebuah sistem dalam server) yang terhubung dengan Host (sebuah sistem dalam server yang lain) secara langsung dan *real time*. [6]

## 3. PHP (*Hypertext Preprocessor*)

PHP adalah bahasa server-side scripting yang menyatu dengan HTML (*Hypertext Markup Language*) untuk membuat halaman web yang dinamis. Maksud dari server-side *scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya akan dijalankan dalam sistem. [7]

## 4. Database/basis data

Basis data adalah dua atau lebih simpanan data dengan elemen-elemen data penghubung, yang dapat diakses lebih dari satu cara. Basis data dinyatakan dengan teknik-teknik formal dan manajemen basis data. Dari definisi diatas, maka dapat dikatakan bahwa basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya. [8]

## 5. MySQL

MySQL (bisa dibaca dengan mai-es-ki-el atau bisa juga mai-se-kuel) adalah suatu perangkat lunak database relasi (*Relational Database Management System* atau DBMS), seperti halnya ORACLE, POSTGRESQL, MSSQL dan sebagainya. SQL merupakan singkatan dari *Structure Query Language*, didefinisikan sebagai suatu sintaks perintah-perintah tertentu atau bahasa program yang digunakan untuk mengelola suatu database. Jadi MySQL adalah *software*-nya dan SQL adalah bahasa perintahnya. [9]

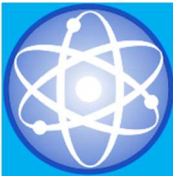
## 6. Trigger

*Trigger* pada MySQL merupakan objek dalam database yang berhubungan dnegan tabel, trigger akan aktif ketika terjadi sesuatu dengan tabel yang dihubungkannya. *Trigger* biasanya digunakan untuk melakukan pemeriksaan nilai-nilai yang masuk ke tabel atau untuk melakukan perhitungan pada nilainilai (*record*) baru yang terlibat dalam proses update. [10]

## HASIL DAN PEMBAHASAN

Dari hasil pengamatan/penelitian yang telah dilakukan, pada proses registrasi ulang mahasiswa, terdapat 3 entiti yang terlibat, yaitu Mahasiswa, Keuangan dan Bank.

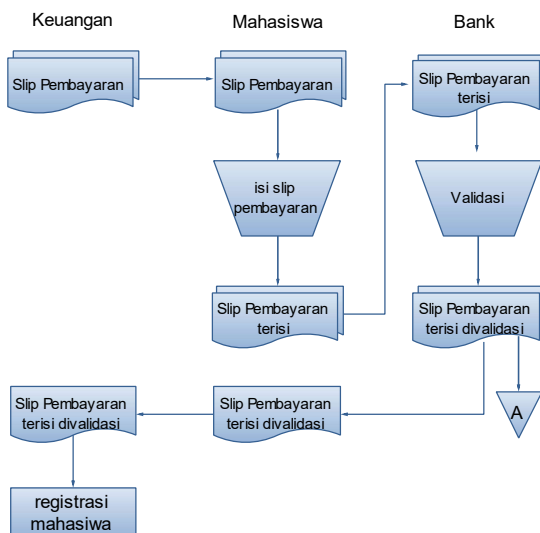
- Mahasiswa menerima slip pembayaran dari bagian keuangan yang terdiri dari 2 rangkap
- Mahasiswa mengisi biodata pada slip pembayaran yang telah diberikan oleh bagian keuangan
- Mahasiswa melakukan pembayaran uang kuliah ke bank dengan membawa slip pembayaran yang telah diisi.
- Bank melakukan validasi slip pembayaran, 1 rangkap diarsipkan oleh bank, 1 rangkap lagi dikembalikan ke mahasiswa
- Mahasiswa menyerahkan slip pembayaran yang telah divalidasi ke



bagian keuangan untuk proses registrasi ulang.

- f. Bagian keuangan mengentrikan registrasi mahasiswa. Pada proses registrasi mahasiswa, data akan tersimpan di dalam database portal akademik pada tabel registrasi. Setelah data mahasiswa diregistrasi barulah mahasiswa bisa melakukan pengisian KRS secara online dengan menggunakan akun portal yang sudah dimiliki.

Proses bisnis registrasi mahasiswa dapat dilihat pada gambar berikut :

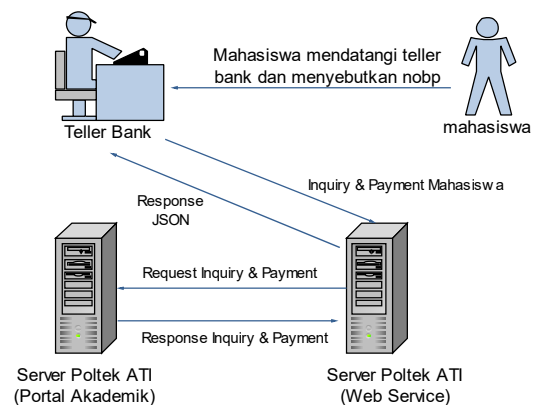


Gambar 2. Proses bisnis registrasi Mahasiswa yang sedang berjalan

Dari uraian di atas, ditemukan adanya kelemahan yaitu mahasiswa yang telah melakukan pembayaran uang kuliah harus menyerahkan kembali slip pembayaran yang telah divalidasi ke bagian keuangan untuk proses registrasi mahasiswa. Bagi mahasiswa yang berdomisili di Padang mungkin hal ini tidak terlalu menjadi masalah, tetapi akan berbeda halnya dengan mahasiswa yang berasal dari luar kota Padang atau yang di luar provinsi Sumatera Barat, hal ini akan menyita waktu dan biaya tambahan karena harus datang hanya untuk menyerahkan slip pembayaran tersebut.

Untuk mengatasi permasalahan yang ditemukan, maka dikembangkan sistem yang sudah ada dengan sistem *Host to host* yang memanfaatkan teknologi *web service* pada proses registrasi mahasiswa. Dengan sistem yang dikembangkan ini mahasiswa secara otomatis akan teregistrasi ketika melakukan pembayaran uang kuliah di Bank. Untuk proses registrasi mahasiswa secara otomatis ketika melakukan pembayaran uang kuliah, maka pada sistem yang dibangun digunakan fasilitas Trigger yang ada pada MySQL.

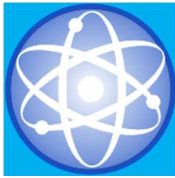
Adapun *work flow* sistemnya dapat dilihat pada gambar berikut :



Gambar 3. Work Flow Host to Host

Berikut adalah penjelasan dari gambar 3 di atas :

- a. Mahasiswa mendatangi *teller* bank dan menyebutkan nobp dan jenis pembayaran “uang kuliah”.
- b. *Teller* bank akan melakukan proses *inquiry* data mahasiswa. Ketika proses *inquiry* oleh bank, maka server *web service* akan melakukan pengecekan(*request*) data ke database portal akademik.
- c. Apabila data mahasiswa ditemukan, maka akan muncul tagihan uang kuliah mahasiswa. Jika data mahasiswa tidak ditemukan maka akan muncul notifikasi data tidak ditemukan.



- d. Mahasiswa menyerahkan uang pembayaran sesuai dengan tagihan yang disebutkan oleh *teller* bank.
- e. *Teller* bank melakukan *Payment*. Saat melakukan *payment*, server *web service* akan meminta *request* lagi ke database portal akademik dan secara otomatis mahasiswa akan langsung teregistrasi.

## 1. Perancangan WSDL

Layanan komunikasi Host 2 host dengan bank yang selanjutnya dirancang ke dalam format WSDL adalah :

1. *get\_tagihssp*
  - a. *get\_tagihsspRequest* : berisi parameter yang harus dapat membedakan mana mahasiswa yang sudah membayar tagihan uang kuliah dan yang belum.
  - b. *get\_tagihResponse* : berisi output berdasarkan hasil *get\_tagihssp*. Output yang ditampilkan berupa nobp, nama mahasiswa, prodi, periode pembayaran, jumlah tagihan dan keterangan.
2. *trans\_bayar*
  - a. *trans\_bayarRequest* : berisi parameter yang meminta layanan pembayaran uang kuliah
  - b. *trans\_bayarResponse* : menampilkan output yang berisi status pembayaran
3. *trans\_reversal*
  - a. *trans\_reversalRequest* : berisi parameter yang meminta layanan pembatalan pembayaran uang kuliah
  - b. *trans\_reversalResponse* : menampilkan output yang berisi status pembatalan transaksi pembayaran

Perancangan *web service* untuk layanan host 2 host ini hanyalah *web service* server, sementara untuk sisi *client*-nya disediakan oleh pihak bank. Untuk perancangan *web service*-nya yang digunakan adalah NUSOAP.

## 2. Kode PHP WSDL

Berikut adalah kode php untuk perancangan WSDL yang berisi *method* *get\_tagihssp*, *trans\_bayar* dan *trans\_reversal*

```
1 <?php
2 include("lib/funct.php");
3 require_once('lib/nusoap.php');
4 $server=new nusoap_server();
5 $server->configureWSDL('wsdlpoltekati','urn:wsdlpoltekati.wsdl');
6 $server->register('get_tagihssp',
7 array('cliid'=>'xsd:string','clipass'=>'xsd:string','nobp'=>'xsd:string'),
8 array('hasil'=>'xsd:string')
9 );
10 $server->register('trans_bayar',
11 array('cliid'=>'xsd:string','clipass'=>'xsd:string','tagihid'=>'xsd:string',
12 'jumlah_bayar'=>'xsd:string','tanggal_bayar'=>'xsd:string','ref_kode'=>'xsd:string',
13 'ket'=>'xsd:string'),
14 array('hasil'=>'xsd:string')
15 );
16 $server->register('trans_reversal',
17 array('cliid'=>'xsd:string','clipass'=>'xsd:string','ref_kode'=>'xsd:string'),
18 array('hasil'=>'xsd:string')
19 );
20 $httprawpostdata=isset($_HTTP_RAW_POST_DATA) ? $_HTTP_RAW_POST_DATA : '';
21 $server->service($httprawpostdata);
22 ?>
```

Gambar 4. Kode php WSDL

## 3. Fungsi-fungsi pada sistem H2H

Dalam sistem H2H yang dibangun, terdapat beberapa *functionphp* yang disesuaikan dengan kebutuhan sistem. Fungsi-fungsi tersebut disimpan dalam sebuah file dengan nama *funct.php*. Berikut adalah fungsi-fungsi program tersebut :

- a. *Function get\_tagihssp*: yaitu fungsi yang berfungsi untuk membaca data mahasiswa berdasarkan nobp ke dalam database. Jika data mahasiswa ditemukan dan belum melakukan pembayaran uang kuliah maka akan menampilkan data mahasiswa beserta jumlah tagihan uang kuliah. Jika mahasiswa sudah melakukan pembayaran uang kuliah maka akan menampilkan keterangan uang kuliah sudah dibayar. Tapi jika data mahasiswa tidak ditemukan maka akan menampilkan pesan data mahasiswa tidak ditemukan
- b. *Function trans\_bayar* : yaitu sebuah fungsi untuk melakukan penyimpanan



data pembayaran mahasiswa ke dalam basis data.

- c. *Function trans\_reversal* : yaitu sebuah fungsi untuk membatalkan transaksi pembayaran uang kuliah yang telah dilakukan oleh mahasiswa.

#### 4. Perancangan basis data

Setelah melakukan perancangan *web service*, maka tahap berikutnya adalah perancangan basis data untuk server Host 2 host. Untuk perancangan basis datanya DBMS yang digunakan adalah MySQL. Basis datanya terdiri dari 10 tabel yaitu : *a\_client*, *a\_bank*, *a\_periode*, *a\_ref\_aktif*, *a\_ref\_jenis*, *a\_ref\_kode*, *a\_ref\_kode\_detil*, *a\_tagih* dan *a\_ref\_rekening*. Berikut ini contoh salah satu tabel yang ada dalam basisdata tersebut :

Tabel 1. Desain tabel *a\_tagih*

| Field             | Type         | Keterangan |
|-------------------|--------------|------------|
| tagihid           | int(11)      | PK         |
| tagihnim          | varchar(50)  |            |
| tagihperid        | varchar(100) |            |
| tagihkodedetilid  | varchar(225) |            |
| tagihtanggalbayar | varchar(10)  |            |
| tagihrefkode      | int(11)      |            |
| tagihket          | varchar(255) |            |
| tagihaktif        | int(1)       |            |
| tagihcatatan      | longtext     |            |
| tagihorder        | enum         |            |
| tagihpass         | varchar(50)  |            |
| tagihvaurut       | decimal      |            |
| tagihvakode       | bigint       |            |
| tagihvaid         | varchar(25)  |            |
| tagihadduser      | varchar(50)  |            |
| tagihaddtime      | datetime     |            |
| tagihupduser      | varchar(50)  |            |
| tagihupdtype      | datetime     |            |

#### 5. Pembuatan Trigger

Pembuatan trigger ini berfungsi untuk mengintegrasikan sistem H2H dengan database portal akademik. Sehingga setiap mahasiswa yang melakukan pembayaran maka langsung teregistrasi secara otomatis. Gambar berikut menjelaskan tentang *sintax* untuk pembuatan trigger.

```
DELIMITER $$
CREATE
/*[DEFINER = { user | CURRENT_USER }]*/
TRIGGER `h2h`.`t_ai_a_trans` AFTER INSERT
ON `h2h`.`a_trans`
FOR EACH ROW BEGIN
SET @nim=(SELECT tagihnim FROM a_trans JOIN a_tagih ON transbayarid=tagihid
WHERE transid=new.transid);
SET @perid=(SELECT periode FROM a_periode WHERE perkodeid='1001' AND peraktif=1);
SET @jmlbayar=(SELECT transjmlbayar FROM a_trans WHERE transid=new.transid);
SET @tglbayar=(SELECT transanggalbayar FROM a_trans WHERE transid=new.transid);
INSERT IGNORE INTO sia_poltek_ati.`registrasi` (regmsnomp, regsem, regjmlbayar,
reganggalbayar, reguserinput, regtanggalinsert) VALUES (@nim, @perid, @jmlbayar,
@tglbayar, 'Peller Bank', NOW());
END$$
DELIMITER ;
```

Gambar 5. Trigger

#### 6. Pengujian dengan SoapUI

Untuk menguji apakah system(*web service server*) ini bisa digunakan atau tidak cukup dengan menggunakan aplikasi jadi yang sudah ada yaitu SoapUI.[11]

Dari hasil pengujian terhadap semua fungsi yang ada dengan aplikasi SoapUI, maka dapat disimpulkan bahwa semua *method* yang ada pada web service servernya dapat berjalan dengan baik.

#### 7. Instalasi sistem

Pada tahap ini yang dilakukan pertama sekali adalah instalasi server, sistem operasi yang digunakan adalah Linux Debian, berikutnya adalah instalasi aplikasi pendukung lainnya yaitu Apache dan MySQL. Setelah proses instalasi dan konfigurasi dilakukan baru mengupload sitem H2H ke komputer server.

#### 8. Pengujian

Pada tahap ini dilakukan pengujian secara berulang. Sampai tidak ditemukannya lagi bug/error pada sistem. Pengujian dilakukan dengan menggunakan data yang sebenarnya bukan data yang bersifat fiktif. Dari hasil pengujian yang dilakukan bahwa sistem H2H yang dibangun dapat berjalan sesuai dengan kebutuhan yang ada.

#### KESIMPULAN

Sistem pembayaran ini dapat diintegrasikan dengan *system* transaksi Bank (host 2 host). Sistem ini dapat meningkatkan pelayanan terhadap mahasiswa, karena



dengan adanya sistem yang terintegrasi ini maka secara otomatis mahasiswa yang melakukan pembayaran uang kuliah akan teregistrasi sehingga mahasiswa tidak perlu lagi harus menyerahkan bukti pembayaran ke bagian keuangan.

Dengan adanya teknologi *web service* ini, maka sistem juga dapat dikembangkan sesuai dengan kebutuhan pada perguruan tinggi POLTEK ATI.

#### DAFTAR PUSTAKA

- [1] I. Fahrurrozi and A. SN, "Proses Pemodelan Software Dengan Metode Waterfall Dan Extreme Programming: Studi Perbandingan," 2012.
- [2] R. A. Pascapraharastyan, A. Supriyanto, and P. Sudarmaningtyas, "Rancang Bangun Sistem Informasi Manajemen Arsip Rumah Sakit Bedah Surabaya Berbasis Web," *JSIKA*, vol. 3, no. 1, 2014.
- [3] A. B. C. Dano, H. F. Wowor, and O. A. Lantang, "Perancangan Web Service Sistem Autentikasi dan Identifikasi Berbasis QR Code Pada Universitas Sam Ratulangi," *E-Journal Tek. Elektro dan Komput.*, 2015.
- [4] A. Firmansyah, "Sistem Transaksi Perbankan di Perguruan Tinggi Studi Kasus : Universitas Padjadjaran," 2011.
- [5] A. A. Slameto, "Data Manajemen Dan Teknologi Informasi," in *Jurnal Ilmiah DASI*, 2015, p. 20.
- [6] E. Purwanto, "Integrasi Sistem Pembayaran Mahasiswa Politeknik Indonusa Surakarta Dengan Bank Rakyat Indonesia (BRI) Syariah Dengan Host To Host System," *J. Inf.*, vol. 3, no. 2, 2016.
- [7] M. R. R. Widodo, M. R. Zainuddin, and L. S. Nusantara, "Sistem Informasi Dan Pengolahan Data Kursus Mobil Berbasis Web Dengan SMS Gateway Di Armada Pasuruan," *J. Inform. Merdeka Pasuruan*, vol. 1, no. 3, 2016.
- [8] A. H. Rangkuti and A. Iskandar, "Perancangan Sistem Informasi Penjualan Tunai Pada PT Klaten Bercahaya," *J. Basis Data, ICT Res. Cent. UNAS*, vol. 3, no. 2, 2008.
- [9] Anisya, "Aplikasi Sistem Database Rumah Sakit Terpusat Pada Rumah Sakit Umum (RSU) 'Aisyiyah Padang Dengan Menerapkan Open Source (Php – MySQL)," *Momentun*, vol. 15, no. 2, 2013.
- [10] P. S. Arthajaya, A. Muliantara, and N. A. ER Sanjaya, "Implementasi GAMMU Sebagai Mesin SMS Gateway di iPhone Bali," *J. Elektron. Ilmu Komput.*, vol. 12, no. 1, 2013.
- [11] S. Software, "SoapUI," 2014. [Online]. Available: <https://www.soapui.org/>.