Perancangan Aplikasi *Single Sign-On* (SSO) Menggunakan Otentikasi Gambar

Guntoro[1], Muhammad Fikri[2]
[1]Teknik Informatika, Fakultas Ilmu Komputer, Universitas Lancang Kuning
[2]Teknik Informatika, Fakultas Sains dan Teknologi, Universitas SUSKA Riau
e-mail: guntoro@unilak.ac.id

Abstrak

Aplikasi single sign-on (SSO) adalah sebuah sistem otentikasi login yang mengizinkan bagi seorang pengguna dapat mengakses banyak sistem hanya dengan satu akun aja. Dengan sistem single sign-on (SSO) tersebut, seorang user sistem aplikasi hanya cukup melakukan otentikasi sekali saja untuk masuk ke semua layanan yang terdapat pada dalam sistem aplikasi. Otentikasi login berbasis teks pada sistem single sign-on (SSO) yang sudah ada saat ini, mempunyai kelemahan, salah satunya adalah pencurian password dengan aplikasi keylogger. Perancangan aplikasi sistem single sign-on (SSO) yang dikembangkan dengan mencoba menambahkan otentikasi menggunakan gambar. Gambar yang digunakan telah diberikan sebuah keamanan yaitu menggunakan teknik steganografi dengan metode Least Significant Bit.

Kata kunci: Single Sign-On, Autentikasi, Gambar Least Significant Bit

Abstract

Single sign-on (SSO) application is a login authentication system that allows a user to access multiple systems with just one account. With a single sign-on (SSO) system, an application system user only just authenticates once to log in to all services contained in the application system. Text-based login authentication on existing single sign-on (SSO) systems, has a weakness, one of which is password theft with keylogger apps. The design of single sign-on (SSO) system applications developed by trying to add authentication using images. The image used has been given a security that is using steganography technique with a method of the Least Significant Bit.

.

Keywords: Single Sign-On, Authentication, Picture, Least Significant Bit

1. Pendahuluan

Otentikasi adalah suatu proses verifikasi untuk menentukan apakah seseorang berhak mengakses suatu aplikasi web maupun tidak [1]. Cara yang paling sederhana adalah dengan menggunakan otentikasi *login*, di mana seorang *user* memasukkan *username* dan *password* (*credential*), selanjutnya akan di verifikasi oleh sistem, apakah *credential* tersebut *valid* atau tidak valid, jika *credential* tersebut *valid* maka seorang *user* tersebut boleh mengakses ke dalam sistem, jika tidak *valid* maka *user* tidak berhak mengakses ke dalam sistem [2]. Sebagian besar aplikasi *web* saat ini menggunakan cara tersebut yaitu menggunakan sistem login.

Menjadi suatu masalah ketika seorang pengguna memiliki banyak aplikasi *web* yang membutuhkan otentikasi. Dia harus menghafal banyak *credential*, walaupun banyak orang membuat *credential* yang sama untuk berbagai aplikasi *web* [3]. Terdapat masalah lagi jika pengguna membuat satu *credential* untuk berbagai aplikasi *web*, karena pengguna harus memasukkan *credential* berulang kali. Oleh karena itu dibutuhkan suatu sistem yang dapat mengintegrasikan seluruh layanan aplikasi dan mengelola proses autentikasi masing-masing sistem layanan, menjadi proses autentikasi. Proses autentikasi pada sistem yang terintegrasi ini memerlukan sebuah sistem tambahan yang menjadi penghubung antara sistem *integrator* dengan sistem layanan aplikasi. Sistem inilah yang dapat menangani seluruh autentikasi setiap aplikasi sistem, sistem ini dikenal dengan Sistem *Single Sign-On* (SSO).

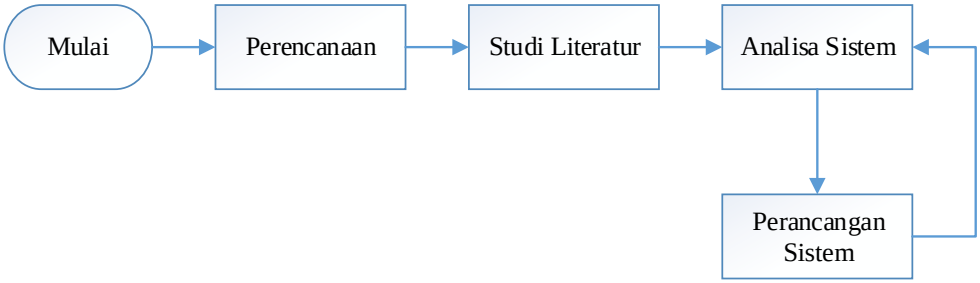
Sistem *Single Sign-On* (SSO) merupakan sebuah teknologi yang mengizinkan pengguna jaringan agar dapat mengakses sumber daya dalam jaringan hanya dengan menggunakan satu akun pengguna saja [4]. Keuntungan dari sistem *single sign-on* (SSO*)* adalah user tidak perlu banyak mengingat username dan password serta memudahkan dalam pemrosesan data [5]. Autentikasi *login* berbasis teks pada sistem *single sign-on* (SSO) yang sudah ada saat ini, mempunyai kelemahan, salah satunya adalah pencurian *password* dengan aplikasi *keylogger*. Oleh karena itu untuk meminimalisir kelemahan tersebut, diterapkan autentikasi berbasis teks dan gambar [6], yang mana gambar tersebut sudah diberikan keamanan menggunakan steganografi berbasis *Least Significant Bit.*

Pada penelitian yang berjudul ”*Open Source in Web-based Applications: A Case Study on Single Sign-On”* oleh [7], penelitian tersebut mengevaluasi sistem *single sign-on* berbasis *open source* menggunakan aplikasi CAS (*Central Authentication Service*) yang dikembangkan oleh Yale University, SourceID dan JOSSO (*Java Open Single Sign-On)* serta melakukan perbandingan terhadap kegita aplikasi *single sign-on* tersebut. Penelitian yang dilakukan oleh Andreas Pashalidis dan Chris J. Mitchell [8] membahas tentang pendekatan sistem single sign-on masa depan dalam konteks yang lebih terstruktur, skema sistem sso serta beberapa perbedaan penting dalam keamanan *single sign-on*. Pada penelitian [9] membahas tentang implementasi sistem *single sign-on* berbasis *java* dengan menggunakan aplikasi *Java Open Single Sign-On* (JOSSO). Penelitian [10] membahasa bagaimana cara melakukan integrasi aplikasi menggunakan *single sign-on* (SSO) berbasiskan LDAP di dalam web portal Bina Nusantara.

Berdasarkan latar belakang diatas maka penulis merumuskan pada penelitian ini adalah bagaimana merancang aplikasi *single sign-on* (SSO*)* menggunakan autentikasi gambar. Adapun autentikasi gambar pada perancangan *single sign-on* (SSO) ini menggunakan metode *least significant bit.* Diharapkan dengan adanya perancangan tersebut dapat membantu dalam hal keamanan sistem *single sign-on* (SSO) yang ada sekarang ini.

2. Metode Penelitian

Penelitian ini dilakukan dengan beberapa tahapan yaitu perencanaan, studi literatur, analisa dan perancangan.



Gambar 1. Metode Penelitian

2.1 Perencanaan

Pada tahapan ini dilakukan dengan mendiskusikan penelitian dengan para ahli yang berkompeten terkait dengan penelitian, menganalisa sasaran dari pengembangan penelitian dan digambarkan berdasarkan permasalahan yang ada. Selanjutnya mempelajari batasan ruang lingkup dari penelitian, kemudian dapat ditentukan metode apa yang digunakan pada penelitian ini.

2.2 Studi Literatur

Pada tahapan ini dilakukan untuk mencari sumber-sumber informasi, konsep-konsep yang mendasar terkait dengan sistem *single sign-on*, konsep *least significant bit*, penggunaan *database*. Materi ini diperoleh dari buku-buku, jurnal, makalah serta artikel-artikel di *internet*.

2.3 Analisa

Analisa dilakukan setelah data yang dikumpulkan telah lengkap, analisa ini menjabarkan beberapa data pendukung serta membahas dan menyelesaikan permasalahan-permasalahan yang akan diterapkan untuk membangun sistem. Adapun analisa yang dilakukan diantaranya yaitu :

1. Analisa sistem *single sign-on* (SSO), serta proses otorisasi sistem
2. Analisa proses metode *least significant bit* yang akan digunakan untuk pengamanan gambar.
3. Analisa penggunaan *web service* REST

Pada saat menganalisa data, ada beberapa tahap yang harus dilakukan, yaitu mengidentifikasi kebutuhan sistem, fungsi sistem, memodelkan sistem dalam bentuk *flowchart* dan DFD.

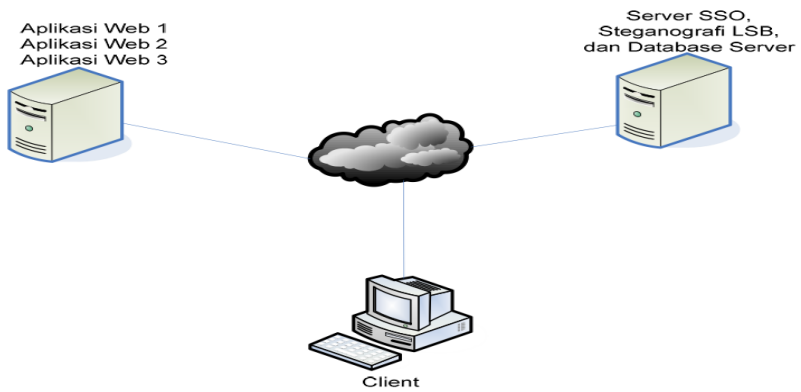
2.4 Perancangan

Setelah tahap analisa selesai maka tahap selanjutnya mulai merancang sistem *Single Sign-On* (SSO). Pada tahap perancangan ini hal yang dilakukan adalah arsitektur sistem *Single Sign-On* (SSO), membangun antarmuka sistem *Single Sign-On* (SSO) serta perancangan *database.*

3. Hasil dan Pembahasan

Masalah utama dari penelitian ini adalah bagaimana membangun sebuah sistem yang dapat melakukan *login* hanya sekali pada banyak aplikasi *web* menggunakan otentikasi gambar. Pada tahap berikutnya akan dibahas deskripsi sistem *single sign-on* (SSO*)* yang akan dibangun, analisis sistem *single sign-on* (SSO) dan bagaimana proses *login* sistem *single sign-on* (SSO) serta analisis penyisipan gambar dengan *least significant bit*.

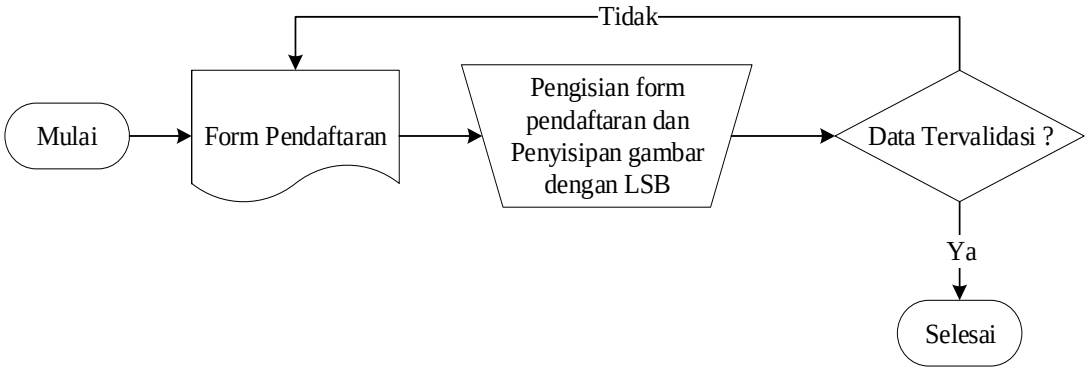
3.1 Deskripsi Sistem *Single Sign-On* (SSO)



Gambar 2. Deskripsi Sistem *Single Sign-On* (SSO) [4]

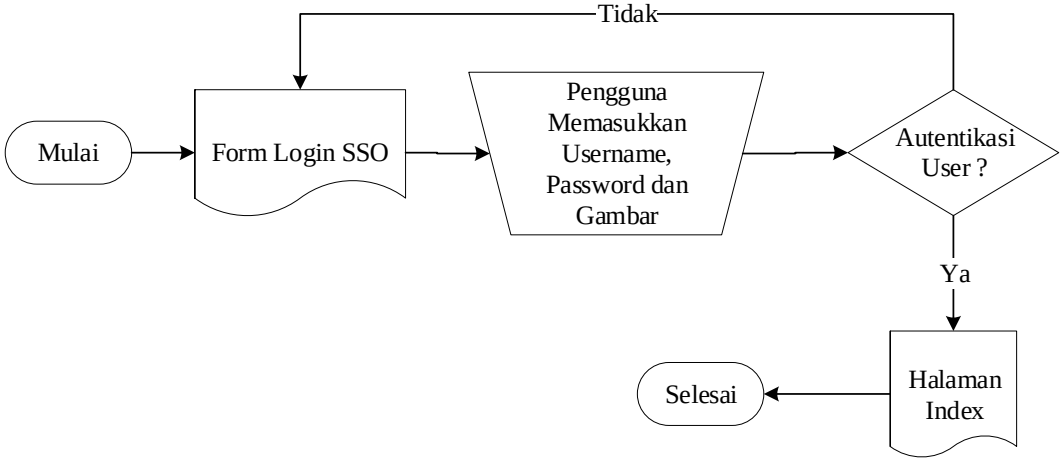
Pada gambar 2 merupakan deskripsi sistem *single sign-on* yang akan dibangun. Topologi sistem *single sign-on* (SSO) terdapat tiga bagian yaitu (1) *Server* sistem *single sign-on* (SSO) berfungsi menyediakan layanan autentikasi kepada pengguna yang membutuhkan autentikasi melalui aplikasi *web*. Pada sistem *single sign-on* (SSO) juga terdapat autentikasi tambahan yaitu autentikasi dengan gambar. (2) Aplikasi Web 1, Aplikasi Web 2 dan Aplikasi Web 3 merupakan *client* dari *server single sign-on* (SSO) (aplikasi web) yang mana permintaan autentikasi diberikan kepada pengguna dan mengelola seluruh aliran autentikasi pengguna. Selanjutnya *client* SSO memvalidasi SSO sesi dan memperoleh informasi pengguna terkait dengan layanan *web server* SSO dengan menggunakan protokol REST. (3) Client adalah pengguna dari sistem *single sign-on*.

3.2 Analisa Sistem SSO



Gambar 3. Analisa Sistem *Single Sign-On* (SSO)

Gambar 3 merupakan alir sistem pendaftaran pengguna baru, dimana seorang pengguna melakukan pendaftaran terlebih dahulu sebelum dapat mengakses sistem *single sign-on*, agar mendapatkan *account*. Calon *member* mengisi *form* pendaftaran serta melakukan penyisipan gambar, yang akan digunakan untuk autentikasi *login.*



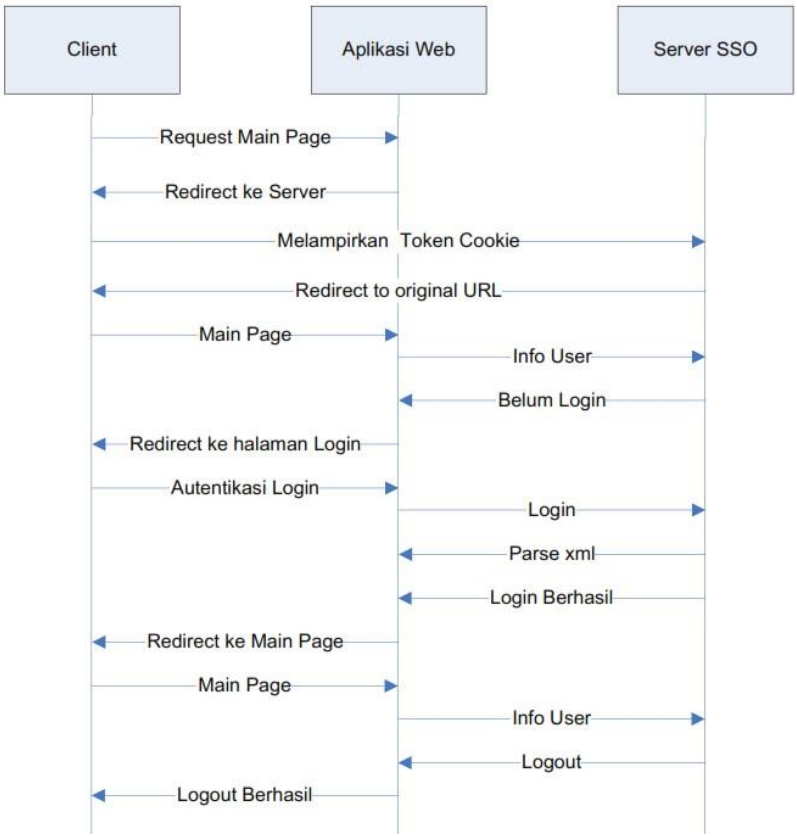
Gambar 4. Diagram Alir *Login* Sistem *Single Sign-On* (SSO)

Gambar 4 merupakan alur sistem *single sign-on* (SSO) yang akan dibangun nantinya. Seorang *member* mengakses sebuah situs *web*, dan melakukan *login*, maka secara otomatis akan dibawa ke halaman Sistem *single sign-on* (SSO) *Server*. *Member* memasukkan *username*,

password dan *file* gambar. Jika data *member* terdaftar pada *server* maka *login* akan berhasil. Jika data *member* tidak terdaftar maka akan dikembalikan lagi ke halaman *login single sign-on* (SSO). Pada proses *login*, autentikasi dengan gambar melakukan perbandingan antara gambar yang ada di *server* dengan gambar yang dimasukkan *member* pada saat *login*. Gambar yang digunakan sebagai autentikasi sebelumnya telah disisipi *text* yaitu menggunakan Metode *Least Significant Bit*

3.3 Analisa Proses Otorisasi *Single Sign-On* (SSO)

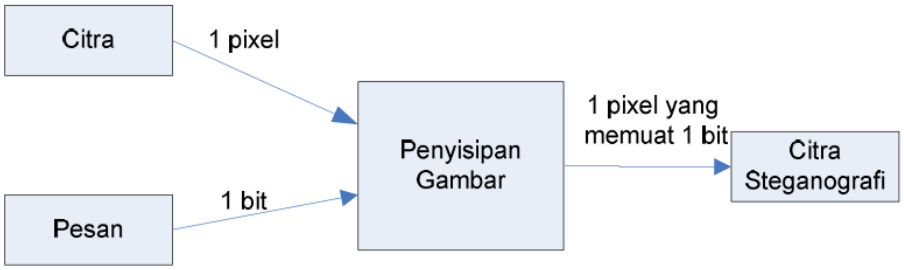
Pada tahapan ini akan dijelaskan bagaimana proses otorisasi pada sistem *single sign-on* (SSO) yang akan dibangun. Pada sistem *single sign-on* (SSO) ini terdapat tiga pihak yaitu : *Client* adalah pengunjung situs *web*, Aplikasi Web adalah situs *web* yang dikunjungi dan *Server single sign-on* (SSO.



Gambar 5. Proses Login Sistem *Single Sign-On* (SSO)

3.4 Analisa Penyisipan Data pada Gambar dengan LSB

Tahap *embedding* atau penyisipan gambar, dimulai dengan mengubah citra digital yang tersusun atas *pixel* dengan menggunakan sinyal RGB, Kemudian bilangan *biner* dalam setiap *pixel* diambil bit rendah yaitu 1 *bit* terakhir dari sinyal *Blue*. Untuk pesan rahasia bertipe karakter akan diubah menjadi bilangan desimal yang kemudian akan diubah menjadi *bit* pesan (bilangan biner). Penyisipan data atau pesan dilakukan dengan mengganti setiap *bit* rendah pada gambar dengan *bit* pesan. Jika *bit* rendah kurang dari *bit* pesan, maka *raster* ditambah 1. Jika *bit* rendah lebih besar dari *bit* pesan, maka raster data dikurang 1. Jika *bit* rendah sama dengan *bit* pesan, maka *raster* tidak dirubah. Setelah itu dilakukan penyusunan kembali *pixel* yang sebelumnya telah disisipi *bit* pesan sesuai dengan *raster* data.



Gambar 6 Metode Penyisipan Gambar

Untuk membentuk tahap *embedding* di perlukan sebuah fungsi sehingga data yang di peroleh dapat disisipkan kedalam *file* pembawa. Fungsi yang digunakan yaitu fungsi *write.*

```
Algortima Least Significant Bit
Function write (data)
   Bits = ascii to biner
   Lenbit = strlen (bits)
     nx = imagesx (image object)
     ny = imagesy (image object)
  for (x=0, bit=0, x<nx, x++)
  for (y=0, y<ny, y++)
     pix = getcolor(image object, x, y)
  array (R,G,B)
end

function Read
 nx = imagesx(image object)
 ny = imagesy(image object)
  for x = 0, y<nx, x++)
  for y = 0, y<ny, y++)
pix = getcolor(image object, x, y)
data = (pix(R + 1) , (pix(G + 1), (pix(B + 1)
return biner to ascii
end
```

Gambar 7 Algoritma LSB

Dari gambar 7 maka *cover carrier* dapat disisipkan pesan dan menghasilkan gambar steganografi dengan tidak menurunkan tingkat kualitas gambar setelah disisipi, Pada analisa tersebut penggunaan gambar berformat PNG lebih kompatibel, karena perbandingan antara *file* asli dengan *file* setelah di lakukan steganografi, tidak memiliki banyak perubahan. Oleh karena itu format PNG tersebut digunakan sebagai autentikasi sistem *single sign-on* (SSO) pada penelitian ini.

3.5 Analisa Penggunaan Web Service REST

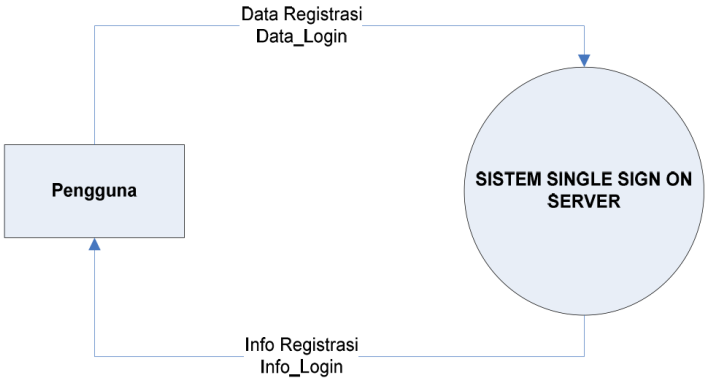
REST (*Representational State Transfer*) *web service* adalah salah satu cara pendistribusian data yang populer saat ini antara *server* dan *client*, dengan menggunakan protokol HTTP. Alasan menggunakan *web service REST* adalah *REST* mudah dipelajari karena aplikasi *web* hanya perlu menggunakan koneksi HTTP ke URL tertentu dan tidak tergantung pada aturan *XML*, *web service REST* lebih digunakan pada *browser* dibandingkan dengan *web service* lainnya. Pada *REST*, metode HTTP yang digunakan diantaranya: (1) POST digunakan untuk membuat sumber daya (*resource*) pada server, (2) GET untuk menerima sumber daya, (3) PUT untuk merubah atau memperbaharui sumber daya, (4) DELETE untuk menghapus sumber daya.

3.6 Deskripsi Fungsional

Aliran informasi ditransformasikan pada saat data bergerak dari input menjadi *output* dapat dilihat di *Context Diagram* dan *Data Flow Diagram* (DFD).

3.6.1 *Context Diagram*

Diagram konteks (*context diagram*) digunakan untuk menggambarkan hubungan *input/output* antara sistem dengan dunia luarnya (kesatuan luar) suatu diagram kontek selalu mengandung satu proses, yang mewakili seluruh sistem. Sistem *single sign-on* (SSO) memiliki entitas yaitu pengguna.

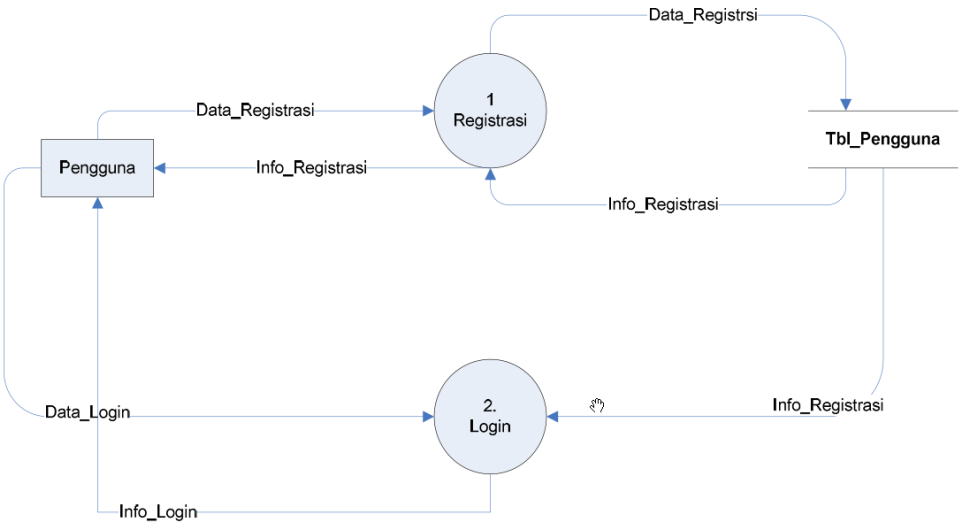


Gambar 7. *Context Diagram*

Entitas luar yang berinteraksi dengan sistem adalah Pengguna yang terdiri dari *Input* Data Registrasi dan *Input* Data *Login*

3.6.2 *Data Flow Diagram*

Data Flow Diagram (DFD) digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika.



Gambar 7. *Data Flow Diagram* (DFD) *Level* 1

Pada gambar 7 merupakan DFD level 1 dari Diagram Kontek diatas yang dipecah menjadi 2 (dua) buah proses beserta aliran datanya. Untuk keterangan masing-masing dapat dilihat pada kamus data pada table 1 berikut ini:

Tabel 1 Keterangan Proses Pada DFD Level 1

No	Nama Proses	Masukan	Keluaran	Deskripsi
1	Registrasi	– *Input* data registrasi	– Status registrasi – Status *login*	Proses untuk melakukan registrasi *account single sign-on*
2	*Login*	– *Input username* – *Input Password* – *Input Autentikasi Gambar*	– Status *login*	Proses untuk melakukan *login* ke dalam sistem *single sign-on*

3.6.3 Perancangan Tabel

Berikut ini deskripsi perancangan tabel dalam *database.*

Tabel Pengguna

Nama : Pengguna
Deskripsi isi : Berisi data Pengguna
Primary key : id

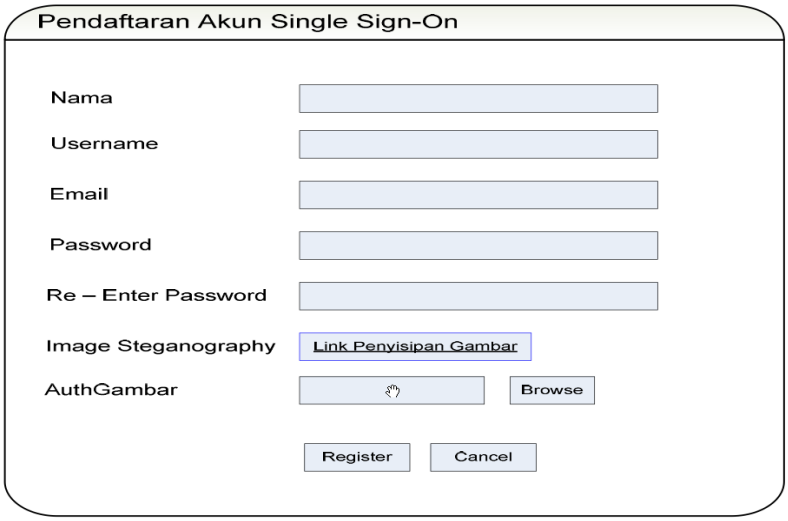
Tabel 2. Pengguna

Nama *Field*	*Type* dan *Length*	Deskripsi	*Null*	*Default*
id	int (10)	ID	No	
nama	Varchar (50)	Nama	No	
username	Varchar (30)	*Username*	No	
password	Varchar (20)	*Password*	No	
email	Varchar (30)	*Email*	No	
image_auth	*Text*	Lokasi *File* Gambar	No	

3.6.4 Perancangan Antar Muka

Pada tahapan ini akan ditampilkan rancangan antar muka sistem *single sign-on* (SSO). Sistem ini memiliki beberapa antar muka yaitu halaman pendaftaran pengguna, halaman *login* sistem *single sign-on* (SSO), halaman penyisipan gambar serta halaman utama atau halaman indeks.

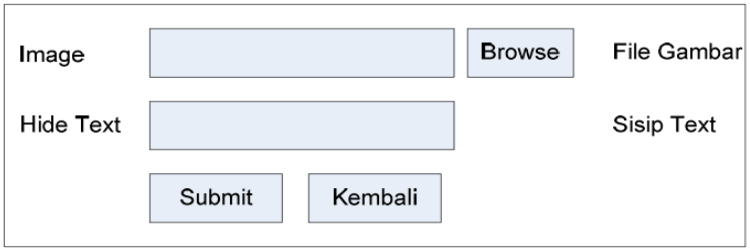
1. **Perancangan Form Registrasi Pengguna**
   Berikut ini adalah perancangan *form* registrasi calon *member*, yang terdiri dari Nama, *Username, Email, Password, Re-Enter Password, Image Steganography* merupakan *link* ke *form* steganografi dan AuthGambar.



Gambar 8. Perancangan *Form* Registrasi

2. Perancangan *Form* Steganografi

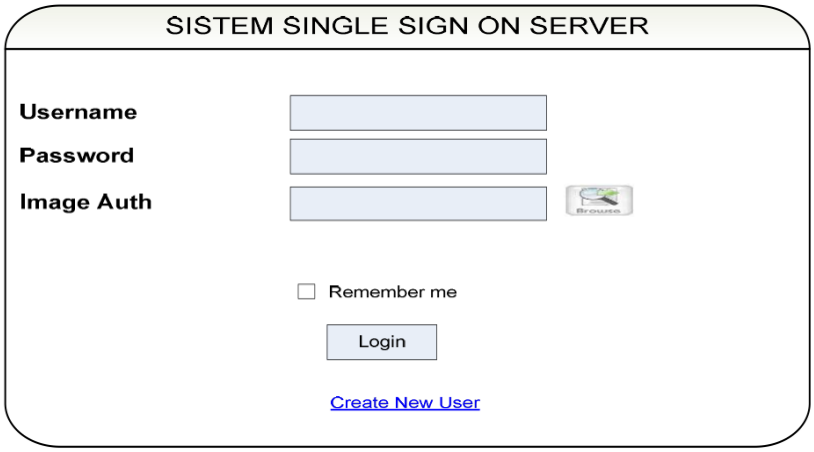
Berikut adalah *form* steganografi, yang terdiri dari *image* atau gambar yang akan di disisipkan sebuah data atau *file*, serta *hide file* yaitu data yang akan disisipkan.



Gambar 9. Perancangan *Form* Steganografi

3. Perancangan *Form Login Single Sign-On* (SSO)

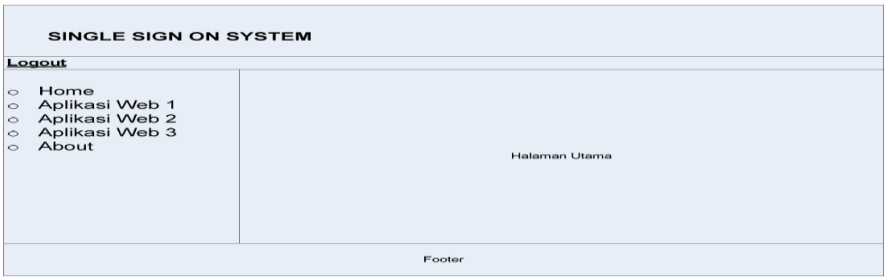
Berikut ini adalah perancangan *form login single sign-on* yang terdiri dari *username*, *password*, serta autentikasi gambar.



Gambar 10. Perancangan *Form Login* SSO

4. Perancangan Halaman Utama

Berikut adalah perancangan halaman utama atau *index*, jadi seorang *member* mengakses berbagai aplikasi *web* dari halaman tersebut. Pada halaman tersebut terdapat beberapa menu, diantaranya adalah *Home*, Aplikasi *Web* 1, Aplikasi *Web* 2, Aplikasi *Web* 3, dan *About.*



Gambar 11. Perancangan Halaman Utama

4. Kesimpulan

Berdasarkan penelitian yang dilakukan telah menghasilkan perancangan sistem *single sign-on* (SSO) dengan menggunakan autentikasi gambar. Perancangan sistem *single sign-on* (SSO) ini memungkinkan bagi pengguna hanya sekali melakukan otentikasi ke dalam beberapa aplikasi web. Perancangan ini juga membuat autentikasi *login* dengan menggunakan gambar dengan metode *least significant bit.* Dengan perancangan ini diharapkan nantinya dapat diimplementasi ke dalam sistem *single sign-on* (SSO).

Daftar Pustaka

[1] Guntoro, "Rancang Bangun Aplikasi Single Sign-on Server Menggunakan Autentikasi Gambar," Universitas Islam Negeri Sultan Syarif Kasim Riau, 2011.
[2] F. Hilmi, R. R. M, and B. Irawan, "Analisis Performansi Authentikasi Single Sign On Pada Web Menggunakan LDAP," vol. 13, no. 2, pp. 93–102, 2012.
[3] G. Ramadhan, "Analisis Teknologi Single Sign On (SSO) dengan Penerapan Central Authentication Service (CAS) Pada Universitas Bina Darma," *J. Mhs. Bina Darma*, vol. XXXIII, no. 2, pp. 1–13, 2012.
[4] N. Heijmink, "Secure Single Sign-On A comparision of protocols," CCV & Radboud University Nijmegen, 2015.
[5] Y. N. Kunang and I. Z. Yadi, "Sistem Single Sign on Universitas Berbasis CAS-LDAP," *Seminar Nasional Inovasi dan Tren (SNIT)*, no. 12, pp. 1–7, 2014.
[6] Nitin, V. K. Sehgal, and D. S. Chauhan, "Image Based Authentication System with Sign-In Seal," 2008.
[7] C. A. Ardagna, F. Frati, and G. Gianini, "Open Source in Web-Based Applications: A Case Study on Single Sign-On," in *Integrated Approaches in Information Technology and Web Engineering: Advancing Organizational Knowledge Sharing*, Hershey - New York: Information Science Reference, 2009, pp. 83–97.
[8] A. Pashalidis and C. J. Mitchell, "A Taxonomy of Single Sign-On Systems," *Inf. Secur. Priv.*, vol. 2727, pp. 249–264, 2003.
[9] Nursyamsi, "Implementasi Sistem Single Sign-on Berbasis Java," Universitas Sumatera Utara, 2009.
[10] Rudy, Riechie, and O. Gunadi, "Integrasi Aplikasi Menggunakan Single Sign on Berbasiskan Lightweight Directory Access Protocol (Ldap) Dalam Portal Binus @Access (Bee-Portal)," *Portal Binus @Access (Bee-Portal)*, pp. 1–7, 2007.