

Pengembangan Teknologi Single Sign On Pada Sistem Informasi Dosen dan Sistem Informasi Kurikulum di Universitas Negeri Jakarta

Hamidillah Ajie¹, Muhamad Insan Rizky², M. Ficky Duskarnaen³

^{1,2,3} Pendidikan Teknik Informatika dan Komputer Fakultas Teknik Universitas Negeri Jakarta

¹ hamidillah@unj.ac.id, ² insanrizky ptik13@mahasiswa.unj.ac.id, ³ duskarnaen@unj.ac.id

Abstrak

Fitur otentikasi dan otorisasi merupakan sebuah fitur yang umum diterapkan dalam sistem informasi. Otentikasi merupakan suatu proses untuk mengidentifikasi pengguna yang masuk ke dalam sistem sedangkan otorisasi merupakan proses pemeriksaan privileges atau hak istimewa bagi pengguna yang mengakses sistem tersebut sehingga fitur-fitur yang ada dapat disediakan sesuai dengan kebutuhan pengguna. Pada umumnya fitur otentikasi dan otorisasi diterapkan dengan menggunakan proses login. Perkembangan teknologi saat ini telah mengantarkan pengguna kepada solusi permasalahan ini yakni dengan menerapkan teknologi yang disebut Single-Sign-On (SSO). Dengan teknologi ini, layanan yang terpisah dapat terintegrasi dengan baik dalam sebuah sistem. Saat ini banyak bermunculan framework aplikasi yang siap pakai untuk mempermudah pengembang aplikasi dalam mengembangkan aplikasi lainnya. Framework Larayel merupakan salah satu framework aplikasi yang dikembangkan dengan bahasa PHP. Adapun dalam mengembangkan SSO, Laravel menyediakan fitur bernama Laravel Passport. Aplikasi yang diintegrasikan dengan sistem otentikasi ini adalah Sistem Informasi Dosen dan Sistem Informasi Kurikulum di lingkungan Universitas Negeri Jakarta. Penelitian dilakukan sejak bulan Oktober 2016 hingga Juni 2017. Metode yang digunakan pada pengembangan SSO ini adalah metode prototyping. Penelitian bermula dari menganalisa kebutuhan sistem, lalu merancang prototype diantaranya merancang database dan tampilannya. Setelah itu membuat prototype dan mengujinya. Setelah pembuatan prototype selesai, prototype diterapkan ke dalam sistem yang diintegrasikan dan dilakukan pengujian kembali.

Kata kunci : Single Sign-On, Sistem Informasi, Framework dan Prototyping

1. Pendahuluan

1.1 Latar Belakang

Penerapan teknologi informasi di berbagai bidang di sebuah perguruan tinggi menghasilkan berbagai aplikasi yang dikembangkan secara terpisah berdasarkan layanan yang dibutuhkan. Hal ini bertujuan untuk memudahkan pengembang aplikasi dalam mengatasi masalah yang terjadi pada masingmasing aplikasi.

Universitas Negeri Jakarta (UNJ) merupakan salah satu perguruan tinggi yang telah menghasilkan beberapa aplikasi yang dikembangkan secara terpisah. Adapun beberapa sistem informasi di UNJ, antara lain: Sistem Penerimaan Mahasiswa Baru (SIPENMABA), Sistem Informasi Uang Kuliah Tunggal (SIUKAT), Sistem Informasi Akademik (SIAKAD), Sistem Informasi Dosen (SIDOS) dan Sistem Informasi Kurikulum (SIKUR).

Dalam pelaksanaannya, proses otentikasi dan otorisasi pengguna pada sistem informasi di UNJ

dilakukan secara terpisah oleh masing-masing sistem yang tersedia. Setiap sistem menyimpan credential dari masing-masing pengguna dengan adanya kemungkinan informasi credential tersebut berbedabeda pada setiap sistem dengan pengguna yang sama. Hal ini dianggap cukup menyulitkan pengguna karena pengguna harus mengirimkan credential setiap kali ingin masuk ke dalam sebuah sistem dalam waktu dekat.

Perkembangan teknologi saat ini telah mengantarkan pengguna kepada solusi permasalahan ini yakni dengan menerapkan teknologi yang disebut Single-Sign-On (SSO). Menurut Doni Djayusman, SSO adalah sebuah teknologi yang mengizinkan pengguna jaringan agar dapat mengakses beberapa layanan dalam sebuah jaringan hanya dengan menggunakan satu akun pengguna saja.

Framework merupakan sebuah software untuk memudahkan para pengembang membuat aplikasi atau web yang isinya adalah berbagai fungsi, plugin, dan kerangka konsep sehingga membentuk suatu sistem tertentu. Dengan menggunakan framework, sebuah aplikasi akan tersusun dan terstruktur dengan rapi. Pemilihan sebuah framework tentu menjadi kebijakan dari masing-masing pengembang tergantung dari kebutuhan dan kemampuan pengembang tersebut. Framework Laravel merupakan salah satu framework aplikasi yang dikembangkan dengan bahasa PHP. Adapun dalam mengembangkan SSO, Laravel menyediakan fitur bernama Laravel Passport pada versi 5.3.

1.2 Identifikasi Masalah

Berdasarkan latar belakang, maka dapat diidentifikasi berbagai masalah sebagai berikut:

- 1. Pengelolaan data akun pengguna dilakukan pada masing-masing sistem informasi sehingga terdapat *redundancy* data;
- Data akun pengguna yang disimpan secara terpisah mengakibatkan pengguna harus mengingat banyak data akun dan harus memasukkan *credential* setiap kali ingin mengakses suatu sistem.

1.3 Tujuan

Berdasarkan perumusan masalah yang telah dirumuskan sebelumnya maka tujuan dari penelitian adalah:

- 1. Mengintegrasikan pengelolaan akun pengguna pada sistem informasi yang tersedia di UNJ;
- Meningkatkan kemudahan pengguna dalam mengakses sistem informasi yang tersedia di UNJ.

1.4 Manfaat

Kegunaan dari penelitian adalah untuk memudahkan dosen dan mahasiswa UNJ dalam menggunakan berbagai sistem informasi yang tersedia hanya dengan satu akun, serta memudahkan pihak UNJ dalam mengelola peran untuk masingmasing akun sehingga otorisasi menjadi lebih teratur.

2. Dasar Teori

2.1. Sistem Informasi Dosen

SIDOS merupakan layanan sistem informasi bagi seluruh dosen UNJ yang mencerminkan profil data pribadi, data pendidikan, data mengajar, data penelitian, data pengabdian masyarakat, data pertemuan ilmiah, data prestasi, data pekerjaan, data keanggotaan profesi, keanggotaan organisasi masyarakat, dan publikasi ilmiah. Tujuan dari SIDOS ialah untuk memberikan informasi dosen UNJ secara lebih luas dan lengkap baik kepada masyarakat internal maupun eksternal UNJ. Layanan SIDOS dapat diakses pada http://sidos.unj.ac.id (Sistem Informasi Dosen, 2017).

2.2. Sistem Informasi Kurikulum

SIKUR merupakan layanan sistem informasi yang mengelola informasi kurikulum yang sudah,

sedang, dan akan digunakan di UNJ berdasarkan masing-masing program studi. Pengguna SIKUR tidak hanya dosen saja melainkan seluruh civitas akademika UNJ dengan *privileges* yang berbeda untuk setiap jenis penggunanya (UPT TIK, 2017).

2.3. Single Sign-On

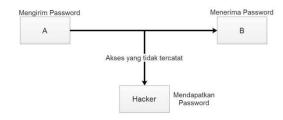
Teknologi *Single Sign On* (SSO) adalah sistem yang mengizinkan pengguna agar dapat mengakses seluruh sumber daya dalam jaringan hanya dengan menggunakan satu *credential* saja. Sistem ini tidak memerlukan interaksi yang manual, sehingga memungkinkan pengguna melakukan proses sekali *login* untuk mengakses seluruh layanan aplikasi tanpa berulang kali memasukkan *password* setiap kali memasuki masing-masing aplikasi. Teknologi ini sangat diminati dalam jaringan yang sangat besar dan bersifat heterogen, dimana sistem operasi serta aplikasi yang digunakan berasal dari banyak *vendor*, dan pengguna diminta untuk mengisi informasi dirinya ke dalam setiap *multi-platform* yang hendak diakses. Djayusman, Doni (2013).

2.4. Laravel Passport

Laravel adalah *framework* PHP dengan kode terbuka (*open source*) dengan desain MVC (*Model-View-Controller*) yang digunakan untuk membangun aplikasi *website*. *Framework* ini pertama kali dibangun oleh Taylor Otwell pada tanggal 22 Pebruari 2012. Tim Air Putih (2014).

Salah satu fitur terbaru dari Laravel versi 5.3 adalah Laravel Passport. Fitur ini diadopsi dari *module* yang bernama *League Oauth2 Server* yang dikembangkan oleh Alex Bilbie pada akun Githubnya. Laravel Passport menyediakan implementasi OAuth2 *Server* secara penuh untuk aplikasi Laravel. Laravel *Documentation*, (2017).

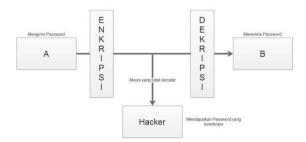
2.5. HTTP & HTTPS



Gambar 2.1 Cara Kerja HTTP

Pada Gambar 2.1, HTTP digambarkan tidak memiliki kemanan yang terjamin. Saat proses pentransferan *password*, *Hacker* dapat dengan mudah mengambil ditengah jalan proses pentransferan tanpa

diketahui oleh pihak pengirim maupun penerima password.



Gambar 2.2 Cara Kerja HTTPS

HTTPS mentransmisikan keamanan data dengan menggunakan koneksi terenkripsi. Pada dasarnya ini menggunakan kunci publik yang kemudian didekripsi pada sisi penerima. Kunci publik digunakan pada server, dan termasuk dalam apa yang Anda ketahui sebagai sertifikat SSL. Sertifikat ini kriptografi ditandatangani oleh Otoritas Sertifikat (CA), dan masing-masing browser memiliki daftar CA secara implisit percaya. Setiap sertifikat yang ditandatangani oleh CA dalam daftar dipercaya diberikan kunci gembok hijau di address bar browser, karena itu terbukti "terpercaya" dan milik domain tersebut. Perusahaan seperti Mari Encrypt sekarang telah membuat proses penerbitan sertifikat SSL gratis.

2.6. Model Prototyping

Prototyping adalah proses iteratif dalam pengembangan sistem dimana kebutuhan diubah ke dalam sistem yang bekerja (working system) secara terus menerus diperbaiki melalui kerjasama antara pengguna dan analis. Prototype juga bisa dibangun melalui beberapa tool pengembangan untuk menyederhanakan proses. Prototyping merupakan bentuk dari Rapid Application Development (RAD). Fatta (2007).

3. Metodologi Penelitian



Gambar 3.1 Diagram Alir Penelitian

Metode yang digunakan pada pengembangan SSO ini adalah metode *prototyping*. Penulis

menggunakan metode ini karena untuk mempercepat proses pengembangan sistem sesuai dengan kebutuhan yang dijelaskan oleh pengguna sehingga apabila terjadi perubahan *system environment*, pengembangan teknologi SSO dapat segera beradaptasi dengan perubahan tersebut.

3.1 Daftar Kebutuhan Fungsional

Dalam pengembangan SSO, dibuat daftar kebutuhan fungsional yang dapat dilihat pada Tabel 3.1 dan Tabel 3.2:

No.	Deskripsi Fungsional			
1	Mengarahkan pengguna ke halaman login apabila pengguna mengakses aplikasi client tanpa memiliki token			
2	Memverifikasi <i>credential</i> yang dimasukkan oleh pengguna dengan data pengguna di <i>database</i>			
3	Mengarahkan pengguna ke halaman callback ketika sedang melakukan verifikasi credential			
4	Memverifikasi secret key dan id aplikasi client yang dikirim dari aplikasi dengan data yang ada di database			
5	Menghasilkan token dan mengirimkannya ke aplikasi client			
6	Mengatur lifetime token yang diberikan ke aplikasi client			
7	Memverifikasi <i>token</i> apabila pengguna mengakses aplikasi yang berbeda ketika sudah melakukan <i>login</i>			
8	Menonaktifkan token yang dikirim pengguna sebagai proses logout dari seluruh aplikasi			

Tabel 3.1. Daftar Fungsional Utama

No.	Deskripsi Fungsional				
1	Mengarahkan tampilan ke halaman panel admin apabila pengguna menekan				
	tombol menu panel admin				
2	Menampilkan daftar aplikasi client yang sudah terdaftar				
3	Menampilkan form untuk pengisian aplikasi client				
4	Menyimpan data aplikasi client baru				
5	Menghasilkan secret key dan id aplikasi client yang baru ditambahkan				
6	Menampilkan data aplikasi client ketika admin menekan tombol ubah pada				
	daftar aplikasi <i>client</i>				
7	Menghapus data aplikasi client ketika admin menekan tombol hapus pada				
	daftar aplikasi <i>client</i> tersebut				
8	Mengarahkan tampilan ke halaman data pengguna apabila admin menekan				
	tombol menu data pengguna				
9	Menampilkan daftar data pengguna yang tersimpan di database				
10	Menampilkan form untuk pengisian data pengguna				
11	Menyimpan data pengguna baru				
12	Menampilkan data pengguna ketika admin menekan tombol ubah pada daftar				
	data pengguna				
13	Menghapus data pengguna ketika admin menekan tombol hapus pada data				
	pengguna tersebut				
14	Mengarahkan tampilan ke domain aplikasi tujuan yang dipilih oleh				
	pengguna				

Tabel 3.2. Daftar Fungsional Admin

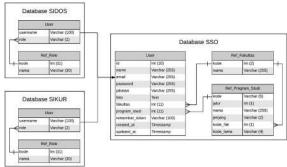
3.2 Model Prototype SSO



Gambar 3.2 Model Single Sign-On

Pada Gambar 3.1 menjelaskan alur proses SSO secara keseluruhan. Diawali dengan pengguna yang memasukkan *credential* ke SSO kemudian *credential* tersebut akan diverifikasi dengan data yang ada pada database SSO. Jika credential tersebut telah terverifikasi, maka SSO akan mengembalikan token kepada pengguna dan token tersebut akan disimpan pada *cookies browser* pengguna.

3.3 Struktur Database



Gambar 3.3 Hubungan Antara *Database* SSO dengan *Database* SIDOS dan *Database* SIKUR

Terdapat 3 container pada Gambar 3.3, yaitu: database SSO, database SIDOS, dan database SIKUR. Pada database SSO terdapat 3 tabel yang akan berkaitan dengan database SIDOS dan SIKUR. Namun sebenarnya, hanya 1 tabel yang benar-benar berkaitan secara langsung dengan SIDOS dan SIKUR, yaitu: tabel user. Ketika pengguna sudah berhasil memperoleh token dari SSO, maka token tersebut dapat digunakan sebagai tiket untuk memperoleh informasi pengguna yang sedang aktif. Data user yang dikirim dari SSO berupa json. Dari json itulah hubungan SSO dengan SIDOS dan SIKUR terjadi. SIDOS dan SIKUR hanya perlu mencari username pada tabel user mereka berdasarkan data json dari SSO.

4. Hasil dan Analisis

Setelah produk berhasil dikembangkan, selanjutnya dilakukan pengujian produk untuk menguji kelayakan dari produk tersebut. Pengujian dengan metode black box testing dilakukan dengan menjalankan seluruh fungsional yang telah didefinisikan dan menganalisis keselarasan antara hasil yang diharapkan pada skenario proses dan feedback dari sistem.

Adapun hasil pengujian produk dapat dilihat pada Tabel 4.1 dan Tabel 4.2:

Tabel 4.1. Pengujian Fungsional Utama

No.	Fungsi	Skenario Proses	Hasil yang Diharapkan	Sistem Bekerja (Ya/Tidak)*
1	Redirecting	Pada saat pengguna mengakses domain client	Diarahkan ke halaman login SSO server dengan membawa kode Client ID	Ya
2	Login Client	Pada saat pengguna memasukkan username dan password, klik tombol login	Diarahkan ke halaman callback client dan melakukan otentikasi. Jika username dan password benar, maka SSO server akan menghasilkan token dan mengarahkan ke domain client utama. Jika salah, muncul pesan kesalahan dan mengarahkan ke halaman login SSO	Ya
3	Otentikasi Terpusat	Pada saat pengguna sudah berhasil <i>login</i> pada satu aplikasi	Diarahkan ke halaman callback client dan melakukan otentikasi. Jika data pengguna terdapat di aplikasi client yang akan	Ya

		client, kemudian pengguna mengakses domain client lainnya	diakses, maka SSO server mengirimkan token ke domain client dan diarahkan ke halaman utama client. Jika tidak, maka diarahkan ke halaman utama SSO server	
4	Otorisasi	Pada saat berhasil login di SSO server dan diarahkan ke halaman utama aplikasi client	User ID akan dicocokkan dengan tabel role database client kemudian menampilkan menu sesuai dengan role tersebut. Jika tidak ada role, maka diarahkan ke halaman utama SSO server	Ya
5	Logout	Pada saat pengguna klik tombol logout	Sistem akan menghapus sesi dan menonaktifkan token pada SSO server dan client kemudian mengarahkan ke halaman login	Ya

Tabel 4.2. Pengujian Fungsional Utama

27	L .	GI :	TT 1	G: (
No.	Fungsi	Skenario Proses	Hasil yang Diharapkan	Sistem Bekerja
		110565	Бінагаркан	(Ya/Tidak)*
1	Navigasi	Pada saat	Tampilan akan	Ya
		pengguna	diarahkan ke panel	
		sudah masuk ke	admin	
		sistem, klik		
		menu panel admin		
2	Navigasi	Pada saat	Sebuah borang akan	Ya
		berada di	muncul bergeser dari	
		halaman panel	atas	
		admin, klik tombol create		
		new client		
3	Navigasi	Pada daftar	Sebuah borang akan	Ya
	, S	client di	muncul beserta data	
		halaman panel	client yang sedang	
		admin, klik	diubah	
		tombol edit pada client		
4	Navigasi	Pada saat	Tampilan akan	Ya
1	3	pengguna	diarahkan ke	
1		sudah masuk ke	halaman data	1
		sistem, klik menu data	pengguna	
		pengguna		
5	Navigasi	Pada saat	Sebuah borang akan	Ya
		berada di	muncul bergeser dari	
		halaman data	atas	
		pengguna, klik tombol tambah		
		pengguna		
6	Navigasi	Pada saat	Borang akan tertutup	Ya
		borang	dengan bergeser ke	
		berbentuk	atas	
		modal muncul, klik tombol		
		cancel		
7	Navigasi	Pada halaman	Tampilan akan	Ya
		utama, klik	diarahkan ke domain client	
8	Menyimpan	salah satu <i>client</i> Pada saat	Data <i>client</i> baru akan	Ya
0	Client	mengisi borang	tersimpan dan	1 4
		client,	tampilan diarahkan	
		kemudian klik	ke panel admin	
		tombol submit	beserta daftar <i>client</i> terbaru akan muncul	1
9	Mengubah	Pada saat	Data <i>client</i> akan	Ya
_	Client	mengisi borang	diperbarui dan	1
		ubah client,	tampilan diarahkan	
1		klik tombol	ke panel admin	1
		submit	beserta daftar <i>client</i> terbaru akan muncul	1
10	Menghapus	Pada daftar	Data client akan	Ya
	Client	client di	terhapus dan	1
		halaman panel	tampilan diarahkan	1
		admin, klik	ke panel admin	
		tombol delete	beserta daftar <i>client</i> terbaru akan muncul	1
11	Menyimpan	Pada saat	Data pengguna baru	Ya
	data	mengisi borang	akan tersimpan dan	1
	pengguna	pengguna,	tampilan diarahkan	
		kemudian klik tombol submit	ke halaman data	1
		toiliboi <i>Suomili</i>	pengguna beserta daftar pengguna	
			terbaru akan muncul	
12	Mengubah	Pada saat	Data pengguna akan	Ya
	data	mengisi borang	diperbarui dan	1
	pengguna	ubah pengguna, klik tombol	tampilan diarahkan ke halaman data	
		submit	pengguna beserta	

			daftar pengguna terbaru akan muncul	
13	Menghapus data pengguna	Pada daftar pengguna di halaman data pengguna, klik tombol delete	Data pengguna akan terhapus dan tampilan diarahkan ke halaman data pengguna beserta daftar pengguna terbaru akan muncul	Ya

4.1 Pembahasan

Pengujian dilakukan oleh 2 orang penguji yang bekerja di UPT TIK Universitas Negeri Jakarta, yaitu: programmer yang bernama Fajar Maulana, S.Pd. dan web designer yang bernama Hanifa Fissalma, S.Pd.. Penguji menggunakan sebuah komputer yang terkoneksi dengan server prototype SSO yang merupakan komputer pribadi penulis dan melakukan pengujian berdasarkan fungsional pada Tabel 3.1 dan Tabel 3.2. Masing-masing penguji menggunakan akun yang berbeda untuk mendapatkan hasil pengujian yang optimal.

Pengujian berlangsung pada tanggal 20 Juli 2017 di UPT TIK UNJ dan berjalan dengan cukup baik namun sempat terjadi error pada saat proses logout. Ketika pengguna menekan tombol logout pada aplikasi SIKUR kemudian kembali mengakses aplikasi SIDOS, pengguna masih bisa menggunakan fitur-fitur yang ada pada SIDOS padahal seharusnya sesi untuk aplikasi SIDOS juga turut dinonaktifkan bersamaan pada saat sesi SIKUR dan SSO dihapus. Namun tidak sebaliknya, ketika pengguna menekan tombol logout pada aplikasi SIDOS kemudian kembali mengakses aplikasi SIKUR, maka pengguna akan diarahkan ke halaman login SSO karena dianggap sudah tidak memiliki sesi yang aktif atau dengan kata lain token telah kedaluwarsa. Setelah diteliti oleh penulis, ternyata terdapat script yang belum dijalankan pada aplikasi SIDOS yakni pengecekan token SSO yang seharusnya dieksekusi pada setiap request sebagai acuan bahwa pengguna masih memiliki sesi aktif di salah satu aplikasi yang terintegrasi dengan SSO. Solusi yang dilakukan adalah menambahkan script tersebut ke dalam aplikasi SIDOS. Setelah perbaikan dilakukan, maka pengujian pada fungsi logout dilakukan kembali dan memberikan hasil yang diharapkan.

Pada akhirnya, berdasarkan hasil pengujian pada Tabel 4.1 dan Tabel 4.2 yang telah dievaluasi oleh penguji, dapat dilihat bahwa hasil pengujian menunjukkan seluruh fungsional sistem otentikasi telah berjalan dengan baik sesuai dengan skenario proses yang diharapkan. Oleh sebab itu, aplikasi SSO dianggap layak untuk digunakan oleh pengguna secara massal.

5. Kesimpulan dan Saran

Dari hasil penelitian yang telah diimplementasikan, maka dapat ditarik kesimpulan sebagai berikut:

 Pengembangan teknologi SSO pada SIDOS dan SIKUR di Universitas Negeri Jakarta dilakukan dengan membuat sistem otentikasi terpusat yang mengarahkan pengguna ke halaman login ketika pengguna mengakses langsung ke aplikasi yang sudah terintegrasi SSO. Kemudian meminta credential pengguna dan memverifikasinya. Jika credential tersebut valid, maka sistem memberikan token sebagai tiket untuk masingmasing aplikasi. Masing-masing aplikasi harus melakukan pengecekan token pada setiap request yang dilakukan pengguna. Pengguna tidak perlu melakukan login lagi jika ingin mengakses aplikasi lainnya. Apabila pengguna telah keluar dari salah satu aplikasi yang terintegrasi dengan SSO, maka sistem akan menonaktifkan token yang dimiliki oleh pengguna dan pengguna harus kembali memasukkan credential untuk proses login;

- 2. Data pengguna untuk otentikasi disimpan secara terpusat pada database SSO dan data role pengguna tetap disimpan dan dikelola oleh masing-masing aplikasi;
- Aplikasi SIKUR yang diintegrasikan dengan SSO dibuat dalam bentuk prototype karena aplikasi tersebut masih dalam tahap pengembangan. Prototype **SIKUR** dibuat dengan batasan kebutuhan manajemen pengguna saja artinya tidak ada fitur SIKUR yang benar-benar aktif kecuali pengelolaan role pengguna dan pengaturan menu berdasarkan role pengguna yang sedang aktif;
- 4. Berdasarkan hasil pengujian dengan menggunakan black box testing pada masingmasing fungsional yang digambarkan pada Tabel 4.1 dan Tabel 4.2, dapat disimpulkan bahwa sistem otentikasi SSO telah berjalan dengan baik sesuai dengan kebutuhan fungsional dan layak untuk digunakan.

Adapun saran untuk penelitian berikutnya antara lain sebagai berikut:

- 1. Melakukan penelitian mengenai Single Sign-On dengan menggunakan LDAP (Lightweight Directory Access Protocol) sebagai media penyimpanan data pengguna;
- 2. Meningkatkan keamanan pada sistem otentikasi.

Daftar Pustaka:

Abdurrahman, Luthfi, (2012), Implementasi Sistem Single Sign On menggunakan OpenAM dengan Otentikasi Kerberos dan OpenLDAP [Skripsi], Jurusan Teknik Elektro, Universitas Sumatra Utara

Aminudin, (2008), Implementasi Single Sign On (SSO) untuk Mendukung Interaktivitas Aplikasi E-Commerce menggunalan Protocol OAUTH. Malang, Jurnal GAMMA, ISSN 2086-3071:109-

Anhar, (2010), Panduan Menguasai PHP & MySQL secara Otodidak, Jakarta Selatan, Mediakita.

Anzizhan, Syafruddin, (2004), Sistem Pengambilan Keputusan Pendidikan, Surabaya, Grasindo.

- Argawal, B.B, Tayal, S.P dan Gupta, M. (2010), *Software Engineering and Computer Software Testing*, United State, Sudbury.
- Arief, Muhammad, (2011), Pemograman *Web* Dinamis menggunakan PHP dan MySQL, Yogyakarta, Andi Offset.
- Buecker, Axel, dkk., (2012), *Enterprise Single Sign-On Design Guide*, Amerika Serikat, International Business Machine Corporation.
- Djayusman, Doni, (2013), Pembangunan Aplikasi *E-Commerce* di *Mag* and *Shoes Shop* [Skripsi]. Bandung, Fakultas Teknik dan Ilmu Komputer, Unikom.
- Fatta Al, Hanif, (2007), Analisis & Perancangan Sistem Informasi, Yogyakarta, CV. Andi Offset.
- Gaffin, J.C, (2007), *Internet Protocol 6*, New York, Nova Science Publisher.
- Hutahaean, Jepersen, (2014), Konsep Sistem Informasi, Yogyakarta, Deepublish.
- IdCloudHost. Pengertian dan Keunggulan Framework Laravel. Diambil dari https://idcloudhost.com/pengertian-dan-keunggulan-framework-laravel/ pada tanggal 28 Juli 2017 Pukul 10.15 WIB.
- Jimmy, Chr, (2008), Sistem Informasi Manajemen. Surabaya, Grasindo.
- Keycdn, (2016). What is the Difference Between HTTP and HTTPS?. Diambil dari https://www.keycdn.com/blog/difference-between-http-and-https/, Diakses pada tanggal 17 April 2017 Pukul 12.53 WIB.
- Kusrini, (2007), Strategi Perancangan dan Pengelolaan Database, Yogyakarta, CV. Andi Offset.
- Laravel Documentation. *API Authentication* (*Passport*). Diambil dari https://laravel.com/docs/5.4/passport pada tanggal 28 Juli 2017 Pukul 10.34 WIB.
- Marimin, dkk., (2006). Sistem Informasi Manajemen Sumber Daya Manusia, Bogor, Grasindo.
- Microsoft, (2003). What is TLS/SSL. Diambil dari https://technet.microsoft.com/en-us/library/cc784450(v=ws.10).aspx, Diakses pada tanggal 17 April 2017 Pukul 13.34 WIB.
- Mozilla, *Tentang Cookie*, Diambil dari https://support.mozilla.org/t5/Cookies-and-cache/Tentang-Cookie/ta-p/17722. Diakses pada tanggal 17 April 2017 Pukul 14.56 WIB.
- OAuth2, *Halaman Utama*, Diambil dari https://oauth.net/2/. Diakses pada tanggal 19 Juli 2017 Pukul 11.22 WIB.
- Octafian, Tri, (2015), Web Multi E-Commerce berbasis Framework Codelgniter. Palembang, Jurnal Teknologi dan Informatika (TEKNOMATIKA):1-22
- PHP, What is PHP?, Diambil dari http://php.net/manual/en/intro-whatis.php. Diakses pada tanggal 17 April 2017 Pukul 14.03 WIB.

- Pressman, Roger. S, (2012). Rekayasa Perangkat Lunak – A Practitioner's Approach, New York : McGraw Hill.
- Ramadhan, Gilang, (2012), Analisis Teknologi *Single Sign On* (SSO) dengan Penerapan *Central Authentication Service* (CAS) pada Universitas Bina Darma [Skripsi], Fakultas Ilmu Komputer, Universitas Bina Darma Palembang.
- Simarmata, Janner, (2010), Rekayasa *Web*, Yogyakarta, CV. Andi Offset.
- Sistem Informasi Dosen, *Halaman Login*. Diambil dari http://sidos.unj.ac.id, Diakses pada tanggal 15 Juni 2017 Pukul 13.56 WIB.
- Stephen, dan Plew, (2000). *Database Design*, USA, Sams Publishing.
- Tim Air Putih, (2014), Panduan Framework Laravel. Jakarta, Creative Commons.
- Universitas Negeri Jakarta, *Halaman Sejarah*. Diambil dari http://unj.ac.id, Diakses pada tanggal 15 Juni 2017 Pukul 13.40 WIB.