Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

Juan Luis Angel Cano Moreno
Asesorado por la Inga. Wendy Lissette Juárez Marroquín

Guatemala, noviembre de 2014

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

TRABAJO DE GRADUACIÓN

PRESENTADO A JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

JUAN LUIS ANGEL CANO MORENO
ASESORADO POR LA ING. WENDY LISSETTE JUÁREZ MARROQUÍN

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2014

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Narda Lucía Pacay Barrientos
VOCAL V	Br. Walter Rafael Véliz Muñoz
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

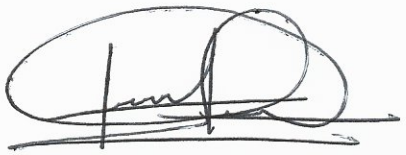
DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Marlon Antonio Perez Türk
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADORA	Inga. Susán Verónica Gudiel Herrera
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha septiembre de 2013.



Juan Luis Angel Cano Moreno

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
SAE/SAP

Guatemala, 01 de Julio de 2014

Ingeniero
Silvio José Rodríguez Serrano
Director de la Unidad de EPS
Facultad de Ingeniería

Estimado Ing. Rodríguez

Por este medio y de la forma más atenta me dirijo a usted para informarle que el estudiante **Juan Luis Angel Cano Moreno,** quien se identifica con el carné universitario ***200511830,*** finalizó de forma satisfactoria el informe final del proyecto "Implementación del Sistema Centralizado de Autenticación y Autorización para las Aplicaciones Web del Centro de Cálculo e Investigación de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala", el cual fue asesorado por mi persona y apruebo para que el estudiante pueda continuar con los trámites correspondientes.

Sin otro particular y agradeciendo de antemano su atención y ayuda, me suscribo,

Atentamente,

Inga. Wendy Lissette Juárez Marroquín
Asesor de Escuela.
No. Colegiado 11338

WENDY LISSETTE JUAREZ MARROQUIN
COL. No. 11338
ING. EN CIENCIAS Y SISTEMAS

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA

FACULTAD DE INGENIERIA

UNIDAD DE EPS

Guatemala, 22 de septiembre de 2014.
REF.EPS.DOC.979.09.2014.

Ing. Silvio José Rodríguez Serrano
Director Unidad de EPS
Facultad de Ingeniería
Presente

Estimado Ingeniero Rodríguez Serrano .

Por este medio atentamente le informo que como Supervisora de la Práctica del Ejercicio Profesional Supervisado, (E.P.S) del estudiante universitario de la Carrera de Ingeniería en Ciencias y Sistemas, **Juan Luis Ángel Cano Moreno** carné No. **200511830** procedí a revisar el informe final, cuyo título es **IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA.**

En tal virtud, **LO DOY POR APROBADO**, solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

"Id y Enseñad a Todos"

Universidad de San Carlos de Guatemala
SUPERVISOR(A) EPS
INGENIERÍA EN CIENCIAS Y SISTEMAS
Unidad de EPS
Facultad de Ingeniería

Inga. Floriza Felipa Ávila Pesquera de Medrano
Supervisora de EPS
Área de Ingeniería en Ciencias y Sistemas

FFAPdM/RA

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA

FACULTAD DE INGENIERIA

UNIDAD DE EPS

Guatemala, 22 de septiembre de 2014.
REF.EPS.D. 525.09.2014.

Ing. Marlon Antonio Pérez Turk
Director Escuela de Ingeniería Ciencias y Sistemas
Facultad de Ingeniería
Presente

Estimado Ingeniero Perez Turk.

Por este medio atentamente le envío el informe final correspondiente a la práctica del Ejercicio Profesional Supervisado, (E.P.S) titulado **IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA,** que fue desarrollado por el estudiante universitario **Juan Luis Ángel Cano Moreno carné No. 200511830** quien fue debidamente asesorado por la Inga. Wendy Lissette Juárez Marroquín y supervisado por la Inga. Floriza Felipa Ávila Pesquera de Medinilla.

Por lo que habiendo cumplido con los objetivos y requisitos de ley del referido trabajo y existiendo la aprobación del mismo por parte de la Asesora y la Supervisora de EPS, en mi calidad de Director apruebo su contenido solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,
"Id y Enseñad a Todos"

Ing. Silvio José Rodríguez Serrano
Director Unidad de EPS

Universidad de San Carlos de Guatemala
DIRECCION
Unidad de Prácticas de Ingeniería y EPS
Facultad de Ingeniería

SJRS/ra

Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 15 de Octubre de 2014

Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación-EPS del estudiante **JUAN LUIS ANGEL CANO MORENO,** carné **2005-11830,** titulado: **"IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA",** y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,

Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación

ESCUELA DE CIENCIAS Y SISTEMAS

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA

FACULTAD DE INGENIERÌA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación ***"IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA",*** *realizado por el estudiante JUAN LUIS ANGEL CANO MORENO, aprueba el presente trabajo y solicita la autorización del mismo.*

"ID Y ENSEÑAD A TODOS"

Ing. Marlon Antonio Pérez Türk
Director, Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 05 de noviembre 2014

Universidad de San Carlos
De Guatemala

Facultad de Ingeniería
Decanato

DTG. 609.2014

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **IMPLEMENTACIÓN DEL SISTEMA CENTRALIZADO DE AUTENTICACIÓN Y AUTORIZACIÓN PARA LAS APLICACIONES WEB DEL CENTRO DE CÁLCULO E INVESTIGACIÓN DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA,** presentado por el estudiante universitario **Juan Luis Angel Cano Moreno,** y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE:

Ing. Alfredo Enrique Beber Aceituno
Decano en Funciones

Guatemala, 6 de noviembre de 2014

/gdech

ACTO QUE DEDICO A:

Mis padres	Por creer siempre en mí, enseñarme los principios y valores que son fundamentales para alcanzar mis metas, además de su amor y cariño que me brindan día a día.
Mis hermanos y hermanas	Por darme su apoyo en todo momento y hacerme sentir su presencia y amor.
Mis amigos y amigas	Por permitirme compartir junto a ellos momentos inolvidables que siempre tendré presentes, y por tener el honor de convivir con ellos como hermanos.
Mis catedráticos	Por todas sus enseñanzas que son el cimiento del conocimiento de muchos profesionales y el mío.
Universidad de San Carlos de Guatemala	Por ser la casa de estudios donde logré alcanzar mi sueño de ser un profesional de bien para mi país, y enseñarme una nueva forma de vida.
Todas las personas que forman parte de mi vida	A quienes que por algún motivo hemos cruzado caminos o palabras, gracias por estar allí; de todas aprendí algo muy valioso.

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por ser el centro donde pude desarrollar todos mis conocimientos, y culminar mi carrera.
Inga. Wendy Lissette Juárez Marroquín	Por apoyarme en la realización de este trabajo de graduación, por guiarme en su desarrollo, motivándome a la realización de un buen trabajo.
Mis amigos y compañeros	Que me apoyaron durante todos mis estudios y que de alguna forma ayudaron a que siguiera adelante.
Mis catedráticos	Por todas sus enseñanzas que son el cimiento del conocimiento de muchos profesionales

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES

FIGURAS

TABLAS

GLOSARIO

AJP	Acrónimo de *Apache Jserv Protocol* es un protocolo binario, que permite enviar solicitudes desde un servidor web a un servidor de aplicaciones que se encuentre detrás del servidor web.
CSS	Hojas de estilo en cascada; tienen, como función principal la de separar la estructura de un documento de la presentación, siendo un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML.
***Firewalls* (cortafuegos)**	Es un sistema de seguridad, que suele ser una combinación de hardware y software, que se utiliza para proteger una red de las amenazas externas procedentes de otra red, incluyendo a internet.
Framework	Definición de estándares de conceptos, prácticas que forman una estructura conceptual y tecnológica con soporte definido.

Hacker	Es una persona dedicada a una tarea de investigación o desarrollo realizando esfuerzos más allá de los normales y convencionales, anteponiéndole un apasionamiento que supera la normal energía.
HTML	Son las siglas de *HyperText Markup Language,* es un lenguaje estándar de marcado, el cuál es utilizado para la creación de páginas web.
Ingeniería social	Manipulación de usuarios para obtener información confidencial como contraseñas.
LDAP	Son las siglas de “Lightweight Directory Access Protocol”, hacen referencia a un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.
MVC	Se refiere a una arquitectura del software basada en la separación del modelo de datos con la vista y que estos se comuniquen mediante un controlador.
OAUTH	Es la abreviación de *Open Authorization,* el cuál es un estándar abierto para la autorización de las credenciales de un usuario en un sistema externo al dominio de seguridad.

REST	Acrónimo de *Representational State Transfer* el cuál es una técnica de Arquitectura de Software para sistemas de hipermedia distribuidos como la *World Wide Web.*
SAML	Es el acrónimo de *Security Assertion Markup Language,* el cuál es un estándard para el intercambio de autenticación y autorización de datos entre dos dominios asegurados.
SSO	Acrónimo en inglés de *single sign on*. Es un procedimiento de autenticación que habilita al usuario para acceder a varios sistemas con una sola instancia de identificación.
WAR	Es un tipo de archivo que empaqueta aplicaciones Java, el cuál contiene *JavaServer Pages, servlets*, clases de Java, archivos XML, entre otros archivos que contiene una aplicación web de Java.
Web	Es un vocablo inglés que significa red, telaraña o malla. El concepto se utiliza en el ámbito tecnológico para nombrar a una red informática y, en general, a internet (en este caso, suele escribirse como Web, con la W mayúscula).

RESUMEN

A medida que los sistemas informáticos proliferan para soportar los procesos del negocio, tanto los usuarios, como administradores de sistemas se enfrentan a una tarea complicada para completar las funciones laborales. Los usuarios típicamente se tienen que autenticar en múltiples sistemas, necesitando una pantalla de autenticación por cada uno de los sistemas, esto podría involucrar usuarios y contraseñas distintas, mientras que los administradores de sistemas se enfrentan a la tarea de estar administrando las cuentas de los usuarios en cada uno de estos sistemas, y de estarlos coordinando para que la información sea consistente e integra de acuerdo a las políticas de seguridad de la organización.

El Centro de Cálculo e Investigación Educativa se estaba enfrentando a esta problemática, por lo que se detectó la oportunidad de mejora en la implementación de un sistema que permita la unificación de usuarios de los distintos aplicativos informáticos que son administrados por la institución y que al mismo tiempo les permitiera escalar en algún futuro a tecnologías que son manejadas a través de internet, como lo es, el sistema de inicio de sesión de Google.

El proyecto consistió en la implementación de un sistema de autenticación único (SSO) que le permite a los usuarios iniciar sesión en un sistema centralizador y que es independiente de la aplicación. De esta forma se hizo transparente la comunicación entre los sistemas informáticos que el usuario utiliza para realizar las labores diarias.

El Ejercicio Profesional Supervisado fue implementado en el Centro de Cálculo e Investigación Educativa de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala. Dicha dependencia está encargada de la elaboración de software y componentes tecnológicos que permiten realizar las operaciones diarias de la Facultad de Ingeniería. Por lo que la implementación de este sistema fue de gran utilidad a la institución.

OBJETIVOS

General

Realizar la implementación de una solución informática que permita a los usuarios iniciar sesión de forma centralizada, consistente, transparente y segura a los sistemas del *dashboard*, expedientes, horarios y reportes. También debe permitir de forma fácil y escalable la integración de nuevos sistemas.

Específicos

1. Disminuir el tiempo de construcción de los sistemas ya que los programadores no invertirán tiempo en el módulo de inicio de sesión, permitiéndo enfocarse en los requerimientos del negocio.

2. Realizar la migración de la base de datos relacional a la base de datos de Ldap para el manejo centralizado de usuarios.

3. Realizar la migración del sistema del *dashboard,* expedientes, horarios y reportes.

4. Permitir la escalabilidad de las aplicaciones al integrarse al sistema de autenticación centralizada.

5. Crear una plataforma escalable y segura de autenticación que soporte los requerimientos del departamento.

INTRODUCCIÓN

En el presente documento se describe el trabajo realizado durante el proceso de EPS que incluye la justificación, objetivos, los recursos necesarios y presupuesto del proyecto titulado Implementación del sistema centralizado de autenticación y autorización para las aplicaciones web del Centro de Cálculo e Investigación Educativa de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala.

El EPS fue implementado en el Centro de Cálculo e Investigación Educativa de la Facultad de Ingeniería. El departamento está encargado de la elaboración del software y componentes tecnológicos de la Facultad, por lo que a medida que se desarrollan nuevas aplicaciones se ha hecho notable la problemática de la integración entre los sistemas en el aspecto de autenticación.

La implementación del sistema de autenticación y autorización para las aplicaciones web permite a los programadores enfocarse más en los requerimientos del negocio, debido a que el componente de autenticación es un componente transversal a los sistemas, de esta forma el proceso de desarrollo será más ágil por lo que tendrán mejoras en calidad de software y en tiempo de desarrollo.

El sistema centralizado de autenticación maneja estándares de seguridad por lo que es una aplicación confiable que permite la escalabilidad de las aplicaciones web desarrolladas en el Centro de Cálculo e Investigación Educativa de la Facultad de Ingeniería.

1. SEGURIDAD LÓGICA

La seguridad lógica le permite a los sistemas informáticos prevenir el ingreso de personas no autorizadas. Esta establece las normas que minimizan los riesgos que puede sufrir la información dentro de un sistema. Estas normas incluyen:

- Horarios de funcionamiento
- Restricciones a ciertos recursos
- Autorizaciones
- Denegaciones
- Perfiles de usuarios
- Planes de emergencia
- Protocolos

La seguridad lógica está concebida para proteger los activos informáticos de una organización, dentro de estos se encuentran:

- La infraestructura computacional: es la parte fundamental de un sistema computacional. En esta se almacena la información de la organización y permite el correcto funcionamiento de los sistemas que hacen uso de la misma.
- Los usuarios: son las personas que hacen uso de la información almacenada en un sistema informático, la gestionan y la modifican. Debe de protegerse el sistema informático para que el uso por parte de ellos no ponga en entre dicho la seguridad de la información.

- La información: es el principal activo de un sistema informático, reside en la infraestructura de computación y es utilizada por los usuarios.

La seguridad lógica utiliza la autenticación para determinar el usuario que ha accedido a un sistema y la autorización para determinar que reglas se deben de aplicar a este usuario.

1.1. Autenticación

En informática se le denomina autenticación al proceso que realiza un usuario para identificarse dentro de un sistema.

Para que un usuario se pueda autenticar en un sistema este debe cumplir con cualquier de los siguientes factores:

- Factor de conocimiento: indica que el usuario debe de conocer la respuesta de una pregunta, por lo general este factor es el que se observa en la mayoría de sistemas informáticos, en el cuál se le pregunta al usuario cuáles son las credenciales de acceso, por lo general conformados por un identificador de usuario y una contraseña que el usuario debe de conocer.
- Factor de posesión: indica que el usuario debe de poseer un objeto que le permita identificarse dentro de un sistema, este objeto puede ser una tarjeta de identificación.
- Factor de herencia: utiliza pertenencias del usuario, como por ejemplo las huellas dactilares, reconocimiento de rostro o DNA.

1.2. Autorización

En informática, la autorización es una parte del sistema que le permite proteger los recursos de información para que no puedan ser accedidos por aquellos usuarios a los que no se les ha otorgado los permisos necesarios. Los recursos incluyen archivos y otros objetos de datos, programas, dispositivos y funcionalidades de un sistema.

La autorización ocurre posterior a que el usuario ya ha sido autenticado o identificado dentro del sistema y es utilizado para decidir si el usuario X tiene permiso para acceder al dato, funcionalidad o servicio Y.

2. ANÁLISIS Y DISEÑO DE LA IMPLEMENTACIÓN

El proyecto consistió en la implementación de un sistema de autenticación único, que permite a las aplicaciones desarrolladas en el Centro de Cálculo e Investigación Educativa de la Facultad de Ingeniería autenticarse de forma segura, delegando el proceso de autenticación a un tercer sistema que maneja las sesiones de usuarios, las variables del contexto y la integración en con otros sistemas.

Para esto fue necesario la implementación de un servidor de SSO, la instalación de los agentes de seguridad, la carga de usuarios a al sistema centralizado especializado en el manejo de usuarios, la migración de las aplicaciones estipuladas por esta dependencia para la validación del correcto funcionamiento del sistema y la capacitación a los desarrolladores para la utilización del sistema y buenas prácticas en la utilización del sistema.

2.1. Definición de productos y tareas

Los productos y tareas realizadas para el correcto funcionamiento de la plataforma fueron los siguientes:

Productos:

- ETL para la migración y adecuación de los usuarios en el modelo relación al servidor de LDAP.
- Diagrama de despliegue de los servidores
- Manual de instalación de LDAP

- Manual de instalación de OpenAm
- Servicios Restful y Restless para facilitar la recuperación de roles y usuarios.

Tareas:

- Instalación y configuración del servidor del SSO
- Instalación y configuración del agente de políticas de SSO para un servidor Java.
- Instalación y configuración del agente de políticas de SSO para un servidor Apache.
- Capacitación de usuarios para la utilización de la plataforma de SSO (OpenAm).
- Migración del *dashboard* a la nueva plataforma
- Migración del sistema de reportes a la nueva plataforma
- Migración del sistema de horarios a la nueva plataforma

2.2. Recursos

Los recursos utilizados durante el período en el que fue realizado el EPS son:

2.2.1. Recursos humanos

- Estudiante que desarrolló el Ejercicio Profesional Supervisado
- Asesor elegido por el estudiante que realizó el Ejercicio Profesional Supervisado.
- Asesor encargado del proyecto en la institución
- Desarrollador encargado del proyecto por parte de la institución

2.2.2. Recursos materiales

- Servidores que fueron provistos por el Centro de Cálculo e Investigación Educativa para la implementación del Ejercicio Profesional Supervisado.
- Computadoras personales del estudiante que desarrolló el Ejercicio Profesional Supervisado.
- Papel y tinta de impresora

2.3. Presupuesto

Se llama presupuesto al cálculo y negociación anticipado de los ingresos y egresos de una actividad económica durante un período, por lo general en forma anual. Es un plan de acción dirigido a cumplir una meta prevista.

2.3.1. Costos

Es el gasto económico que representa la fabricación de un producto o la prestación de un servicio. Dicho en otras palabras, el costo es el esfuerzo económico.

Tabla I. **Costo de la implementación**

Recursos	Cantidad	Costo Unitario (6 meses)	Subtotal
Estudiantes de Ejercicio Profesional Supervisado	1	Q 48 000	Q 48 000
Asesor de EPS	1	Q 24 000	Q 24 000
Electricidad		Q 900	Q 900
Suministros (papel y tinta de impresora)		Q 500	Q 500
		Total	Q 73 400

Fuente: elaboración propia.

2.3.2. Beneficios

- Costo de la planificación del proyecto (beneficios directos para el personal de la Facultad de Ingeniería). El costo de la planificación por parte del estudiante hacia la institución donde se prestó el servicio tuvo un costo estimado de: Q 48 000,00.

- Beneficio a los programadores, pues estos ya no deben crear el módulo de autenticación por cada una de las aplicaciones que tengan a cargo, esto disminuye el tiempo de programación en aproximadamente una semana.

- Los usuarios tienen como beneficio la autenticación centralizada en un solo sistema, lo que permite que el usuario únicamente tenga que memorizar una contraseña y con esta pueda acceder a todos los sistemas de la institución.

- La aplicación permite una integración con todas las aplicaciones web que maneja esta dependencia, lo que permite la facilidad de uso en los redireccionamientos entre aplicaciones, pues para el usuario es un solo sistema al que está accediendo.

2.4. Descripción preliminar de la solución

Para la elaboración del proyecto se seleccionaron un conjunto de herramientas *OpenSource* que cumplen con la funcionalidad de proveer una plataforma de SSO. Las herramientas que se investigaron para la elaboración del proyecto se describen a continuación:

Tabla II. **Herramientas de *SSO***

Herramienta	**Vendedor**	**Soporta Java EE**	**Soporta PHP**	**Licencia**	**Soporta OAuth 2.0**
Account & SSO	Nokia, Intel	NO	NO	GNU LPGL 2.1	NO
Distributed Access Control System(DACS)	*Distributed System Software*	SI	SI	*Apache License 2.0*	NO
Enterprise Sign On Engine	*Queenland University*	SI	SI	Apache License 2.0	NO
JBoss SSO	JBoss	SI	SI	LGPL	NO
Java Open Single Sign On	*JOSSO*	SI	NO	LGPL	NO
OpenAM	ForgeRock	SI	SI	CDDL	SI
Persona	Mozilla	SI	NO	MPL	SI

Fuente: elaboración propia.

De las posibles soluciones encontradas en el mercado se seleccionaron tres que por las características se acoplan mejor a la arquitectura que utiliza el Centro de Cálculo e Investigación Educativa.

2.4.1. JBoss SSO

JBoss SSO es un producto desarrollado por JBoss que permite la autenticación única y centralizada, está elaborado en Java y es multiplataforma. Utiliza una licencia LGPL la cual permitiría la utilización libre para los fines de la institución.

Dentro de las características principales tiene:

- Interacción entre aplicaciones que utilizan el standard *SAML*

- Una aproximación descentralizada que permite tener de forma transversal el sistema de autenticación a los demás sistemas del negocio.
- Interfaz adaptable a otros productos de *JBoss*, como lo sería *JBoss* Portal.

La última versión estable desarrollada de esta aplicación data del 1 de noviembre de 2006, lo que indica que esta herramienta se encuentra desactualizada y sin actividad en la comunidad virtual. Esto es un riesgo a tomar en consideración al momento de tomar esta herramienta como una solución para la plataforma.

2.4.2. OpenAM

OpenAM es una plataforma de SSO que permite la autenticación única y centralizada, desarrollado en java y mantenida por *ForgeRock*. *OpenAM* es el sucesor de *OpenSSO*, una herramienta que era mantenida por *Sun Microsystems* y que fue descontinuado por Oracle, cuando este compro *Sun Microsystems*.

OpenAM es un proyecto activo y que tiene soporte por *ForgeRock*, el último *release* del proyecto en fue en diciembre del 2013 y permite la integración de distintas plataformas de programación en un único sistema centralizado de autenticación, que permite la correcta coherencia entre los sistemas del negocio.

Asímismo, OpenAM suporta 20 métodos de autenticación y tiene la flexibilidad para encadenar estos métodos, permitiendo autenticarse de distintas formas a los mismos sistemas.

También permite la alta disponibilidad y la federación de la autenticación manejando otros estándares como lo son: *OAuth2, Fedlet, OpenID Connect,* entre otros. *ForgeRock* también provee otros sistemas como lo es *OpenIDM* y *OpenDJ*, para la la gestión de usuarios y para el almacenamiento de los usuarios respectivamente. Lo cuales al ser implementados en conjuntos crea un ambiente integrado, desde la creación de usuarios, almacenamiento y autenticación de los mismos de una forma coherente e integra.

2.4.3. Enterprise Sign On

Es una plataforma OpenSource para el control de accesos y federación. Originalmente fue construido por la Universidad de Queensland y subsecuentemente se hizo disponible bajo la licencia de código Apache 2.0.

Permite la integración con sistema de almacenamiento de usuarios como LDAP y el *Active Directory de Microsoft.*

2.5. Selección de herramienta

La herramienta seleccionada para crear la plataforma de autenticación única fue OpenAM, debido a que es una plataforma que se encuentra activa, soporta los estándares más recientes en el ámbito de las plataformas de *SSO,* permite una gran flexibilidad entre las plataformas que se pueden integrar a la herramienta, permite la escalabilidad de la plataforma y se soporta las plataformas de desarrollo con las cuáles se trabajan actualmente en el Centro de Cálculo e Investigación Educativa.

2.6. Casos de uso

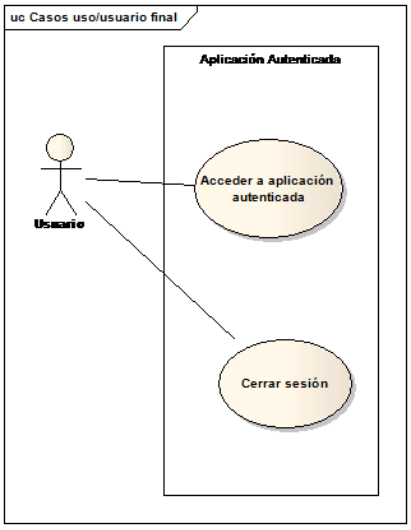
Se detallan los casos de usos relacionados con la implementación del sistema centralizado de autenticación para las aplicaciones web desde las siguientes perspectivas:

- Perspectiva del usuario final
- Perspectiva del arquitecto de software
- Perspectiva del administrador de usuarios

2.6.1. Perspectiva usuario final

Este únicamente se relación con la plataforma de SSO al momento de autenticarse en una aplicación asegurada por la plataforma.

Figura 1. **Caso de uso, perspectiva del usuario final**



Fuente: elaboración propia.

Tabla III. **Caso de uso, acceder a aplicación autenticada**

No.	1
Nombre	Acceder a aplicación autenticada
Actores	Usuario final
Propósito	Entrar a las aplicaciones
Resumen	El usuario entre a la aplicación y se autentica parar realizar las tareas diarias
Tipo	Primario y esencial
Cursos normal de los eventos; acción del usuario	**Respuesta del sistema**
Cuando el usuario ingresa a las aplicaciones web	Muestra la pantalla de autenticación en la cual puede ingresar las credenciales de acceso.
Hace click sobre el botón de autenticación	Consulta en el servidor de *OpenAM* si el ya iniciado sesión.
	Solicita nombre de usuario y contraseña
Introduce el nombre de usuario, contraseña y presiona el botón de inicio de sesión	Se revisan los usuarios en el servidor de *OpenLdap* y este devuelve si los datos son correctos.
	Muestra el recurso solicitado por el usuario.
Cursos alternos: Línea 2: el usuario ya ha iniciado sesión en *OpenAM,* continúa en el paso 8. Línea 4:el usuario no tiene registrado una cuenta en *OpenLdap*. No permite el ingreso.	

Fuente: elaboración propia.

Tabla IV. **Caso de uso, cerrar sesión**

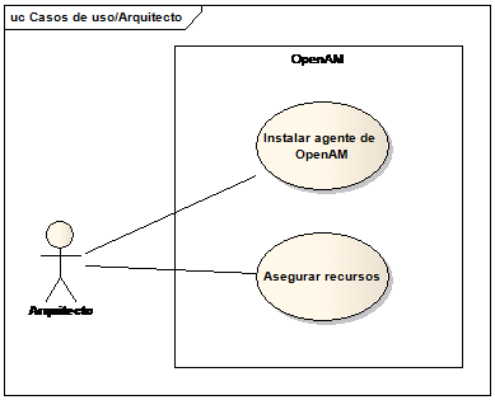
No.	2
Nombre	Cerrar sesión
Actores	Usuario final
Propósito	Salir de las aplicaciones autenticadas
Resumen	El usuario sale de la aplicación y esta cierra la sesión
Tipo	Primario y esencial
Cursos normal de los eventos; acción del usuario	**Respuesta del sistema**
Cuando el usuario presiona el *link* de Cerrar sesión	Solicita cerrar la sesión en el servidor de *OpenAM*
	Muestra la página de inicio de sesión de *OpenAM*

Fuente: elaboración propia.

2.6.2. Perspectiva del arquitecto de software

Desde la perspectiva del arquitecto de software es importante asegurar recursos y poder agregar agentes de seguridad de *OpenAM*.

Figura 2. **Caso de uso, perspectiva del arquitecto de software**



Fuente: elaboración propia.

Tabla V. **Caso de uso, instalar agente de *OpenAM***

No.	3
Nombre	Instalar agente de *OpenAM*
Actores	Arquitecto
Propósito	Instalar un agente de seguridad de *OpenAM* en un servidor *Web*
Resumen	El usuario ingresa al instalador de agentes e instala el agente en el servidor *Web*
Tipo	Primario y esencial
Cursos normal de los eventos; acción del usuario	**Respuesta del sistema**
Cuando el usuario ejecuta el instalador del agente de seguridad	Muestra la pantalla en la cual solicita los datos del agente de seguridad: • Host del servidor de *OpenAM* • Contraseña del agente de seguridad • Puerto del agente de seguridad
Ingresa los datos solicitados	Muestra el resultado de la instalación

Fuente: elaboración propia.

Tabla VI. **Caso de uso, asegurar recursos**

No.	4
Nombre	Asegurar recursos
Actores	Arquitecto
Propósito	Asegurar un recurso web para que el acceso sea autorizado por la herramienta de *OpenAM*
Resumen	El usuario ingresa a las configuraciones del agente de seguridad y lo actualiza para que pueda asegurar el acceso a un recurso.
Tipo	Primario y esencial

Cursos normal de los eventos; acción del usuario	**Respuesta del sistema**
Cuando el usuario ingresa a la archivo de configuración del agente de seguridad.	
Lo edita para agregar los recursos autenticados	

Fuente: elaboración propia.

3. IMPLEMENTACIÓN

OpenAM es una herramienta *OpenSource* que permite implementar la autenticación centralizada mediante agentes de seguridad, estos deben de ser instalados en cada uno de los servidores web y contenedores de aplicaciones donde se requiere la autenticación centralizada.

3.1. Flujo de autenticación

El flujo de autenticación de la herramienta *OpenAM* se describe dependiendo si la aplicación debe de ser autenticada por el *dashboard*, sistema legado que permite un único punto de autenticación, pero que por escalabilidad se necesita pasar a una plataforma estándar de autenticación.

3.2. Aplicaciones en el *dashboard*

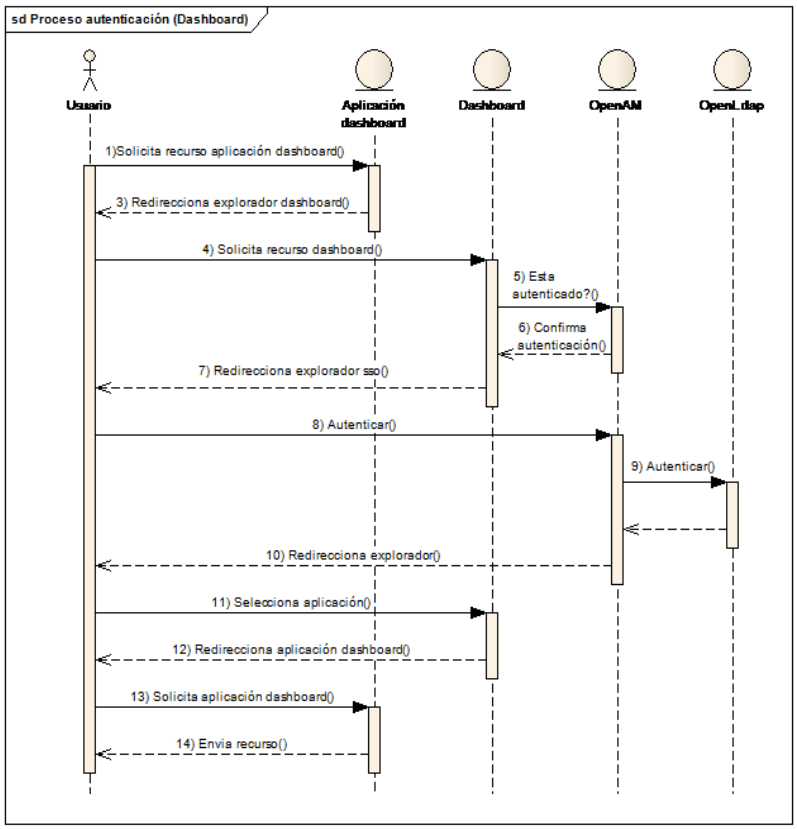
Para las aplicaciones del *dashboard* el flujo de autenticación sigue la siguiente lógica:

- El usuario solicita un recurso a una aplicación del *dashboard*
- La aplicación de *dashboard* navega a la página de autenticación del *dashboard*.
- El agente de SSO valida el estado de la sesión, si el usuario no tiene una sesión valida, el agente redirecciona a la página de autenticación del SSO.
- El usuario ingresa las credenciales en la página de autenticación

- Al ser autenticado correctamente, SSO redirecciona al explorador a la página al *dashboard.*
- En el *dashboard* se selecciona el rol y la unidad para el sitio al que se desea conectar.
- El *dashboard* almacena la información de autenticación y la bitácora de la sesión del usuario.
- El *dashboard* envía al explorador a la página seleccionada

El flujo seguido por la aplicación se ilustra en la figura a continuación:

Figura 3. **Proceso de autenticación para una aplicación del *dashboard***



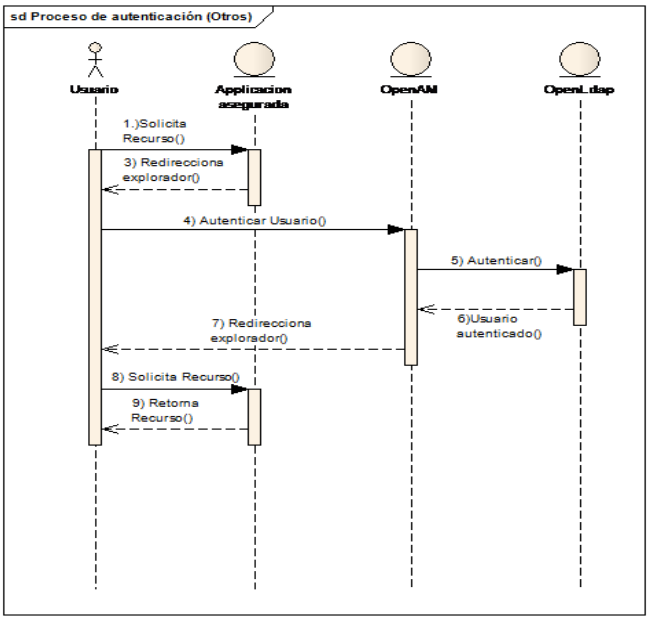
Fuente: elaboración propia, con programa Photoshop.

3.3. Autenticación para otras aplicaciones *web*

Las aplicaciones que no se deben de autenticar por medio del *dashboard* siguen un flujo más simple, el cual se describe de la siguiente manera:

- El usuario solicita un recurso de una aplicación asegurada por SSO
- El agente de SSO valida la sesión del usuario
- El agente de SSO envía el navegador a la página de autenticación
- El usuario ingresa sus credenciales en la página y se autentica
- OpenAM valida las credenciales en OpenLDAP
- OpenLDAP acepta la autenticación
- El explorador del usuario solicita el recurso asegurado
- La aplicación asegurada provee el recurso solicitado

Figura 4. **Diagrama de secuencia, otras aplicaciones**



Fuente: elaboración propia, con programa Photoshop

3.4. Componentes

Los componentes necesarios para realizar la implementación del sistema de SSO son:

3.4.1. OpenLDAP

OpenLDAP es un sistema *OpenSource* que implementa un protocolo ligero de acceso a directorios, por las siglas en inglés. Permite el almacenamiento de los usuarios en un sistema centralizado y especializado para el control de usuarios en una estructura de directorios. Asímismo, permite una alta gama de mecanismos de encriptación para las contraseñas de los usuarios, lo que permitió hacer correctamente la migración de usuarios de la base de datos relacional a este sistema.

3.4.2. OpenAM

Es el sistema que integra los distintos componentes de la plataforma de SSO, permite almacenar información de la sesión del usuario autenticado de manera segura y escalable, creando una interfaz consistente entre las aplicaciones desarrolladas por el Centro de Cálculo e Investigación Educativa, Asimismo, gestiona la administración de las sesiones de usuario, roles y agentes de SSO.

3.4.3. Agente de OpenAM para Apache

El agente de OpenAM para apache permite administrar las políticas de seguridad en un servidor de páginas web Apache, está encargado de gestionar

las páginas aseguradas y de mantener de forma consistente la sesión del usuario en el servidor donde ha sido instalado.

3.4.4. Agente de OpenAM para Java EE

El agente de OpenAM para Java EE permite asegurar las aplicaciones desplegadas en un contenedor de aplicaciones empresariales de Java, este es instalado en el contenedor de aplicaciones y permite la administración de las páginas aseguradas a través de configuraciones básicas en el descriptor de la aplicación.

3.4.5. Dashboard

Sistema legado del Centro de Cálculo e Investigación Educativa permite la unificación de aplicaciones PHP en una misma interfaz, asimismo, permite la selección de roles y unidades organizativas a través de una interfaz consiste e intuitiva para el usuario final.

3.4.6. Expedientes

El sistema de expedientes fue seleccionado para mostrar la escalabilidad de la implementación haciendo que este utilice la seguridad a través de OpenAM. Este sistema está desarrollado en Java y utilizaba una seguridad hecha en casa, la cual no manejaba ningún estándar de seguridad y no estaba integrada a ningún sistema único de autenticación. Por lo que esta aplicación se migró a la nueva tecnología permitiendo tener una interfaz más coherente entre las aplicaciones que el usuario debe acceder.

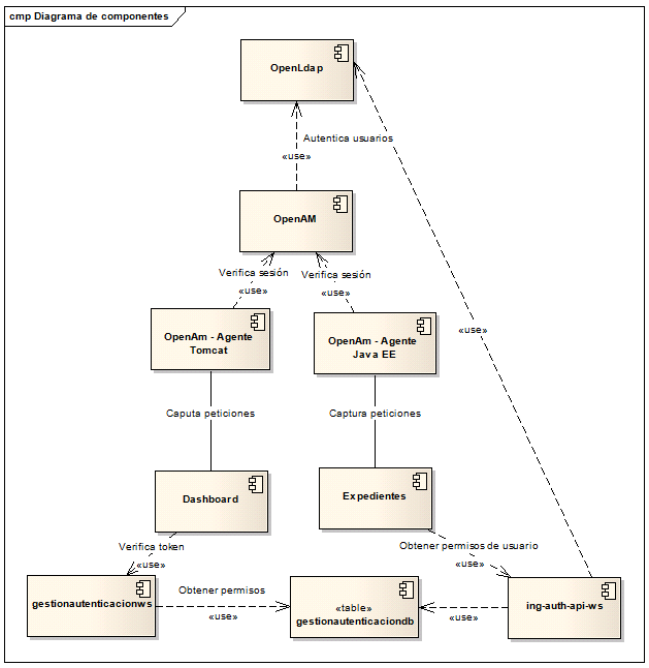
3.4.7. Gestionautenticacionws

Este es un sistema que permite el acceso a los datos que están almacenados en una base de datos relacional e implementa un *webservice* que permite la interacción con los métodos provistos por este sistema.

3.4.8. Ing-auth-api-ws

Este webservice permite el acceso a los roles del usuarios y permite la interacción entre la sincronización de la base de datos relacional con los datos almacenados en *OpenLdap*. Permitiendo tener consistencia en ambos almacenes de datos.

Figura 5. **Diagrama de componentes**



Fuente: elaboración propia, con programa Photoshop.

4. DESPLIEGUE EN PRODUCCIÓN

Para realizar la implementación correcta de la plataforma de autenticación única fue necesaria la instalación de los siguientes componentes, tanto en servidores nuevos como en los servidores existentes de producción.

4.1. Servidor *OpenAM*

En el servidor al que se le denomina Servidor *OpenA* se realizó la instalación de un contenedor de aplicaciones *Apache Tomcat,* en el cuál se hizo la instalación del sistema de *OpenAM*, este mismo se configuró para que se comunicara con el servidor de *OpenLDAP* para la administración de usuarios y autenticación por medio de los algoritmos que el servidor de *OpenLdap* soporta, en este caso se utilizó, la encriptación de la contraseña en MD5.

4.2. Servidor *OpenLdap*

En el servidor de *OpenLdap* se realizó la instalación del sistema *OpenLdap*, el cual actualmente almacena los usuarios y credenciales de acceso.

4.3. Servidor de producción apache

En el servidor de producción de apache se hizo la instalación y configuración del agente de seguridad de *OpenAM*, el cual se configuró para que aseguraran las páginas de las aplicaciones del *dashboard* y para que redireccionara a las páginas apropiadas de autenticación y de cierre de sesión.

Asímismo, se realizó la migración de los recursos necesarios para soportar la autenticación por medio de *OpenAM*.

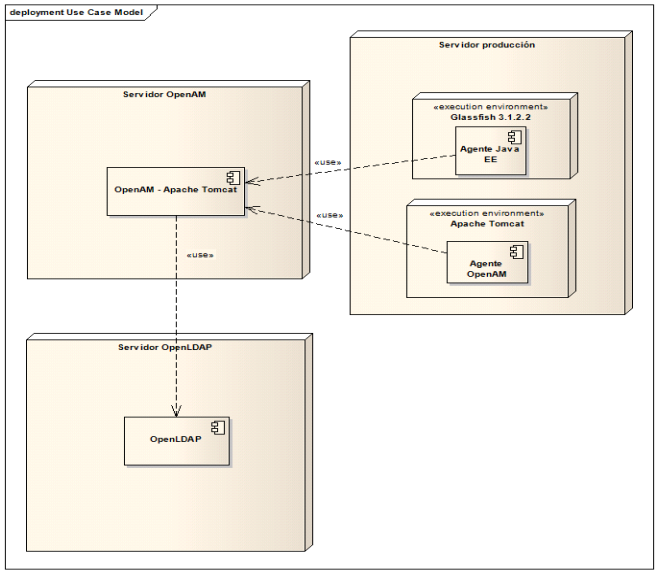
4.4. Servidor de producción Glassfish 3.1.2.2

En el servidor de producción de *Glassfish* se realizó la instalación del agente de seguridad para de *OpenAM* para que pudiera administrar la seguridad para las aplicaciones Java EE.

4.5. Diagrama de despliegue

A continuación se presenta el diagrama de despliegue, en el cual se muestran cada uno de los servidores que interactúan en la implementación de la plataforma de autenticación única.

Figura 6. **Diagrama de despliegue**



Fuente: elaboración propia, con programa Photoshop.

4.6. Instalación

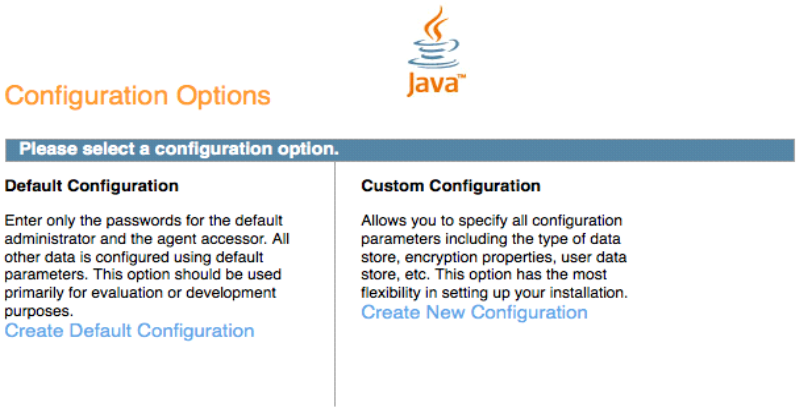
A continuación se detallan los pasos de configuración que se deben realizar para hacer la correcta instalación de la plataforma de autenticación única, utilizando como núcleo del sistema *OpenAM* como gestor de la sesión y manejo de la seguridad y OpenLDAP como administrador de los usuarios del sistema.

4.6.1. Instalación del servidor de *OpenAM*

Para realizar la instalación de *OpenAM* es necesario instalar *Apache Tomcat* y desplegar el *war* de *OpenAM* en el servidor. Posterior a esto se deben seguir los siguientes pasos:

- Ingresar a la página http://<host>:8080/opensso/, esto mostrará la página que se muestra en la figura 6.

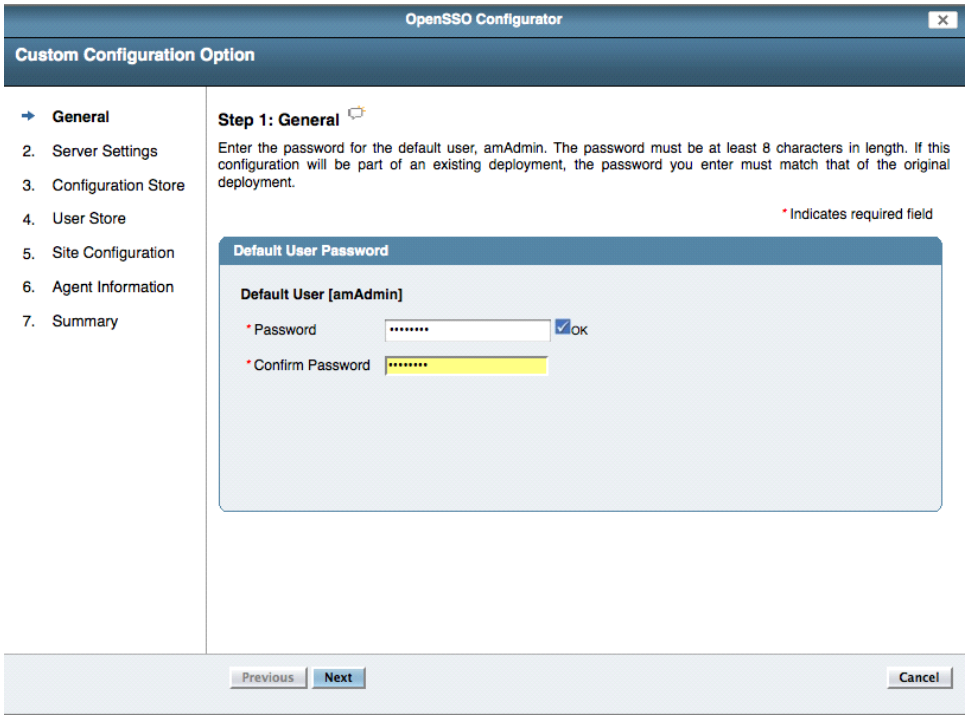
Figura 7. **Página inicial de instalación de *OpenAM***



Fuente: www.google.com/search?q=configuration%20options. Consulta: 10 de marzo de 2014.

- En la página que muestra seleccionar *Custom Configuration*
- A continuación mostrará la página que se detalla en la figura 7, en esta se solicita usuario y contraseña que tendrá el administrador del sistema.

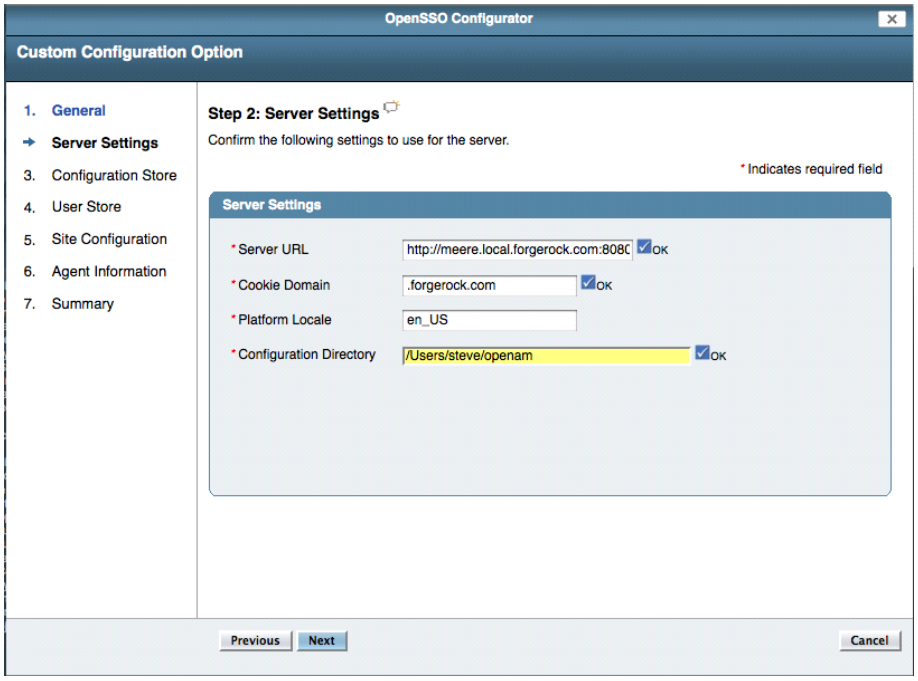
Figura 8. **Página general de configuración de *OpenAM***



Fuente:.https://www.google.com/search?q=configuration%20options. Consulta: 10 de marzo de 2014.

- Clic en el botón *next* y a continuación se desplegará la página donde se deben ingresar la configuración del servidor. Ver figura 8.

Figura 9. **Página de configuración del servidor de *OpenAM***

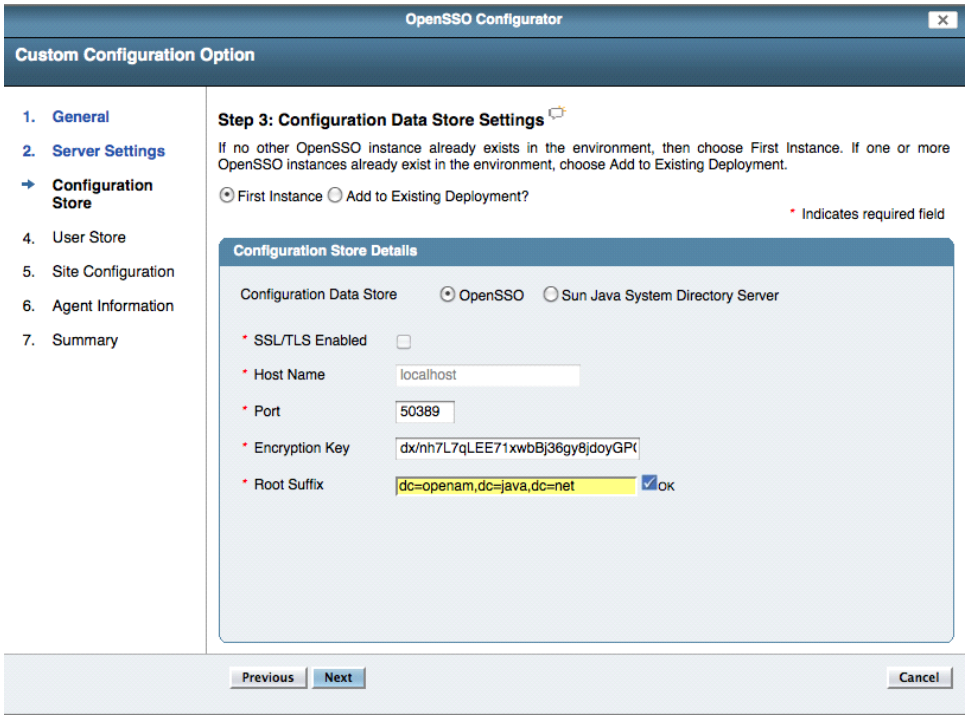


Fuente: https://www.google.com/search?q=configuration%20options. Consulta: 10 de marzo de 2014.

- Ingresar los datos del servidor:
  - *Server URL*: dirección *URL* del servidor, esto es importante, pues el dominio y el puerto deben de ser los que se serán utilizados en un ambiente de producción.
  - *Cookie Domain: OpenAM* maneja la sesión con base en una *cookie* de seguridad, esta es utilizada para autenticar la sesión, por lo que es necesario en que dominio esta *cookie* tendrá validez.
  - *Platform Locale*: se debe ingresar la configuración local de la plataforma. Es decir en que idioma se mostrarán los textos.
  - *Configuration Directory*: esto indica en que carpeta del sistema operativo se guardará la configuración de *OpenAM.*

- Clic en *next* esto mostrará la página de Configuración del almacén de datos ilustrada en la figura 9.

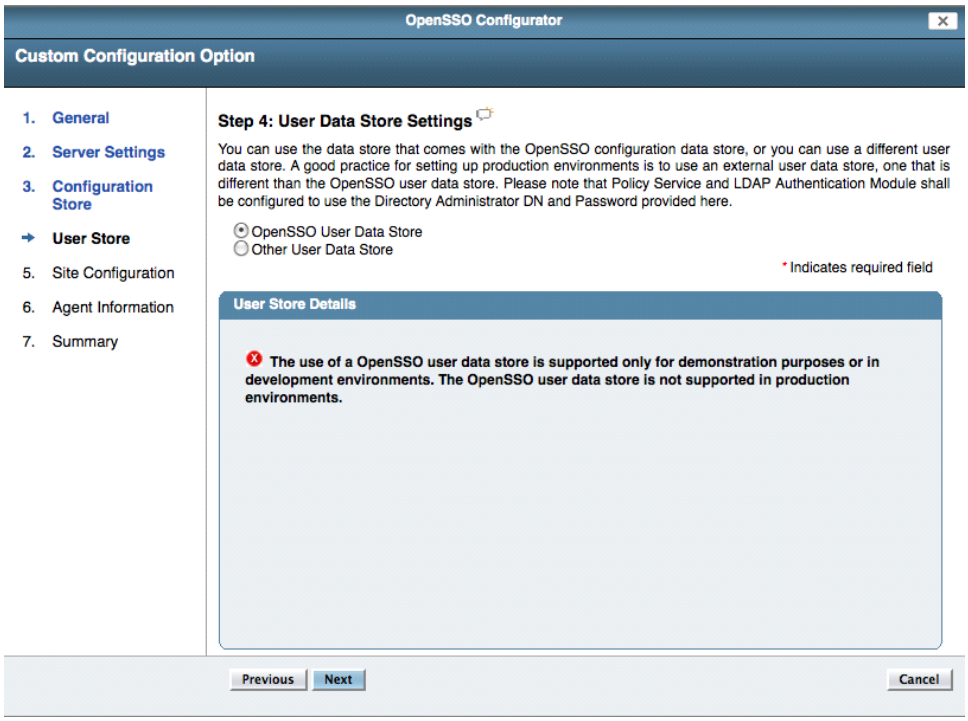
Figura 10. **Página de configuración del almacén de datos**



Fuente: https://www.google.com/search?q=configuration%20options. Consulta: 10 de marzo de 2014.

- Se recomienda dejar los datos *default* de esta página y hacer clic en *next*, esto mostrará los datos de la configuración del almacén de usuarios ilustrado en la figura 10.

Figura 11. **Página de configuración del almacén de usuarios**



Fuente: https://www.google.com/search?q=configuration%20options. Consulta: 10 de marzo de 2014.

- Seleccionar la opción *Other User Data Store* e ingresar los datos de conexión del servidor de *Ldap* instalado.
- En la configuración de sitio, seleccionar que el despliegue no está detrás de un balanceador de carga.
- En la página de configuraciones por defecto del agente de seguridad, ingresar la contraseña y usuario del mismo. Esta no debe de coincidir con la contraseña ingresada previamente del administrador del sistema.
- A continuación el instalador procederá a hacer las configuraciones necesarias para realizar la correcta instalación de *OpenAM.*

4.6.2. Instalación del agente de seguridad en Apache

Para realizar la instalación del agente de seguridad de Apache es necesario haber descargado el instalador de la página de *OpenAM* y seguir los pasos descritos a continuación:

- Verificar que exista la variable de entorno *JAVA_HOME*.
- Descomprimir el instalador en el lugar de preferencia. Se descomprimió en la carpeta */opt/OpenAM/*.
- Ingresar a la página de administración de *OpenAM* y crear un nuevo perfil para el agente de seguridad. Para esto acceder a la página de administración y navegar al control de acceso, agente, web.
- Crear un archivo que contenga la contraseña del agente de seguridad.
- Correr el programa de instalación con la opción *--install*, este se encuentra en la carpeta de *web_agents,* de la carpeta donde se descomprimió el instalador.
- Ingresar las configuraciones del servidor en el instalador.

4.6.3. Configuración de Glassfish para mod_proxy_ajp

Para configurar *Glassfish* para que utilice la seguridad de Apache es necesario instalar el módulo de apache mod_proxy_ajp y configurarlo, para que funcione como un *proxy* entre las peticiones del cliente y las respuestas de proporcionadas por *Glassfish*.

Pasos para hacer configuración de *Glassfish* para que utilice las configuraciones de apache:

- Si apache no tiene instalado *mod_proxy*, *mod_proxy_http* y *mod_proxy_ajp*, instalarlos con el comando: *a2enmod proxy proxy_http proxy_ajp*
- Configurar en la consola de administración de *Glassfish* el escucha para *AJP*.
  - Ingrese *Server config -> Servicio HTTP -> Listeners HTTP*.
  - Clic en Nuevo.
  - En el parámetro de puerto ingrese el puerto: 8009.
  - En el parámetro de *Listener JK* marcar el cuadro de cheque.
  - Clic en Guardar.
- En el directorio de configuración del dominio, generalmente ubicado en: *<Raíz glassfish>/domains/domain1/config/*.
  - Crear un archivo llamado *glassfish-jk.properties*. Con el texto: "*tomcatAuthentication=true"*
- En el archivo de configuración de apache, realizar la siguiente configuración por cada una de las aplicaciones que se quieren asegurar:
  <Location /foo>
  ProxyPass ajp://localhost:8009/foo
  ProxyPassReverse ajp://localhost:8009/foo
  </Location>

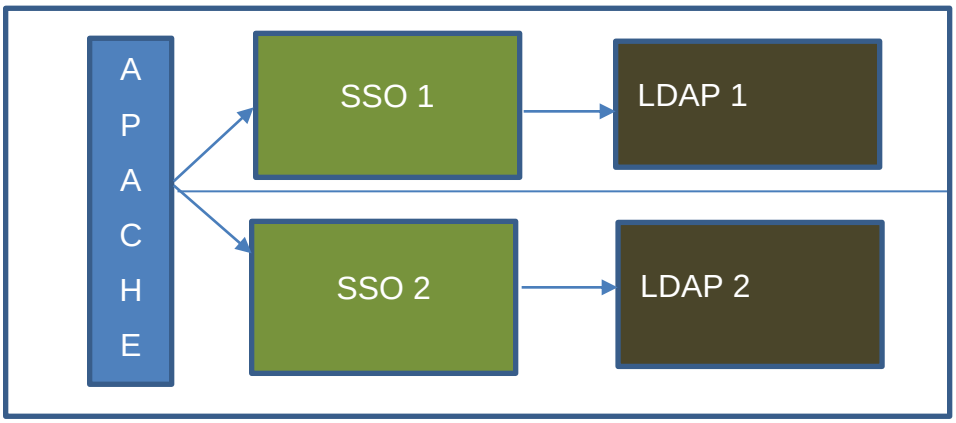
  Donde *foo* es la aplicación que se desea redireccionar a *Glassfish*.

- En la configuración del agente de seguridad, agregar el contexto asegurado.
- Agregar la siguiente propiedad de la máquina virtual de *Java*. Dcom.sun.enterprise.web.connector.enableJK. property File = *<Direccion del archivo glassfish-jk.properties>*
- Reiniciar *Apache*.

4.7. Balanceo de carga

Para el despliegue en producción se configuraron dos servidores de *LDAP* y dos servidores de *OpenAM,* los cuales están detrás de un servidor de *Apache* el cual realizó la funcionalidad de balancear la carga entre los servidores de *OpenAM, esto* con el fin de tener una plataforma escalable y que le permita al Centro de Cálculo e Investigación Educativa soportar una gran cantidad de usuarios.

Figura 12. **Despliegue de servidores para balanceo de carga**



Fuente: elaboracion propia.

5. INDUCCIÓN AL SISTEMA

Para la capacitación e inducción al sistema se determinaron dos áreas importantes que fueron instruidas:

- Los programadores, a quienes se les debió instruir la funcionalidad de la plataforma, las ventajas, desventajas, la facilidad de uso, metodología de desarrollo en un ambiente de *SSO* y servicios proporcionados por la herramienta.

- Asimismo, fue necesario capacitar al arquitecto de software del departamento, pues este debe tener los conocimientos necesarios para poder agregar nuevos agentes de seguridad, configurar páginas aseguradas y realizar configuraciones de la plataforma, también debe tener los conocimientos de las ubicaciones de los archivos más importantes y como recuperarse en caso de desastre.

5.1. Capacitación Realizada

La capacitación se realizó en cuatro sesiones, distribuidas de la siguiente forma:

- En la primera sesión se instruyó a los desarrolladores del horario diurno, en esta sesión, se capacitó acerca de la utilización de la plataforma y como acceder a los servicios proporcionados desde el punto de vista de los desarrolladores.

- En la segunda sesión se instruyó a los desarrolladores del horario nocturno, abarcando los mismos temas vistos en la primera sesión.
- En la tercera sesión, se capacitó al arquitecto de la institución, a quién se capacito en el mantenimiento de la plataforma y recuperación de desastres.
- Finalmente, en la última sesión se vio conjuntamente con las personas encargadas de la institución, las ventas y mejoras que tienen al tener una plataforma de autenticación única.

5.2. Material elaborado

Para la capacitación e inducción al sistema se desarrolló el siguiente material:

- Presentación de PowerPoint con explicación de los componentes desarrolladores, diagrama de despliegue y de secuencia.
- Manual de OpenAM con explicación de instalación y configuración del servidor.
- Manual de OpenAM para aplicaciones Java
- Manual de OpenAM para aplicaciones PHP

CONCLUSIONES

1. OpenAM es una plataforma poderosa de SSO que permite la integración con varias plataformas de programación web. En el caso de Java, permite la integración fácil a través del archivo de configuraciones *web.xml* y permite disminuir el tiempo de desarrollo, porque los desarrolladores se pueden enfocar en la lógica del negocio y no se deben de preocupar por el módulo de seguridad y autenticación, ya que SSO administra la seguridad de las aplicaciones.

2. La implementación de la plataforma ahora le permite a los programadores enfocarse en la programación de los requerimientos del negocio sin perder tiempo en el módulo de seguridad.

3. La plataforma permitió el aseguramiento de las aplicaciones web de forma segura y escalable, siguiendo estándares internacionales de seguridad web.

4. La plataforma les permitirá la integración con aplicaciones web aseguradas en la nube, como lo son los servicios de Google y de Microsoft, ya que la misma implementa servicios de comunicación estándares de autenticación.

5. La plataforma permite la administración de usuarios y credenciales de forma centralizada, lo que le permite a los administradores enfocarse en el negocio sin el desgaste que implicaría tener el almacenamiento de usuarios de forma descentralizada.

6. La aplicación de políticas de seguridad puede traer muchos beneficios a la organización, pero si estas no son bien aplicadas o analizadas, pueden crear conflictos dentro de la organización.

7. La inversión en un sistema de seguridad siempre es recomendable pero no se debe olvidar el factor humano, que si no es educado para que interactúe con los sistemas, siempre será la principal amenaza, una amenaza que es difícil de apreciar y reducir.

RECOMENDACIONES

1. A los desarrolladores: realizar un análisis sobre que recursos deben de ser asegurados y que recursos deben de ser públicos, a la vez colocar los recursos asegurados dentro de una carpeta asegurada y los recursos no asegurados colocarlos en otra carpeta, esto con el fin de facilitar la administración de los recursos.

2. A los arquitectos: utilizar la plataforma como un medio de integración con servicios legados en la nube como lo son las aplicaciones de Google.

3. A los desarrolladores: leer la documentación oficial de *OpenAM* para que puedan utilizar la plataforma implementada de la manera adecuada.

BIBLIOGRAFÍA

1. BERENGUELA CASTRO, Alfonso Antonio; CORTÉS COLLADO, Juan Pablo. *Metodología de medición de vulnerabilidades en redes de datos de organizaciones* [en línea]. Dirección: Raúl Astorga Barrios, Chile: INACAP, 2006. http://seguinfo.zzl.org/tesis/medicion-vulnerabilidades.zip. [Consulta: 20 de junio de 2014].

2. CRAIG, Mark; JANG Mike; RICHIE Vanessa. *OpenAM Installation Guide* [en línea]. Estados Unidos: ForgeRock, 2014. http://openam.forgerock.org/openam-documentation/openam-doc-source/doc/install-guide/index.html.[Consulta: 14 de junio de 2014].

3. CRAIG, Mark. *Getting Started With OpenAM* [en línea]. Estados Unidos: ForgeRock,2014. http://openam.forgerock.org/ openam-documentation /openam-doc-source/doc/getting-started/index.html [Consulta: 13 de junio de 2014].

4. GAVARRETE, Ana Cristina. *Seguridad informática en las empresas para la protección de datos*. Trabajo de graduación de Ing. en Sistemas. Universidad Mariano Gálvez, Facultad de Ingeniería, 2004. 116 p.