

Perancangan dan Implementasi Sistem Informasi Pembayaran pada Perguruan Tinggi Menggunakan *Web Services* Metode H2H dari Perbankan

Iis Pradesan

STMIK GI MDP: Jl. Rajawali No. 14 Palembang, 0711-376400/0711376360

Jurusan Sistem Informasi STMIK GI MDP, Palembang

e-mail: iis.pradesan@gmail.com

Abstrak

Teknologi telah berpengaruh besar di segala aspek kehidupan manusia dari kebutuhan komersial, hingga dimanfaatkan sebagai alat hiburan semata. Teknologi tersebut juga telah mempengaruhi aktivitas perbankan, nasabah tidak lagi harus bertatap muka dengan teller, cukup dengan smartphone dan koneksi internet maka semua aktivitas perbankan terselesaikan saat itu juga. Salah satu bentuk perkembangan teknologi perbankan yang telah dirilis saat ini adalah sistem pembayaran Host to Host (H2H), sistem ini akan terhubung melalui jaringan internet langsung dengan server (host) yang dimiliki sebuah institusi, sehingga request yang dilakukan terhadap data transaksi perbankan dapat diresponse secara realtime. Salah satu institusi yang dapat memanfaatkan H2H antara lain perguruan tinggi terkait aktivitas pembayaran yang dilakukan mahasiswa. Namun H2H harus dijumpai sebuah interface/aplikasi yang berfungsi sebagai web services yang tentunya mempermudah dalam melakukan komunikasi dengan H2H. Untuk itu penelitian ini bertujuan melakukan perancangan dan pengimplementasian Web Service dengan metode H2H dari perbankan, adapun metode penelitian yang digunakan adalah deskriptif dengan metodologi pengembangan sistem informasi RUP dan UML sebagai tool dalam pemodelannya dan berbasis web application menggunakan framework Laravel dalam tahap pengembangannya.

Kata kunci: H2H, Web Services, Perguruan Tinggi, Sistem Informasi, RUP, UML, Web Application.

Abstract

Technology has great influence in all aspects of human life from commercial needs, to be used as a mere entertainment tool. The technology has also affected the banking activity, the customers no longer have to face to face with the teller, simply with a smartphone and internet connection then all banking activities are resolved on the spot. One form of development of banking technology that has been released at this time is the payment system Host to Host (H2H), this system will be connected through the internet network directly with the server (host) owned by an agency, so that the request made to the data banking transactions can be responded realtime. One of the institutions that can utilize H2H among other universities related to the payment activities conducted by the students. But H2H must be bridged an interface/application that serves as a web services that certainly facilitate in communication with H2H. Therefore, this research is aimed at designing and implementing Web Service with H2H method from banking, while the research method used is descriptive with RUP and UML information system development methodology as a tool in modeling and web-based application using Laravel framework in its development stage.

Keywords: H2H, WebServices, University, Informatin System, RUP, UML, Web Application.

1. PENDAHULUAN

Kemajuan teknologi pada masa sekarang ini sudah merambah ke berbagai lini industri, termasuk di dalamnya adalah industri perbankan yang dengan berbagai instrument teknologinya telah mempermudah nasabah dalam melakukan transaksi. Perkembangan teknologi juga membuat sistem pembayaran mengalami perkembangan dari awalnya menggunakan uang tunai sebagai alat pembayaran hingga menjadi pembayaran non tunai [1], yang tadinya melayani nasabah dengan harus datang ke kantor cabang namun kini cukup menggunakan *smartphone*, dengan layanan SMS *Banking* atau *Internet Banking* maka seketika itu juga kebutuhan transaksi terselesaikan.

Lini lainnya yang memanfaatkan kemajuan teknologi adalah perguruan tinggi, pemanfaatan tersebut bukan sekedar prestise pendidikan tinggi modern, namun memang merupakan kebutuhan yang sangat mendesak karena mengingat kondisi saat ini yang berhadapan dengan kompleksitasnya data yang harus diolah yang tentu saja tidak memungkinkan lagi diselesaikan dengan sistem konvensional yang tidak terkomputerisasi. Adapun wujud dari pemanfaatan tersebut adalah Sistem Informasi Akademik (SIKAD). SIKAD adalah sistem yang dibangun untuk melakukan sentralisasi data akademik secara elektronik / *online* sehingga memudahkan semua aktor yang terlibat dalam kegiatan akademik tersebut. Secara spesifik, SIKAD memiliki beberapa karakter yang cukup luas, yaitu [2]:

- SIKAD bermakna sebagai pendekatan-pendekatan dalam melakukan proses manajemen.
- Komputer hanya merupakan komponen, atau alat bukan fokus sentral dari sistem informasi akademik.
- Pimpinan berperan aktif dalam rangka sistem sebagai pengguna informasi bukan sebagai tenaga teknis ataupun operator komputer.
- Esensi sistem informasi administrasi terletak pada sistem terpadu dan sistem terencana, bukan hanya urusan mekanisme pengolahan data.

Adapun lingkup dari SIKAD yaitu dapat dilihat pada gambar 1 berikut:



Gambar 1. Lingkup SIKAD

Dari gambar lingkup SIKAD di atas dapat dijelaskan beberapa hal sebagai berikut:

- Academic Management* meliputi absensi, jadwal, nilai, dan materi.
- Finance* meliputi pembayaran biaya pendaftaran, biaya bangunan, sks, wisuda dan pembayaran lainnya.
- Online Registration* berkaitan dengan proses penerimaan mahasiswa baru dan proses seleksi.
- Human Resource* meliputi penerimaan beasiswa, profil jenjang dosen dll.
- Digital Library* meliputi proses pendataan dan peminjaman buku/jurnal dll.
- Asset Management* berkaitan dengan pengelolaan *asset* yang dimiliki kampus seperti komputer, server, meja, kursi dll.
- Career Center* berkaitan dengan informasi lowongan pekerjaan.
- PDDIKTI Integration* berkaitan dengan proses sinkronisasi *Academic Management* dengan sistem yang dimiliki PDDIKTI dalam hal ini aplikasi *Feeder*.

Jika ditinjau dari sisi perbankan, kebutuhan akan layanan-layanan perbankan yang bersifat korporat/industri/institusi yang dalam hal ini terkait perguruan tinggi menjadi

kesempatan untuk dikembangkan teknologinya, maka munculah teknologi *Host to Host* (H2H) yang dibuat untuk menciptakan jembatan data perbankan dengan institusi khususnya perguruan tinggi.

Menurut Peraturan Otoritas Jasa Keuangan Nomor 19/POJK.3/2014 yang dimaksud dengan *H2H* adalah “sistem elektronik terenkripsi yang terhubung secara dua arah dan *realtime online* diantara dua institusi yang melakukan kerjasama”. Metode ini merupakan pendekatan terbaru dari dunia perbankan, yang salah satu alasan terciptanya metode ini adalah kebutuhan institusi atau dalam hal ini perguruan tinggi yang memiliki intensitas pembayaran yang bersifat masif sehingga tidak memungkinkan lagi menggunakan cara-cara konvensional, metode ini menekankan pada akuntabilitas dalam proses pembayaran. *H2H* akan menghubungkan infrastruktur pembayaran perguruan tinggi sebagai pengguna layanan *H2H* dengan infrastruktur perbankan dengan perantara penyedia layanan *H2H*. Karena bersifat transaksional dan terhubung dengan sistem perbankan maka dalam hal ini lembaga pengguna *H2H* bersifat pasif dalam arti hanya menyediakan *request* untuk “ditulis” oleh *H2H* dan menyediakan *response* untuk “dibaca” oleh *H2H* dengan menerapkan token yang terenkripsi yang diketahui oleh masing-masing pihak. Sebagai ilustrasi dari proses *H2H* dapat dilihat pada gambar 2 di bawah ini.



Gambar 2. Arsitektur *H2H*

Awal proses *H2H* dimulai dari infrastruktur SIAKAD, operator keuangan akan memasukan data tagihan beserta identitas mahasiswa yang akan menerima tagihan tersebut, kemudian mahasiswa memasukan identitas pembayaran pada *channel* pembayaran bank maka secara *realtime* juga *H2H* akan melakukan *request* dan menerima *response* dari SIAKAD terkait data tagihan/pembayaran dari mahasiswa bersangkutan, sehingga data pasca pembayaran murni sepenuhnya dikendalikan oleh pihak perguruan tinggi.

Berikut adalah beberapa keuntungan menggunakan layanan *H2H*:

- Bagian keuangan tidak perlu sibuk mencatat serta melayani padatnya antrian mahasiswa yang akan melakukan pembayaran.
- Semua data pembayaran termonitoring secara *realtime*.
- Sistem yang dikembangkan dibangun dengan standar keamanan yang tinggi. Standar operasi yang baku menjamin semua proses terkontrol dengan baik.
- Mahasiswa dapat melakukan pembayaran kuliah kapanpun.
- Mahasiswa dapat melakukan pembayaran melalui *channel* pembayaran apapun.
- Memberikan rasa aman kepada mahasiswa pada saat melakukan pembayaran biaya kuliah karena mahasiswa tidak perlu membawa uang *cash* ke kampus

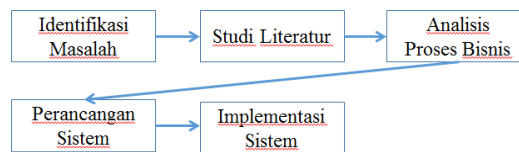
Dari penjelasan di atas maka adapun tujuan dari penelitian ini adalah melakukan perancangan dan implementasi Sistem Informasi pembayaran pada perguruan tinggi menggunakan *webservices* metode *H2H* dari perbankan.

2. METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini adalah metode penelitian deskriptif yaitu sebuah penelitian yang bertujuan untuk memberikan gambaran fakta suatu permasalahan yang terjadi saat ini dan berusaha memecahkannya menggunakan pendekatan prosedur/kaedah ilmiah.

2.1 Metodologi Penelitian

Metodologi adalah tahapan yang dilakukan dalam melakukan sebuah penelitian, dari identifikasi masalah sampai pada implementasi solusi, adapun metodologi dalam penelitian ini dapat dilihat pada gambar 1 di bawah ini:



Gambar 3. Metodologi Penelitian

a. Identifikasi Masalah

Proses pengenalan terhadap lingkungan objek penelitian harus dilakukan guna mengidentifikasi berbagai macam keluhan atau permasalahan yang dialami semua *resources* yang ada pada objek penelitian, baik dari sisi sistem berjalan maupun dari manusia yang terlibat.

b. Studi Literatur

Penyelesaian masalah harus memiliki landasan teori yang kuat, guna memangkas kesalahan-kesalahan yang timbul ketika solusi dijalankan, tahapan ini dapat berupa melakukan literasi dari *bestpractice* yang pernah dilakukan terhadap permasalahan serupa.

c. Analisa Proses Bisnis

Setelah mendapatkan prioritas permasalahan yang akan ditangani, tahap berikutnya melakukan analisa yang lebih mendalam terhadap berbagai hal yang menyebabkan permasalahan tersebut timbul dan dampaknya bagi permasalahan lain, analisa dapat berupa melakukan klasifikasi pengguna dan klasifikasi tugas dan wewenang dalam sistem.

d. Perancangan Sistem

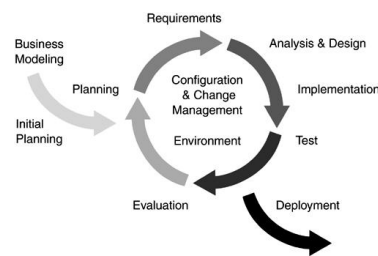
Setelah mendapatkan kata sepakat terhadap solusi tersebut maka tahap berikutnya adalah mulai menerapkan kaedah pengembangan sistem, yaitu menggunakan pendekatan *RUP*, pada tahap ini juga dilakukan pengujian/*debugging* terhadap aplikasi secara menyeluruh.

e. Implementasi Sistem

Tahap akhir dari penelitian ini adalah melakukan implemementasi sistem.

2.2 Metodologi Pengembangan Sistem

Pada penelitian ini digunakan metodologi *RUP* adalah proses rekayasa perangkat lunak yang bersifat iteratif, arsitektur-sentris dan *use-case-driven* yang terdefinisi dan terstruktur dengan baik.



Gambar 4. Proses Iterasi Dalam RUP [3]

Requirement, *analysis* dan *design* menjadi fokus terbesar dalam proses awal iterasi, kemudian implementasi dan pengujian ditahap berikutnya. Proses iterasi menyebabkan respon terhadap perubahan/kesalahan jauh lebih fleksibel untuk diperbaiki dibandingkan model *waterfall*, sehingga perangkat lunak yang dihasilkan jauh lebih berkualitas, adapun tahapan yang digunakan pada RUP adalah:

- Inception*, tahap ini meliputi penentuan ruang lingkup, *Business Case*.
- Elaboration*, tahap ini melakukan analisa kebutuhan dan resiko.
- Construction*, tahap ini melakukan analisis perancangan, implementasi dan *testing*.
- Transition*, tahap ini melakukan persiapan untuk kemudian sistem diserahkan kepada *client* dalam hal ini melakukan pembuatan buku manual, pelatihan dll.

2.3 Unified Modeling Language (UML)

UML adalah pemodel yang digunakan untuk mengkomunikasikan sistem melalui diagram dan teks pendukung, adapun tiga aspek dalam *UML* [4] yaitu:

- Language*, adalah cara berkomunikasi antara sistem dengan anggota tim pengembang (subjek) sehingga masing-masing anggota tetap dapat saling berkolaborasi untuk keberhasilan sistem.
- Model, aspek ini akan mengkonversikan *Language* kedalam sebuah model yang merepresentasikan sistem secara keseluruhan, sehingga anggota tim dapat dengan mudah memahami dampak ketika terjadi perubahan saat sistem sedang dikembangkan.
- Unified*, aspek ini mengistilahkan *UML* sebagai pemersatu antara Sistem Informasi dengan kebutuhan bisnis sehingga keduanya dapat saling berkontribusi dalam proses pengembangan.

Pemodel *UML* dibutuhkan dikarenakan sifat dari metodologi *RUP* berorientasi pada *Object* dimana proses pengembangan akan dikonversi kedalam *class-class*, adapun diagram yang akan digunakan pada tahapan ini adalah *Usecase*, *Activity*, *Sequence*, dan *Class Diagram*

2.5 Web Services

Web services adalah teknologi berbasis website yang disiapkan untuk proses pertukaran data dengan sistem/platform lain [5], *Web services* dibangun atas 5 [6] dasar standar:

- Extensible Markup Language (XML)* adalah format data standar yang digunakan dalam proses pertukaran data.
- Hypertext Transfer Protocol (Secure) *HTTP (S)* adalah Protokol pertukaran data berbasis teks yang saat ini telah menjadi protocol standar pengaksesan internet melalui browser dan media lainnya.
- Web Services Definition Language (WSDL)* adalah Standar *Web service* yang menyediakan aturan fungsi dan object.
- Simple Object Application Protocol (SOAP)* yaitu protokol yang mengemas fungsi dan objek pada saat proses pertukaran data.
- Universal Description, Discovery, and Integration (UDDI)* yaitu penampung *registry web services* yang telah dikembangkan agar dapat diakses dan diintegrasikan.

2.6 Laravel

Laravel merupakan *framework* PHP yang digunakan dalam tahap *Construction*, dengan *programming pattern Model View Controller* (MVC) dapat memudahkan dalam proses konversi dari tahap *Elaboration*. Selain menyediakan fitur *Application Programming Interface* (API) yang memungkinkan berinteraksi dengan bahas pemrograman dan platform lainnya, *Laravel* juga menyediakan fungsi *Cross-Site Request Forgery* (CSRF) pada fitur keamannya, fungsi ini melakukan *enkripsi* dua arah ketika terdapat *request* oleh *user*, sehingga hanya enkripsi yang terbentuk oleh sistem saja yang dianggap valid. Kedua fitur ini menjadi penting karena terkait interaksi dengan aplikasi/platform lain dengan kebutuhan keamanan akan data pembayaran/keuangan di dalam sistem *H2H*. *Laravel* bukan sekedar *framework*, keseluruhan tools telah dikembangkan untuk membuat aplikasi PHP lebih cepat dan nyaman [7].

3. HASIL DAN PEMBAHASAN

Pengidentifikasi kebutuhan pada penelitian ini dilakukan dengan bantuan PIECES yang membagi kebutuhan kedalam 5 kategori yaitu *Performance, Information, Control, Economy* dan *Service*. Dalam hasilnya Diagram PIECES ini digunakan untuk mendapatkan inti permasalahan yang ada saat ini, hal ini dilakukan untuk mendapatkan solusi dari permasalahan yang sedang terjadi saat ini, adapun kesimpulan permasalahan yang didapat dari hasil wawancara yang telah dilakukan yaitu berupa kesimpulan dari diagram PIECES yaitu:

Tabel 1. PIECES

P	Kesederhanaan proses
I	Informasi yang <i>realtime</i>
C	<i>Centralisasi</i> instrumen pembayaran
E	Akuntabilitas data keuangan
S	<i>feedback</i> informasi terhadap <i>user</i>

3.1 Usecase

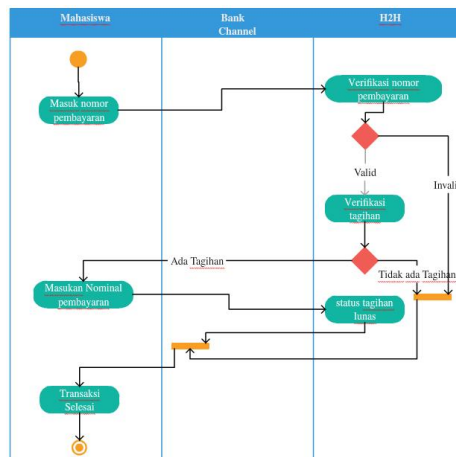
Terdapat 2 aktor utama dalam sistem ini yaitu operator keuangan dan mahasiswa itu sendiri, keduanya terhubung melalui SIAKAD untuk aktivitas tagihan dan *channel* perbankan sebagai media pembayaran. Sementara *usecase* yang digunakan adalah tagihan dan pembayaran, adapun *usecase* tagihan merupakan tanggungjawab dari operator keuangan dalam pengelolaannya, seperti menentukan jenis, tanggal awal dan akhir, besaran nominal dan mahasiswa-mahasiswa yang dibebankan tagihan tersebut.



Gambar 5. Usecase Diagram

3.2 Activity Diagram

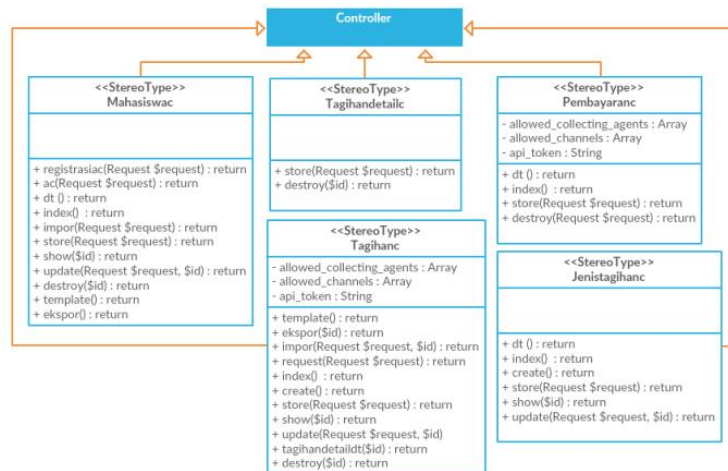
Activity Diagram pada gambar 6 dimulai dari mahasiswa memasukkan nomor pembayaran pada *channel-channel* pembayaran yang dimiliki Bank terkait, untuk kemudian nomor pembayaran tersebut divalidasi oleh *H2H*, jika dinyatakan valid maka sistem akan langsung melakukan pengecekan pada data tagihan, jika terdapat tagihan aktif maka sistem akan menampilkan deskripsi beserta nominal tagihan, untuk kemudian dilakukan pembayaran.



Gambar 6. Activity Diagram Pembayaran

3.2 Class Diagram

Pada gambar 7 di bawah ini, terdapat 4 class utama yaitu Mahasiswac, Tagihanc, Tagihandetailc, Pembayaranc, dan Jenistagihanc, yang merupakan subclass dari class Controller yang terdapat pada gambar x. Class Tagihanc dan class Pembayaranc adalah class yang digunakan sebagai API untuk H2H, dengan attribute allowed_collecting dan allowed_channels bertipe data array yang menyimpan nama Bank dan pilihan channel seperti via sms, teller, mbank, atm dll, serta juga memiliki attribute api_token bertipe data Uuid yang terenkripsi.



Gambar 7. Class Diagram H2H

Dikarenakan terhubung melalui web service maka dibutuhkanlah pemetaan data atau XML Key yang telah dimasukan pada class diagram di atas, XML Key tersebut dapat dilihat pada tabel 2 di bawah ini.

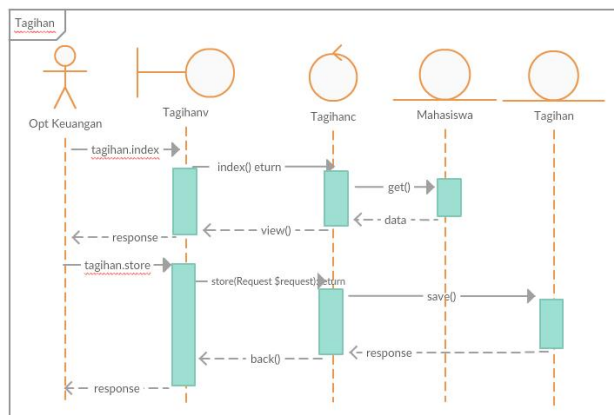
Tabel 2. XMLKey

Tagihan		Pembayaran	
Request	Response	Request	Response
Nomor pembayaran	Nomor pembayaran	Nomor pembayaran	Nomor pembayaran
Kode Bank	Id tagihan	Id Tagihan	Id tagihan
Kode Channel	Deskripsi	Kode Bank	Nomor Induk
Kode Terminal	Nominal	Kode Channel	Nama

	Nomor Induk Nama Jurusan Fakultas	Kode Terminal Id Transaksi	Jurusan Fakultas
--	--	-------------------------------	---------------------

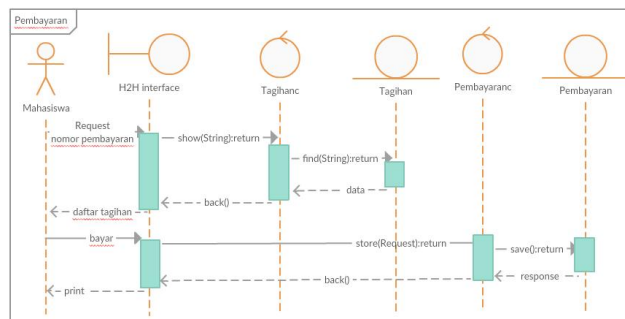
3.3 Sequence Diagram

Setelah memetakan sistem kedalam bentuk *AcitivityDiagram* maka pemodelan berikutnya dilakukan dengan menggunakan *Sequence Diagram Tagihan* yang menjelaskan seperti terlihat pada gambar 8 di bawah ini. Operator keuangan akan berinteraksi dengan halaman Tagihan yang berisikan data dari model Tagihan dengan *Controller Tagihanc* sebagai *drivernya*, *methodstore* akan dieksekusi ketika operator menambahkan tagihan baru beserta mahasiswa tertagih di dalamnya, *methodstore* akan mengeksekusi *methodsave* ke model mahasiswa dan model tagihan.



Gambar 8. Sequence Diagram Tagihan

Setelah proses pengaktifan status tagihan maka secara otomatis tagihan tersebut akan langsung terkoneksi dengan *channel-channel* pembayaran perbankan, itu artinya Mahasiswa akan berinteraksi dengan *H2H Interface (Channel)*, untuk melakukan validasi nomor tagihan oleh *controller Tagihanc* terhadap model tagihan. Jika data mahasiswa valid maka dilanjutkan melakukan pembayaran oleh *controller* pembayaran dan terakhir data pembayaran tersimpan pada model pembayaran dengan status pembayaran “Lunas”. Proses tersebut dapat dilihat pada gambar 9 di bawah ini.

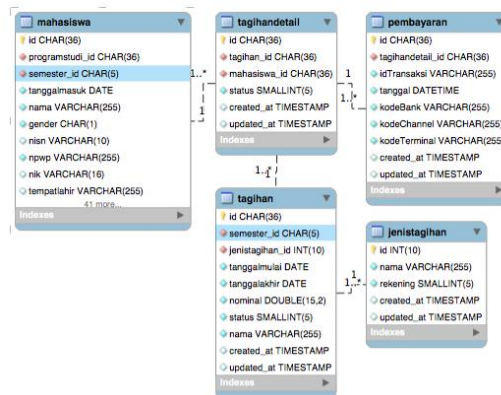


Gambar 9. Sequence Diagram Pembayaran

3.4 Struktur Data

Struktur data yang dibutuhkan terdiri dari 5 tabel yaitu mahasiswa, jenis tagihan, tagihan, tagihan detail dan pembayaran dengan 2 tabel yang terkoneksi secara langsung dengan *H2H*,

yaitu tabel tagihan dan tabel pembayaran. Untuk lebih jelasnya dapat dilihat pada gambar 10 Struktur tabel H2H di bawah ini.

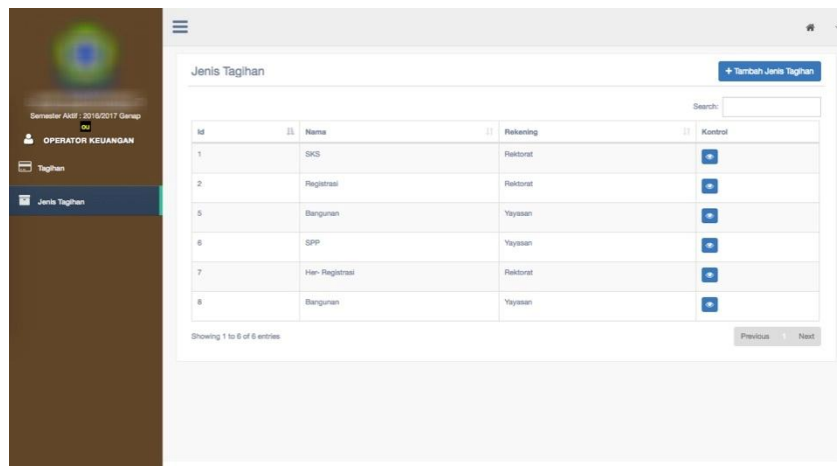


Gambar 10. Struktur Tabel H2H

3.6 Antarmuka

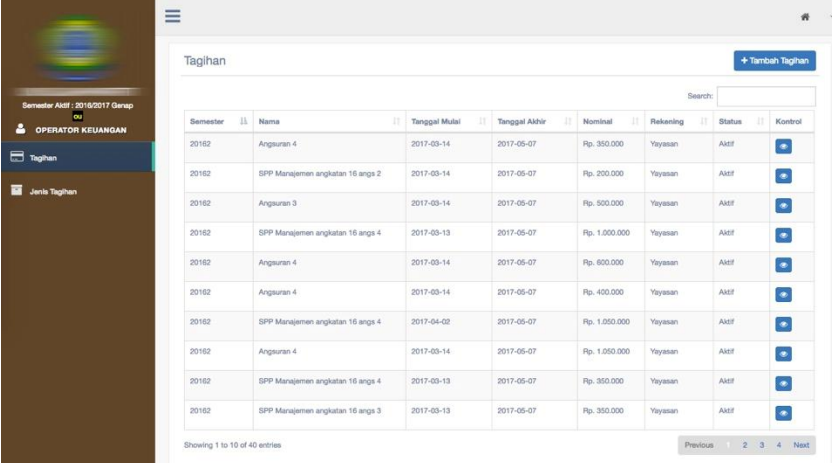
Sistem ini diimplementasikan berbasis *web application* dibangun menggunakan bahasa pemrograman PHP dengan *framework* Laravel. Adapun antarmukanya yaitu:

- Jenis Tagihan, halaman ini berkaitan dengan pengolahan jenis tagihan seperti: SKS, Registrasi, Bangunan, SPP, Wisuda dan lain-lain. Penambahan jenis tagihan dapat dilakukan dengan menekan tombol “Tambah Jenis Tagihan”, sementara untuk melakukan modifikasi data dapat dilakukan dengan menekan tombol pada kolom “Kontrol”. Untuk lebih jelasnya dapat dilihat pada Gambar 11 di bawah ini.



Gambar 11. Dashboard Jenis Tagihan

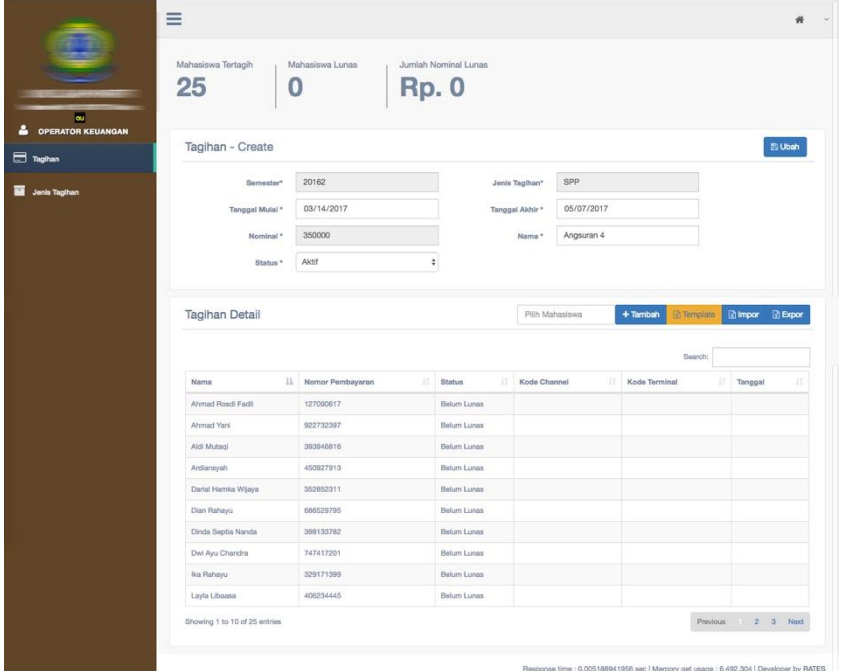
- Tagihan, operator keuangan akan melakukan pendataan tagihan yang akan *publish* ke mahasiswa dengan cara memilih jenis tagihan, nama tagihan, tanggal awal dan akhir tagihan serta nominal yang ditagihkan. Ketika tagihan berhasil dibuat maka secara *default* status dibuat non aktif, untuk kemudian diubah menjadi aktif sesuai kebutuhan, dan tagihan pun secara *realtime* terhubung dengan H2H. Antarmuka tagihan dapat dilihat pada Gambar 12 di bawah ini.



Semester	Nama	Tanggal Mulai	Tanggal Akhir	Nominal	Rekening	Status	Kontrol
20162	Angsuran 4	2017-05-14	2017-05-07	Rp. 350.000	Yayasan	Aktif	
20162	SPP Manajemen angkatan 16 angs 2	2017-05-14	2017-05-07	Rp. 200.000	Yayasan	Aktif	
20162	Angsuran 3	2017-05-14	2017-05-07	Rp. 500.000	Yayasan	Aktif	
20162	SPP Manajemen angkatan 16 angs 4	2017-05-13	2017-05-07	Rp. 1.000.000	Yayasan	Aktif	
20162	Angsuran 4	2017-05-14	2017-05-07	Rp. 600.000	Yayasan	Aktif	
20162	Angsuran 4	2017-05-14	2017-05-07	Rp. 400.000	Yayasan	Aktif	
20162	SPP Manajemen angkatan 16 angs 4	2017-04-02	2017-05-07	Rp. 1.050.000	Yayasan	Aktif	
20162	Angsuran 4	2017-05-14	2017-05-07	Rp. 1.050.000	Yayasan	Aktif	
20162	SPP Manajemen angkatan 16 angs 4	2017-05-13	2017-05-07	Rp. 350.000	Yayasan	Aktif	
20162	SPP Manajemen angkatan 16 angs 3	2017-05-13	2017-05-07	Rp. 350.000	Yayasan	Aktif	

Gambar 12. *Dashboard* Tagihan

- c. Tagihan Detail, proses penambahan mahasiswa yang tertagih dapat dilakukan pada antarmuka tagihan detail seperti pada Gambar 13 di bawah ini, dimana pencarian identitas mahasiswa dapat berupa nama atau nim kemudian menekan tombol “tambah”, maka akan secara otomatis masuk kedalam daftar mahasiswa tertagih. Informasi seperti jumlah mahasiswa tertagih, jumlah mahasiswa yang telah melunasi, serta jumlah nominal uang dari mahasiswa yang telah melunasi juga tertera pada halaman tersebut.



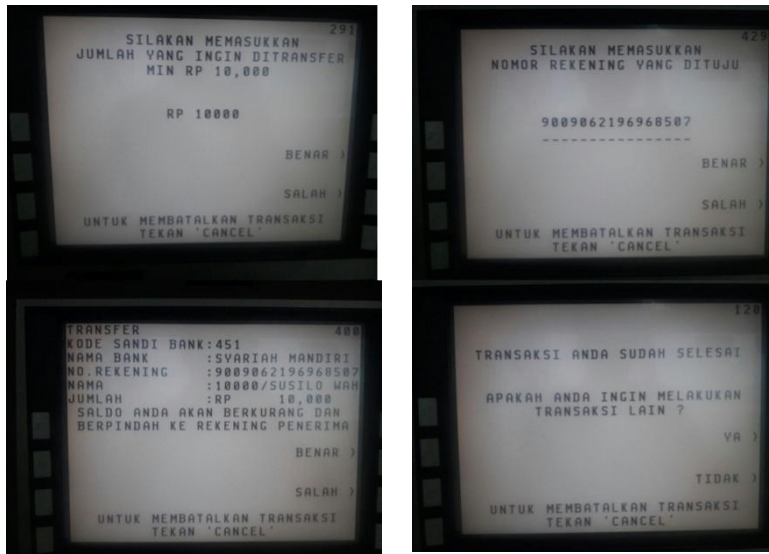
Nama	Nomor Pembayaran	Status	Kode Channel	Kode Terminal	Tanggal
Ahmad Rosdi Fadli	12709617	Belum Lunas			
Ahmad Yani	922732397	Belum Lunas			
Azil Mutaqi	393945916	Belum Lunas			
Aslanajah	450927913	Belum Lunas			
Daniel Hamka Wijaya	352852311	Belum Lunas			
Dian Rahayu	686529795	Belum Lunas			
Dinda Septia Nanda	398133782	Belum Lunas			
Dei Ayu Chandra	747417201	Belum Lunas			
Ika Rahayu	329171399	Belum Lunas			
Layli Libassa	498234445	Belum Lunas			

Gambar 13. *Dashboard* Pembayaran

- d. *Channel* ATM, setelah status tagihan diubah menjadi “aktif” maka secara otomatis tagihan tersebut akan langsung dapat diakses melalui *channel* pembayaran yang disediakan Bank terkait, adapun caranya dengan memasukkan kode Bank dan nomor pembayaran, kemudian akan tertera identitas mahasiswa dan nominal tertagih, untuk kemudian dilakukan proses

pembayaran. Gambar 14 merupakan antarmuka dari proses pembayaran tagihan melalui *channel* ATM BCA untuk pembayaran pada *H2H* Bank Syariah Mandiri.

e.



Gambar 14. Antarmuka Pembayaran pada *Channel* ATM BCA

4. KESIMPULAN

Penelitian ini telah berhasil menghasilkan *output* sebuah sistem informasi pembayaran pada perguruan tinggi menggunakan *webservice* metode *H2H* dengan menggunakan kaedah-kaedah ilmiah penelitian dan pengembangan sistem informasi, dan penelitian ini juga telah berhasil diimplementasikan baik pada sisi perguruan tinggi hingga sampai pada *channel-channel* perbankan. Diharapkan penelitian ini dapat membantu perguruan tinggi mengelolah transaksi pembayaran dengan lebih maksimal.

5. SARAN

H2H merupakan proses pembayaran yang disiapkan oleh lembaga perbankan untuk institusi-institusi yang memiliki jumlah transaksi yang besar dan rutin, itu artinya tidak hanya sebatas dapat diterapkan di perguruan tinggi saja seperti yang dilakukan pada penelitian ini. Diharapkan kedepan terdapat penelitian/pembuatan sistem informasi yang memiliki lingkup yang lebih luas dan umum, sehingga *H2H* dapat dimanfaatkan diberbagai kebutuhan institusi/perusahaan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberi dukungan baik moral dan finansial terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] Adiyanti, A.I. and Pudjihardjo, M., 2014, *Pengaruh Pendapatan, Manfaat, Kemudahan Penggunaan, Daya Tarik Promosi, dan Kepercayaan Terhadap Minat Menggunakan Layanan E-money*, Jurnal Ilmiah Mahasiswa FEB, 3(1).
 - [2] Indrayani, E., 2011, *Pengelolaan Sistem Informasi Akademik Perguruan Tinggi Berbasis Teknologi Informasi Dan Komunikasi (TIK)*, Jurnal Penelitian Pendidikan, 12(1), p.1.
 - [3] Kroll, P. and Kruchten, P., 2003, *The Rational Unified Process Made Easy: A Practitioner's Guide to The RUP*, Addison-Wesley Professional.
 - [4] Alhir, S.S., 2003, *Learning Uml*, "O'Reilly Media, Inc."
 - [5] Josuttis, N.M., 2007, *SOA in Practice: The Art of Distributed System Design*, "O'Reilly Media, Inc."
 - [6] Firmansyah, A., 2017, *Implementasi Services Oriented Architecture (SOA) Dalam Sistem Transaksi Perbankan di Perguruan Tinggi Studi Kasus: Universitas Padjadjaran*, Konferensi Nasional ICT-M Politeknik Telkom.
 - [7] Bean, M., 2015, *Laravel 5 Essentials*, Packt Publishing Ltd.
-