UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería

Propuesta de aplicación web-java para validación de permisos en formularios web con autenticación y autorización dinámica de un sistema de tickets integrando el módulo Shiro Cas para una empresa del gremio azucarero de Guatemala.

Trabajo de graduación presentado por Kenneth Franssua de León Cazzali para optar al grado académico de Licenciado en Tecnología de Sistemas Informáticos.

Guatemala,

2022

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería

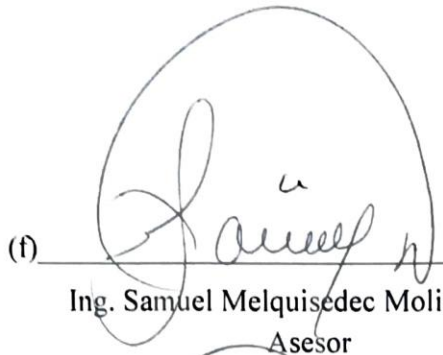
Propuesta de aplicación web-java para validación de permisos en formularios web con autenticación y autorización dinámica de un sistema de tickets integrando el módulo Shiro Cas para una empresa del gremio azucarero de Guatemala.

Trabajo de graduación presentado por Kenneth Franssua de León Cazzali para optar al grado académico de Licenciado en Tecnología de Sistemas Informáticos.

Guatemala,

2022

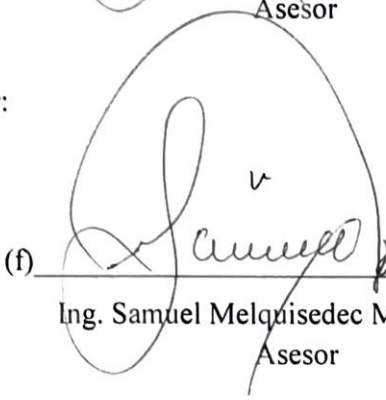
Vo.Bo. :



(f)_____

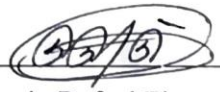
Ing. Samuel Melquisedec Molina Donis
Asesor

Tribunal Examinador:



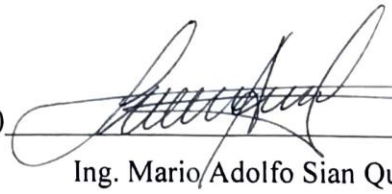
(f)_____

Ing. Samuel Melquisedec Molina Donis
Asesor



(f)_____

Ing. Ricardo Rafael Figueroa Reyes
Examinador



(f)_____

Ing. Mario Adolfo Sian Quisque
Director

Fecha de aprobación: Guatemala, 13 de diciembre de 2022

ÍNDICE

LISTA DE FIGURAS

LISTA DE CUADROS

RESUMEN

El presente trabajo propone la integración de Shiro Cas para validar permisos en formularios web con java server faces para el sistema de tickets de una empresa de la región sur del gremio azucarero. Derivado a la gran cantidad de colaboradores con los que cuenta la empresa y obsolescencia de su actual sistema de validación, realizar la migración de todos esos datos al nuevo protocolo presenta un gran reto; poniendo en práctica conocimientos en el manejo del módulo de protección de aplicaciones web Shiro Cas se encontró el desafío interesante y posible de realizar.

Se iniciará con un diagnóstico de la compatibilidad del actual sistema de tickets de una empresa del gremio azucarero de la región sur del país con el módulo de protección de aplicaciones web Shiro Cas.

Con la propuesta de integración del módulo Shiro Cas se pretende fortalecer la seguridad y optimizar el procedimiento de verificación actual. Aplicando estas mejoras se fortalecerá el sistema de tickets actual con el que cuenta la empresa.

La propuesta de esta actualización de seguridad en el sistema de tickets de la empresa se pretende evitar ataques externos o suplantación de identidad dentro de la empr

I. INTRODUCCIÓN

El enfoque de este trabajo gira en torno a la propuesta del desarrollo de la aplicación web-java para validación de permisos en formularios web con autenticación y autorización dinámica de un sistema de tickets integrando el módulo Shiro Cas para una empresa del gremio azucarero de Guatemala.

Tiene como fin lograr un mejor control de los permisos de los usuarios a efecto de disminuir el riesgo de suplantación de identidad que permita poner en peligro la información de la empresa.

La propuesta de desarrollo de la aplicación se sustenta en las razones del por qué utilizar Shiro Cas y se debe a que es fácil de usar, integral, flexible y compatible con la web. Y porque en la actualidad ya algunas empresas utilizan Shiro para proteger su software o sitios web.

II. OBJETIVOS

A. Objetivo general

1. El presente trabajo propone la creación de una aplicación de control de permisos para formularios web-java en un sistema de tickets de una empresa del gremio azucarero de Guatemala para lograr un mejor control de los permisos de los usuarios.

B. Objetivos específicos

1. Crear una aplicación de control de permisos para formularios web-java en un sistema de tickets con autentificación y autorización dinámica para una empresa del gremio azucarero de Guatemala.

2. Gestionar la autentificación y autorización de permisos en formularios web integrando el módulo de protección de aplicaciones web Shiro Cas para reducir el riesgo de suplantación de identidad de los usuarios del 70% al 30%.

3. Reforzar la seguridad para validación de permisos en formularios web-java en un 70% para el sistema de tickets para una empresa del gremio azucarero de Guatemala.

III. JUSTIFICACIÓN

Debido a la gran cantidad de colaboradores con los que cuenta la empresa del gremio azucarero de Guatemala, existe la necesidad de tener un mejor control de autenticación y autorización de permisos en formularios web ya que actualmente no cuentan con un módulo de protección de aplicaciones web.

La propuesta se realiza con el objetivo de tener una disminución de riesgo asumible de la información de una empresa del gremio azucarero de Guatemala, utilizando la aplicación web-java para validación de permisos en formularios web con autenticación y autorización dinámica de un sistema de tickets integrando el módulo de protección de aplicaciones web Shiro Cas.

Por lo tanto, con la propuesta de integración del módulo de protección de aplicaciones web Shiro Cas es posible tener un mejor control en la verificación de identidad del usuario y validación de permisos, reduciendo el riesgo de suplantación de identidad de los usuarios.

IV. MARCO TEÓRICO

A. Ciberseguridad

La evolución de la ciberseguridad nos brinda un contexto más detallado de cómo se ha ido transformando el mundo digital y los riesgos que surgieron en el proceso.

El primer hacker de la historia Fur Nevil Maskelyne, en el año de 1903 interceptó la primera trasmisión de telégrafo inalámbrico, demostrando las vulnerabilidades del sistema desarrollado por Marconi.

John Draper fue el primer ciberdelincuente, se le conoció como "capitán Crunch", descubrió que el sonido emitido por un silbato que se obsequiaba en las cajas de cereal de Cap´n Crunch, podían engañar a la señal de la central telefónica y poder realizar llamadas gratis.

En los años 70's apareció el primer malware de la historia, Creeper, un programa que se replicaba así mismo. Este malware mostraba el mensaje "i´m a creeper, catch me if you can!". A partir de ese hecho nació el primer antivirus llamado Reaper el cual su función principal era la de eliminar las infecciones de equipos con Creeper.

En los años 80's el malware incrementó su presencia ya que a finales de esta década Kevin Mitnick utilizó ingeniería social para tener acceso a información personal y confidencial; este ciberataque, el cual comenzó en esa época sigue siendo uno de los más populares para vulnerar los activos de una empresa, sin embargo, se pueden prevenir y reducir con una buena estrategia, formación de colaboradores y protocolos.

La ciberseguridad consiste en proteger con tecnologías, procesos y aprendizaje los sistemas, redes y programas de ciberataques, daño y accesos no autorizados. Por lo general un ataque da como resultado desde a un robo de identidad, manipulación de los datos, destrucción de información confidencial, intentos de extorsión o interrupción de la

continuidad de trabajo de las empresas. Hay tres pilares en ciberseguridad que en conjunto se conoce como el modelo de confidencialidad, integridad y disponibilidad. (CIA por sus siglas en inglés)

1. Confidencialidad

La información debe de ser accesible únicamente por usuarios autorizados a lo que le corresponde.

2. Integridad

La información solo debe de ser modificada por las personas o los procesos adecuados, garantizando que no sea comprometida voluntaria e involuntariamente.

3. Disponibilidad

La información siempre debe de estar disponible a cualquier usuario o proceso autorizado siempre que éste lo requiera.

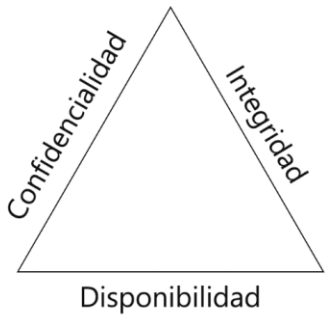


Figura 1. Triángulo confidencialidad, integridad y disponibilidad.
(Descripción de los conceptos básicos de la ciberseguridad / Descripción de amenazas, ataques y mitigaciones básicos de ciberseguridad)

B. Autenticación

Este es un proceso en el que consiste que una persona demuestre que es quien dice ser, usualmente se solicita nombre de usuario el cual no es suficiente por sí solo para que demuestre que sí es quien dice ser, por ello también se solicita que se ingrese una contraseña que únicamente el usuario debe de conocer. La autenticación es muy importante para garantizar que sólo los usuarios autorizados puedan obtener acceso a la información y recursos que pueden ser confidenciales, reduciendo el riesgo que puedan hacerse pasar por alguien más para obtener el control, manipular o copiar información.

C. Métodos de autenticación

Estos métodos se diferencian unos de otros siendo interpretados como capas de seguridad que mantienen en confidencialidad los datos permitiendo confirmar que un usuario es quien dice ser. Se puede dividir en tres tipos: algo que sabe, algo que se tiene y algo que forma parte de usted.

1. Algo que sabe

Este tipo de autenticación se basa en información que solo el usuario conoce.

- Contraseñas.
- PIN.
- Preguntas de seguridad.

2. Algo que tiene

Este tipo de autenticación se basa por pertenencias del usuario.

- Documentos de identidad.
- Llaves USB.
- Computadoras.
- Smartphone o teléfono móvil.

3. Algo que forma parte de usted

Este tipo de autenticación se basa en alguna característica física del usuario.

- Huella digital.
- Reconocimiento facial.
- Escaneo de retina.

Figura 2. Ejemplo de tipos de autenticación.
(Descripción de los conceptos básicos de la ciberseguridad / Descripción de la autenticación y la autorización en ciberseguridad)

D. Autorización

Cuando el usuario se haya autenticado, tendrá que observar lo que se le permita ver y a donde puede ir, siendo éste el proceso por el cual el usuario identificado es autorizado a acceder a determinados recursos, a esto se le denomina autorización. Su función es determinar el nivel de acceso de una persona a los datos y recursos.

E. Vulnerabilidades

En ciberseguridad se entiende como vulnerabilidad a las debilidades que un atacante puede aprovechar para tener acceso a la información confidencial o sabotear operaciones, otra forma por las que se les conoce es, agujeros de seguridad los cuales ponen en riesgo

los datos de una empresa, tienen la ventaja que al ser descubiertas pueden ser solventadas. Existen vulnerabilidades del día cero, las cuales generalmente no fueron reveladas durante algún tiempo o se descubrieron recientemente, dándoles a los atacantes oportunidad y tiempo más que suficiente para lograr su cometido.

Vulnerabilidades comunes

El conocer las vulnerabilidades comunes que ponen en riesgo la seguridad de la empresa en la red, favorece la protección de los recursos tecnológicos e información de la empresa. Los tipos de vulnerabilidades comunes son:

- Contraseñas débiles.
- Apuntar la contraseña y dejarla de forma accesible a las demás personas.
- El factor humano causado por la falta de formación y concienciación.
- Fallos en la validación de datos introducidos.
- Fallos en la gestión de permisos, privilegios y/o control de acceso.

F. Amenaza

Es toda posibilidad de sufrir un ataque y que las credenciales del usuario sean descubiertas por un tercero para fines maliciosos, también conocido como un evento con un gran potencial de afectar negativamente a las operaciones de una empresa. Unas de las amenazas más importantes son:

- Robo de identidad

  Esta amenaza consiste en que el usuario facilite de forma involuntaria sus credenciales de acceso a un tercero que si lo desea las utilizará de forma fraudulenta.

- Denegación de servicio

  Se le conoce como ataque de DDoS, pretende colapsar los servidores enviándoles una enorme cantidad de peticiones haciendo que no puedan ser atendidas.

- Negligencia

  Los errores humanos y el incumplimiento de las políticas y normas de seguridad de la empresa ponen en peligro a los datos de la empresa.

G. Riesgo

Es toda posibilidad que en el sistema ocurra un incidente de seguridad y que una amenaza se materialice causando daños, todo riesgo aparece cuando una amenaza tiene probabilidad de convertirse en un desastre con pérdidas de algún tipo.

H. Diferencia entre vulnerabilidad, amenaza y riesgo

Se entiende como vulnerabilidad a cualquier debilidad que permita atacar y ponga en peligro a una organización, aunque ésta haya realizado esfuerzos en protegerse. La amenaza se refiere a un incidente nuevo o recién descubierto con potencial de causar algún daño y el riesgo es todo el potencial de daño o pérdida cuando una amenaza explota una vulnerabilidad.

I. Framework

Es un marco o esquema de trabajo, conjunto de herramientas y módulos reutilizables para varios proyectos sirviendo como punto de partida para crear y diseñar software. Utilizar frameworks simplifica la elaboración de una tarea ya que sólo es necesario completarlo de acuerdo con lo que se desea realizar, por esta razón es una de las herramientas más utilizadas por programadores.

Sirve para crear un proyecto con un menor tiempo, permitiendo tener un código más limpio de forma rápida y eficaz. También hace posible que se disminuyan considerablemente las pruebas y los errores que se puedan presentar, agilizando varios procesos en el desarrollo, permitiendo principalmente reutilizar herramientas o módulos.

Permite reutilizar el código las veces que consideremos que sean necesarias inclusive optimizarlo, dándonos ciertas ventajas que conlleva este proceso. También permite optimizar tareas en la programación dándonos como resultado ser más eficaz al disminuir el tiempo en el que se programa. Al elegir el framework se debe de conocer a detalle el proyecto, debido a que dependiendo del tipo de framework que elijamos dependerá lo antes mencionado.

Al reducir tiempos nos da una mayor optimización del trabajo, facilitar tareas y poder evitar errores, teniendo esto en cuenta es posible notar que es un recurso que facilita en gran parte las tareas de los desarrolladores. Asimismo, se obtienen las siguientes ventajas:

- Disminuye la cantidad de riesgos al tiempo que facilita la resolución de errores, garantizando mayor seguridad al tener gran parte de las potenciales vulnerabilidades resueltas, gracias a la gran cantidad de desarrolladores, comunidades y foros en los cuales se puede consultar y resolver dudas respecto al uso del framework.

- Facilita los desarrollos colaborativos, esto quiere decir que favorece el trabajo colaborativo, al dejar definidos unos estándares que permiten que distintos miembros de un mismo equipo puedan trabajar de forma coordinada, inclusive favorece que se comparta código y reduzca la curva de aprendizaje propia o de otros miembros del equipo.

- Al estar ampliamente extendido se puede encontrar fácilmente herramientas, módulos e información para su uso; esto quiere decir que es de fácil acceso a recursos e información útil. Como complemento de lo mencionado, permite utilizar programación avanzada a la que de otras formas sería mucho más difícil llegar.

J. Entorno de desarrollo integrado

Los desarrolladores acuden a distintas herramientas como editores de texto para la creación de un proyecto, adicionalmente durante el ciclo de vida del desarrollo tienen que incluir bibliotecas de códigos, compiladores y plataformas de prueba requiriendo algo más que solo el conocimiento práctico del código lo que conlleva utilizar algo más de esfuerzo y tiempo.

Un entorno de desarrollo integrado (IDE) es una aplicación de software que reúne y combina herramientas y recursos como editar, crear, probar y empaquetar software en una aplicación facilitando el trabajo, permitiendo acceder a ellos a través de una única interfaz de usuario (GUI). Con esto el usuario debería ser capaz de realizar en su mayoría las tareas de desarrollo de un proyecto desde dentro del IDE, pudiendo permitir una entrega más rápida de tareas y proyectos teniendo un control más detallado aumentando la productividad.

Los entornos de desarrollo integrado se clasifican distintas categorías, algunos de los tipos de IDE son:

1. IDE locales

Los IDE locales son aquellos los que los desarrolladores instalan en sus equipos. Posteriormente también descargan e instalan varias bibliotecas adicionales dependiendo de preferencias o necesidades. Estos son personalizables, no requieren de una conexión a internet. Algo importante de los IDE locales es que algunos pueden llegar consumir unos considerablemente recursos del equipo ralentizándolo de una forma significativa.

2. IDE en la nube

El IDE en la nube lo configuran los equipos de trabajo que desean escribir, editar y compilar para prescindir de la necesidad de descargar software en sus equipos ofreciendo algunas ventajas en comparación con los IDE tradicionales, dichas ventajas son las siguientes:

- Un entorno de desarrollo estandarizado debido a que se pueden configurar de forma central para crear un entorno de desarrollo estándar ayudando a evitar errores que puedan ocurrir debido a las distintas configuraciones de los equipos locales.

- Mejoran el rendimiento ya que requieren menos capacidad de recursos, un IDE tradicional requiere una gran capacidad de recursos del equipo y muy posiblemente pueda llegar a ralentizar el equipo del desarrollador.

Los IDE son eficaces ya que proporcionan todo lo que necesita un desarrollador para crear y ejecutar aplicaciones, no todos los IDE comparten los mismos componentes, sin embargo, tienen herramientas comunes que todos las incluyen en un paquete de software IDE, siendo las siguientes:

- Editores de código fuente

Posiblemente la función principal de un IDE es editar texto. Todo entorno de desarrollo integrado un editor de texto que ayuda a escribir y revisar el código. Agregado a ello emplea funciones con indicaciones visuales mediante una interfaz simple como el resultado de la sintaxis, relleno automático específico para el lenguaje empleado y comprobación de errores a medida que se escribe el código.

- Compiladores

Los compiladores convierten el código de alto nivel creador en el editor de texto y lo traducen en instrucciones en el lenguaje de máquina que pueda comprender la unidad de procesamiento central (CPU) de un ordenador digital.

- Depuradores

Sirve para que el código que está escrito y compilado en fase de validación se proceda al proceso de depuración el cual es corregir todos los errores o fallas revelados en las pruebas. Fueron diseñados con la función de ayudar a localizar errores en el código que a su vez prueba el rendimiento y funcionalidad de la aplicación.

- Finalización del código

Estas opciones facilitan aún mas el programar mediante la identificación y adición automática de componentes de código estándar. Los IDES con finalización de código permiten la reducción de probabilidad de errores de codificación.

- Compatibilidad con lenguajes de programación

La mayoría de los IDE están diseñados para funcionar con un lenguaje de programación en específico (como ejemplo Python, Java o Ruby), pero algunos admiten varios lenguajes de manera conjunta.

- Integraciones/complementos

Un entorno de desarrollo integrado reúne software y herramientas esenciales en un solo lugar para el desarrollo de aplicaciones, debe de funcionar como parte del ecosistema general de TI de una organización. Los IDE permiten que los usuarios puedan integrar más herramientas que tienden a crear un flujo de trabajo más optimizado de aquellos que carecen de capacidades de integración.

K. Entorno de desarrollo integrado NetBeans

NetBeans IDE es un entorno de desarrollo integrado gratuito y de código abierto para el desarrollo de aplicaciones en los sistemas operativos Windows, Mac, Linux y Solaris. Simplifica el desarrollo de aplicaciones web, empresariales y móviles que utilizan las plataformas Java y HTML5, además, ofrece soporte para el desarrollo de aplicaciones PHP y C/C++.

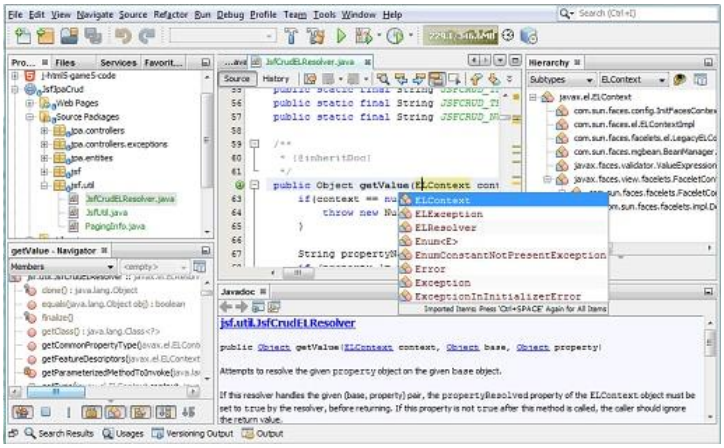


Figura 3. Ejemplo de interfaz IDE NetBeans.

(Oracle NetBeans IDE

NetBeans proporciona soporte gratuito y completo para tecnologías y plataformas clave de Oracle:

Java

NetBeans ofrece herramientas para el desarrollo de aplicaciones móviles, escritorio, empresariales y web Java. Es el primero en admitir las ultimas versiones de JDK, Java EE y JavaFX. Brinda resúmenes inteligentes que ayudan a comprender y administrar aplicaciones, incluyendo compatibilidad inmediata con tecnologías populares, como Maven.

- JDK

El Kit de desarrollo de Java (JDK) es un entorno de desarrollo de software utilizado para desarrollar aplicaciones y applets de Java. Incluye el Java Runtime Enviroment (JRE), un intérprete / cargador (java), un compilador (javac), un archivador (jar), un generador de documentación (javadoc) y otras herramientas necesarias para el desarrollo de Java.

- Java EE

Java Enterprise Edition (Java EE) es el estándar de la industria para desarrollar aplicaciones Java portables, robustas, escalables y seguras.

- JavaFX

Es un conjunto de paquetes gráficos y medios que permite a los desarrolladores diseñar, crear, probar, depurar e implementar aplicaciones con aspecto vanguardista y contenidos avanzados, audio y video.

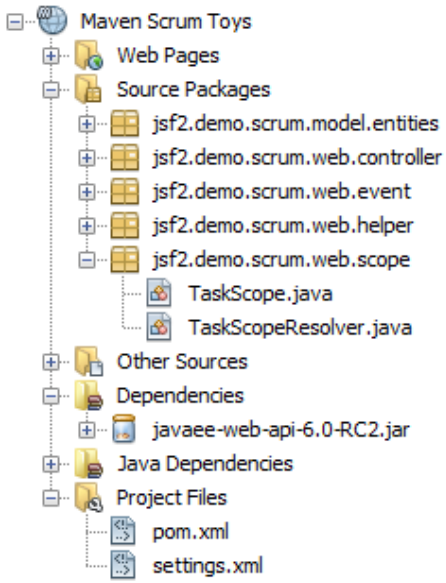


Figura 4. Ejemplo de la estructura de proyecto en NetBeans.

(Oracle NetBeans IDE)

L. Lenguaje de programación Java

Es un lenguaje de programación orientado a objetos, concurrente, de propósito general, que fue diseñado específicamente parar tener pocas dependencias como fuera posible. Permite el desarrollo de aplicaciones en diversas áreas, como de seguridad, animación, acceso a base de datos, aplicaciones cliente-servidor, interfaces gráficas, páginas web interactivas y desarrollo de aplicaciones móviles. La intención de java es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (WORA o "write once, run enywhere"" en inglés) lo que significa que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. A partir del 2012 se volvió uno de los lenguajes de programación mas populares en uso, en particular en aplicaciones de cliente-servidor de web, con unos 10 millones de usuarios reportados.

El lenguaje de programación Java fue originalmente desarrollado por James Gosling de Sun Microsystems (la cual fue adquirida por la compañía Oracle) y publicado en 1995 como un componente fundamental de la plataforma Java de Sun Microsystems. Su sintaxis deriva en gran medida de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera

de ellos. Las aplicaciones de Java son generalmente compiladas a bytecode (clase Java) que puede ejecutarse en cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente.

La compañía Sun desarrolló la implementación de referencia original para los compiladores de Java, máquinas virtuales y librerías de clases en 1991 y las publicó por primera vez en 1995. A partir de mayo de 2007, en cumplimiento con las especificaciones del Proceso de la Comunidad Java, Sun volvió a licenciar la mayoría de sus tecnologías de Java bajo la Licencia Pública General de GNU. Otros también han desarrollado implementaciones alternas a estas tecnologías de Sun, tales como el Compilador de Java de GNU y el GNU Classpath.

La importancia del lenguaje de programación Java permitir diseñar softwares que puedan ser ejecutados y distribuidos en distintas plataformas sin tener que modificarlos e incluso sin pensar en la arquitectura del equipo.

Java fue creado principalmente con cinco fundamentos los cuales son:

- Debería usar el paradigma de la programación orientada a objetos.
- Debería permitir la ejecución de un mismo programa en múltiples sistemas operativos.
- Debería incluir por defecto soporte para trabajo en red.
- Debería diseñarse para ejecutar código en sistemas remotos de forma segura.
- Debería ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos, como C++.

Java tiene unas características que lo diferencia de otros lenguajes de programación y esta son:

- Es simple

Debido a que Java está inspirado y deriva de C y C++, omitiendo todas las características menos usadas y confusas lo hace uno de los lenguajes de programación más sencillos.

- Orientado a objetos

El enfoque orientado a objetos (OO) se refiere a un tipo de programación el cual utiliza a los objetos resultantes en sus interacciones. Un objeto contiene varios datos bien estructurados que pueden ser o no visibles. Sus principales características son el polimorfismo y la herencia.

- Independiente a la plataforma

Significa que todos los proyectos que hayan sido escritos en el lenguaje de programación Java pueden ejecutarse en cualquier tipo de hardware, lo que lo hace convenientemente portable.

- Recolector de basura

Previene posibles fugas de memoria con el recolector de basura de Java que borra a un objeto cuando no hay referencias localizadas en dicho objeto.

- Multihilo

Logra llevar a cabo varias tareas simultáneamente dentro del mismo programa permitiendo mejorar el rendimiento y la velocidad de ejecución.

M. Framework security

Es una compilación de políticas y procesos que incluyen instrucciones específicas útiles para manejar la información de una manera que se garantice un menor margen de vulnerabilidad a los riegos relacionados con la seguridad. Las empresas han manifestado hacer esfuerzos por integrar estas pautas de seguridad ya que demuestran ser útiles en industrias enteras.

N. Apache Shiro

Es un Java security framework (Marco de seguridad de Java) open-source que realiza autenticación, autorización, criptografía y administración de sesiones. Este fue diseñado de una forma en la cual pueda proporcionar protección rápida con implementación relativamente fácil a cualquier aplicación ya sea aplicaciones pequeñas o aplicaciones más grandes como las empresariales.

Shiro nos brinda la API de seguridad de la aplicación permitiéndonos tener los siguientes pilares de seguridad en una aplicación:

- Autenticación de la identidad del usuario para verificar si es quién dice ser.

- Autorización o control de acceso para determinar si el usuario tiene acceso o no a determinado botón.

- Criptografía para proteger u ocultar la información de aquellos que no deben de conocer la información.

- Gestión de sesiones durante un periodo de tiempo por usuario cuando usan la aplicación.

Apache shiro como framework de seguridad presenta las siguientes características:

- Simplificar la seguridad por medio de una implementación rápida y fácil.

- Una fácil integración con aplicaciones web brindando soporte a servicios como Rest.

- Almacenamiento en caché el cual permite desarrollar aplicaciones web en menos tiempo ser eficaces.

Al revisar la historia de Shiro encontramos que su predecesor, Jsecurity, fue fundado en 2004 por Les Hazlewood y Jeremy Haile ya que no se podía encontrar un marco de seguridad Java adecuado que funciona bien al nivel de aplicación. El proyecto continuó creciendo hasta que Apache aceptó en el programa incubador para ser administrado por mentores con el fin de convertirse un proyecto de Apache de primer nivel.

1. Arquitectura general de Apache Shiro

La arquitectura de Shiro tiene tres conceptos principales los cuales son subject, securitymanager y realms.

- Subject (Asunto)

Es una “vista” especifica del usuario que se está ejecutando, básicamente todo lo que esté interactuando con el software.

- SecurityManager (Gestor de seguridad)

Es el corazón de la arquitectura Shiro actuando como coordinador de sus componentes de seguridad internos y todos estos están configurados para una aplicación.

- Realms (Reinos)

Actúan como “puente” o “conector” entre los Shiro y los datos de seguridad de la aplicación. Encapsula los datos asociados a disposición para realizar autenticación y autorización.

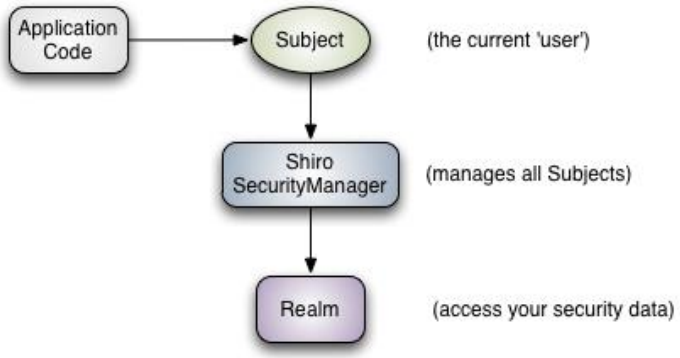


Figura 5. Ejemplo de arquitectura general de Apache Shiro.
(Apache Shiro Architecture)

O. ¿Por qué usar Apache Shiro?

La adopción de Shiro por parte de las empresas durante los años demuestra el crecimiento significativo y solidez que tiene en proyectos pequeños y grandes. Las razones del por qué lo utilizan las empresas se debe a que es fácil de usar, integral, flexible y compatibilidad con la web. Por mencionar algunas empresas que utilizan Shiro para proteger su software o sitios web son Sonatype y Mulesoft.

Las razones por las que seguir usando Apache Shiro:

- Fácil de usar

El objetivo de Shiro ha sido la facilidad de uso, ofrecer seguridad a las aplicaciones sin ser confuso o frustrante para los programadores novatos.

- Integral

Afirma que no existe otro marco de seguridad con la amplitud de Apache Shiro por lo que muy probablemente sea un “servicio integral”.

- Compatible con la web

Tiene un gran soporte de aplicaciones web sin exigir ninguna especificación ni tener muchas dependencias.

- Conectable

Shiro facilita la integración de muchos marcos y aplicaciones. Se integra sin problemas con Spring, Grails, Wicket, Tapestry, Mule, Apache Camel, Vaadin entre otros.

V. METODOLOGÍA

A. Alcance

El alcance de esta propuesta es permitir la autenticación del usuario por medio de una contraseña y conforme a la autorización de permisos el usuario pueda ver lo que tenga autorizado ver.

B. Diagnóstico de compatibilidad

Antes de iniciar con la propuesta se buscó la asesoría de un miembro de una empresa del gremio azucarero de Guatemala para corroborar la compatibilidad de Apache Shiro con el sistema de tickets de una empresa del gremio azucarero de Guatemala, además, de llevar a cabo una investigación acudiendo a la documentación oficial de Shiro. De esta forma se determinó que es compatible y es posible la implementación del módulo de protección de aplicaciones web Shiro.

C. Características de la aplicación web-java

- Acceso del usuario

El usuario tendrá acceso por medio de credenciales ya registradas, por ende, utilizará un usuario y contraseña registrados para su acceso.

- Mostrar información

La información es accesible para el usuario por medio de un botón el cual permite visualizarla.

- Modificar información

El usuario pude modificar la información ya ingresada por medio de un botón el cual permite dicho proceso.

- Eliminar la información

Esta función solo es accesible para determinados usuarios, permite eliminar la información ya ingresada por medio de un botón.

D. Tipos de usuario y sus funciones

- Diagrama de caso de uso general

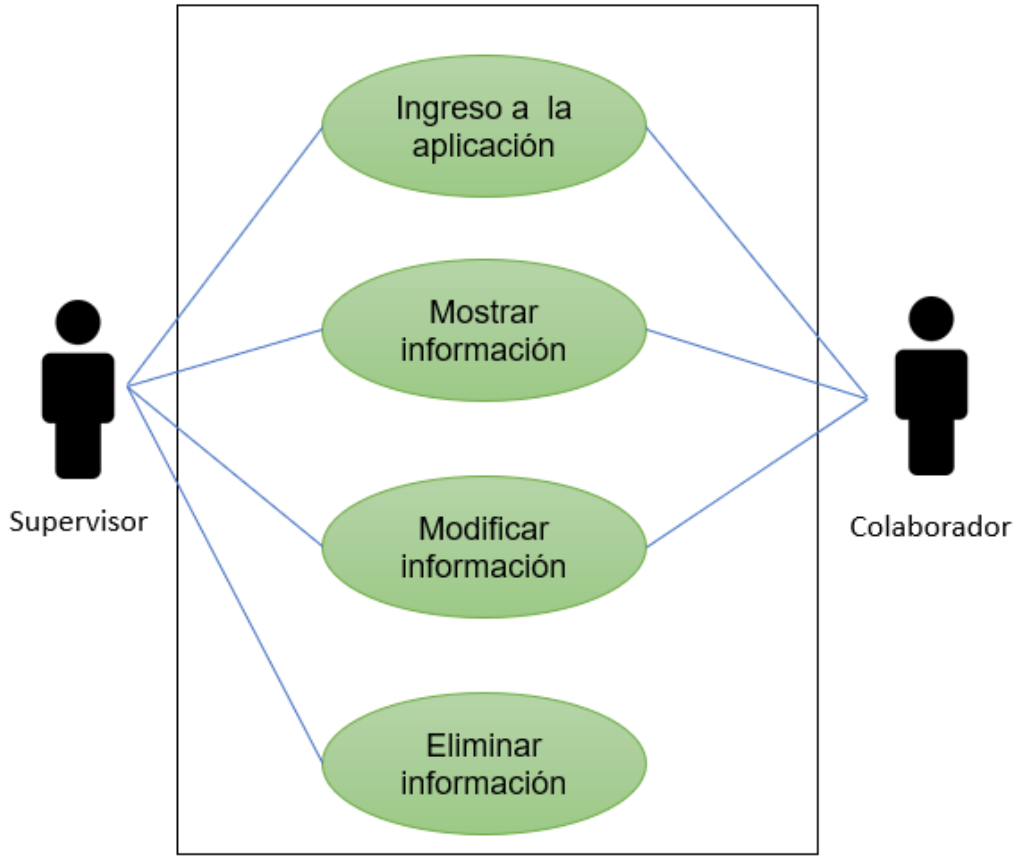


Figura 6. Diagrama de caso de uso general.

- Supervisor

Rol:	Supervisor
Funciones:	- Ingreso a la aplicación web-java - Mostar información - Modificar información - Eliminar información

Tabla 1. Usuario supervisor.

Caso de uso descriptivo usuario supervisor	
Responsable	Supervisor
Descripción	El usuario accede a la aplicación por medio de una contraseña, al ingresar el módulo Shiro Cas determina su autorización de permisos.
Actividades	- Mostar información - Modificar información - Eliminar información
Visualización	Puede ver lo que el módulo Shiro Cas le autorice ver por medio de su Rol.

Tabla 2. Caso de uso descriptivo supervisor.

- Colaborador

Rol:	Colaborador
Funciones:	- Ingreso a la aplicación web-java - Mostar información - Modificar información

Tabla 3. Usuario colaborador.

Caso de uso descriptivo usuario supervisor	
Responsable	Colaborador
Descripción	El usuario accede a la aplicación por medio de una contraseña, al ingresar el módulo Shiro Cas determina su autorización de permisos.
Actividades	- Mostar información - Modificar información
Visualización	Puede ver lo que el módulo Shiro Cas le autorice ver por medio de su Rol.

Tabla 4. Usuario colaborador.

VI. RESULTADOS

A. Funcionalidad de login y autenticación

La finalidad de login solicita que se ingrese los datos de usuario y contraseña. El proceso de autenticación lo proporciona el framework Apache Shiro verificando los datos de usuario y contraseña que el usuario ingresó, para permitir su ingreso.

Figura 7. Pantalla de login de usuarios.

B. Autorización

Cuando el usuario haya ingresado al sistema, el framework Apache Shiro ayuda a determinar si el usuario tiene las funcionalidades del usuario supervisor o colaborador.

C. Funcionalidades supervisor

El usuario supervisor puede ingresar datos nuevos, modificar datos, mostrar los datos y la función característica del usuario supervisor de eliminar los datos.

Figura 8. Pantalla de usuario supervisor.

D. Funcionalidades colaborador

El usuario colaborador puede ingresar nuevos datos, mostrar los datos y modificar los datos.

Figura 9. Pantalla de usuario colaborador.

VII. ANÁLISIS DE RESULTADOS

1. La implementación del framework Apache Shiro permite realizar el fortalecimiento de la seguridad del sistema de tickets, con el fin de pretender evitar ataques externos o suplantación de identidad de los usuarios ya que proporciona el proceso de autenticación solicitando el ingreso de un usuario y contraseña y verificando los datos ingresados, para permitir su ingreso.

2. Esta propuesta responde a la necesidad de tener un mejor control de autenticación y autorización de permisos en formularios web con el objetivo disminuir el riesgo de que la información se vea comprometida fortaleciendo la seguridad del sistema de tickets de una empresa del gremio azucarero de Guatemala ya que cuando el usuario haya ingresado al sistema, el framework Apache Shiro ayuda a determinar si el usuario tiene las funcionalidades del usuario supervisor o colaborador.

3. La empresa del gremio azucarero de Guatemala tiene una gran cantidad de colaboradores y no cuenta con un módulo de protección de aplicaciones web, permitiendo que la información de la empresa se pueda ver comprometida de forma voluntaria o involuntaria por lo que el usuario supervisor es el único con la funcionalidad de eliminar datos.

4. El usuario colaborador tiene permitidas las funcionalidades de ingresar nuevos datos, mostrar los datos y modificar los datos sin que tenga autorización de eliminar los datos.

VIII. CONCLUSIONES

Teniendo en cuenta el Objetivo General planteado se puede concluir que:

1. Para la creación de una aplicación de control de permisos para formularios web-java en un sistema de tickets de una empresa del gremio azucarero de Guatemala y lograr un mejor control de los permisos, se determinó que el entorno de desarrollo integrado más adecuado para la elaboración del proyecto es NetBeans, por estar diseñado para limitar los errores de codificación y optimizar productividad.

En lo que respecta a los Objetivos Específicos planteados, en su orden, se concluye que:

1. La propuesta de integración del módulo Apache Shiro se puede considerar una buena alternativa para la autenticación y autorización dinámica para una empresa del gremio azucarero de Guatemala, debido a que la información es uno de los activos más importantes que una empresa puede tener.

2. La funcionalidad del login consiste en solicitar al usuario que ingrese un usuario y contraseña con el fin de que el módulo Apache Shiro sea el que gestione la autenticación y autorización de forma dinámica las credenciales y permisos del usuario. En el prototipo de la aplicación se logró integrar el módulo Shiro Cas lo cual permitió determinar que el riesgo de suplantación de identidad de los usuarios pasó a ser del 30%.

3. Se hizo una prueba general del prototipo de la aplicación con el módulo Shiro Cas integrado la cual da a conocer que se llega a reforzar la seguridad en un 70% en el sistema de tickets para una empresa del gremio azucarero de Guatemala.

IX. RECOMENDACIONES

A continuación, se enumeran una serie de recomendaciones en base a las conclusiones a que se llegó:

1. Se recomienda tener el entorno de desarrollo integrado NetBeans y sus herramientas de desarrollo para evitar posibles incompatibilidades.

2. Debido a que la información es uno de los activos más importantes que tiene una empresa del gremio azucarero de Guatemala, se recomienda considerar la integración de más módulos de seguridad en el sistema de tickets, a razón de que los métodos de suplantación de identidad con el tiempo van cambiando.

3. Se recomienda agregar a la funcionalidad de login y autenticación, la técnica de CAPTCHA. Ésta se trata de una prueba de desafío-respuesta en donde los usuarios deben de ingresar lo que se les solicite, con el fin de determinar si es humano o no.

4. Por último, se recomienda que la empresa del gremio azucarero de Guatemala brinde una educación a los usuarios que permita reducir aún más el riesgo de suplantación de identidad.

X. BIBLIOGRAFÍA

- Application security with apache shiro (Les Hazlewood. Marzo 14, 2022) Encontrado en: https://www.infoq.com/articles/apache-shiro/

- Amenazas y vulnerabilidades, ¿cuáles son las diferencias? Encontrado en: https://ciberseguridad.com/amenzas/

- Amenazas y vulnerabilidades, ¿cuáles son las diferencias? Encontrado en: https://ciberseguridad.com/amenzas/

- Apache Shiro (Chistopher Lynch) Encontrado en: https://home.cs.colorado.edu/~kena/classes/5448/f12/presentation-materials/lynch.pdf

- Apache Shiro website Encontrado en: https://shiro.apache.org/

- Apache Shiro Architecture Encontrado en: https://shiro.apache.org/architecture.html

- Ciberseguridad (infosecurity México) Encontrado en: https://www.infosecuritymexico.com/es/ciberseguridad.html#:~:text=La%20ciberseguridad%20es%20el%20conjunto,m%C3%B3viles%2C%20redes%20y%20sistemas%20electr%C3%B3nicos

- Descripción de los conceptos básicos de la ciberseguridad Encontrado en: https://learn.microsoft.com/es-es/training/paths/describe-basic-concepts-of-cybersecurity/

- Descripción de los conceptos básicos de la ciberseguridad / Descripción de amenazas, ataques y mitigaciones básicos de ciberseguridad. Encontrado en: https://learn.microsoft.com/es-es/training/modules/describe-basic-cybersecurity-threats-attacks-mitigations/2-describe-what-is-cybersecurity

- Descripción de los conceptos básicos de la ciberseguridad / Descripción de la autenticación y la autorización en ciberseguridad. Encontrado en: https://learn.microsoft.com/es-es/training/modules/describe-authentication-authorization-cybersecurity/2-define-authentication

- Diferencias entre amenaza, vulnerabilidad y riesgo (Febrero 22, 2022) Encontrado en: https://www.ambit-bst.com/blog/diferencias-entre-amenaza-vulnerabilidad-y-riesgo

- Framework (Edix. Agosto 26, 2022) Encontrado en: https://www.edix.com/es/instituto/framework/

- Framework de seguridad: Apache Shiro. Encontrado en: https://1library.co/article/framework-seguridad-apache-shiro-consideraciones-arquitectura-software-nivel.eqo55oky

- Integrating apache shiro with CAS SSO server Encontrado en: https://shiro.apache.org/cas.html

- Java EE. Encontrado en: https://www.oracle.com/java/technologies/java-ee-glance.html

- Nebrass Lamuochi (2016) *Pairing Apache Shiro and Java EE*. Encontrado en: https://www.infoq.com/minibooks/apache-shiro-ee-7/?itm_source=minibooks&itm_medium=link&itm_campaign=more_guides

- Oracle NetBeans IDE. Encontrado en: https://www.oracle.com/mx/tools/technologies/netbeans-ide.html

- ¿Qué es la ciberseguridad? Encontrado en: https://www.cisco.com/c/es_mx/products/security/what-is-cybersecurity.html#~temas-relacionados

- ¿Qué es la autorización y control de acceso? (Dave Piscitello. Diciembre 2, 2015) Encontrado en: https://www.icann.org/es/blogs/details/what-is-authorization-and-access-control-2-12-2015-es

- Que es Framework. Encontrado en: https://www.arimetrics.com/glosario-digital/framework

- ¿Qué es un entorno de desarrollo integrado (IDE)? Encontrado en: https://www.servicenow.com/es/now-platform/what-is-ide.html

- ¿Qué es un IDE? Encontrado en: https://aws.amazon.com/es/what-is/ide/

- ¿Qué es el kit de desarrollo de Java (JDK) Encontrado en: https://es.theastrologypage.com/java-development-kit

- ¿Qué es el lenguaje de programación Java? Encontrado en: https://www.ictea.com/cs/index.php?rp=/knowledgebase/8790/iQue-es-el-lenguaje-de-programacion-JAVA.html

- ¿Qué es el lenguaje de programación Java? Encontrado en: https://www.ictea.com/cs/index.php?rp=/knowledgebase/8790/iQue-es-el-lenguaje-de-programacion-JAVA.html

- ¿Qué es Java? Conoce las particularidades de este lenguaje de programación (Rock Content. Junio 5, 2019) Encontrado en: https://rockcontent.com/es/blog/que-es-java/

- Tipos de vulnerabilidades en ciberseguridad. Encontrado en: https://www.campusciberseguridad.com/blog/item/118-tipos-de-vulnerabilidades-en-ciberseguridad

- Triángulo de seguridad informática: Qué es y sus objetivos (Dani Flores. Agosto 25, 2021) Encontrado en: https://openwebinars.net/blog/triangulo-de-seguridad-informatica-que-es-y-sus-objetivos/#:~:text=El%20tri%C3%A1ngulo%20de%20la%20seguridad%20inform%C3%A1tica%20consta%20de%3A%20Confidencialidad%2C%20Integridad,los%20datos%20que%20se%20manejan

- What is a security framework? Encontrado en: https://www.techslang.com/definition/what-is-a-security-framework/

- What is a vulnerability? Encontrado en: https://www.techslang.com/definition/what-is-vulnerability/