ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

"GUÍA DE BUENAS PRÁCTICAS DE DESARROLLO DE APLICACIONES WEB SEGURAS APLICADO AL SISTEMA CONTROL DE NUEVOS ASPIRANTES EMPRESA GRUPO LAAR"

TESIS DE GRADO

Previa a la obtención del Título de

INGENIERO EN SISTEMAS INFORMÁTICOS

Presentado por:

ELVIRA NATALY YÁNEZ ROMERO

RIOBAMBA – ECUADOR

2014

Desde lo más profundo de mí ser agradezco a Dios por bendecirme y ser una guía constante en mi vida, a mis padres por su amor infinito, su apoyo incondicional y sobre todo por hacer de mí una mujer real con valores y principios bien marcados. Agradezco infinitamente a mi director de tesis Ing. Gloria Arcos por su tiempo y paciencia, por guiarme de una manera adecuada en el desarrollo de este trabajo de tesis. Y antes de finalizar mi más profundo agradecimiento a todos mis amigos por todo el cariño brindado, el apoyo recibido y la paciencia que siempre me tuvieron

Elvira Nataly Yánez Romero

Dedico este trabajo de tesis a mi familia por todo el apoyo y el esfuerzo que siempre realizaron por mí, en especial a mis padres Pablo y Julia que con su amor, trabajo, cuidados y desvelos me lo han dado todo para que esta primera meta se cumpla. A mis hermanos Pablo, Carlitos y Maita que siempre tuvieron una palabra de aliento, un abrazo lleno de amor para brindarme ánimo y no dejarme caer en los momentos difíciles. A mis Tíos Mario y Susana que han sido mis como mis segundos padres. A mis primos Daniel y Pauly por todo el apoyo y los consejos brindados. A mis amigos Anita, Fernanda, Belén, Karla, Eulalia y Carlos por su valiosa amistad y sobretodo todo a Dios por la llenarme de fe y fortaleza en cada momento de mi vida y permitiré cumplir con uno de mis objetivos.

Elvira Nataly Yánez Romero.

FIRMAS DE RESPONSABILIDAD

	FIRMA	**FECHA**
Ing. Iván Menes **DECANO DE LA FACULTAD** **INFORMÁTICA Y ELECTRÓNICA**	**.....**	**.....**
Ing. Jorge Huilca **DIRECTOR DE LA ESCUELA** **INGENIERÍA EN SISTEMAS**	**.....**	**.....**
Ing. Gloria Arcos **DIRECTORA DE TESIS**	**.....**	**.....**
Ing. Jorge Menéndez **MIEMBRO DEL TRIBUNAL**	**.....**	**.....**
DIRECTOR CENTRO DE **DOCUMENTACIÓN**	**.....**	**.....**

“Yo ELVIRA NATALY YÁNEZ ROMERO soy responsables de las ideas, doctrinas y resultados expuestos en esta tesis; y el patrimonio intelectual de la Tesis de Grado pertenece a la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO”.

_____

ELVIRA NATALY YÁNEZ ROMERO

ÍNDICE DE ABREVIATURAS

LDPA	Protocolo Ligero de Acceso a Directorios
SQL	Lenguaje de Consulta Estructurado
XSS	Secuencias de comandos en sitios cruzados
CSRF	Falsificación de petición en sitios cruzados
URL	Localizador Uniforme de Recursos
PHP	Procesador de Hipertextos
MYSQL	Lenguaje de Consulta Estructurado
CSS	Hojas de Estilo en Cascada
ERP	Planificación de recursos empresariales
IT	Tecnologías de la Información
API	Interfaz de Programación de Aplicaciones.
DOM	Modelo de Objetos del Documento.
ESPOCH	Escuela Superior Politécnica de Chimborazo.
GUI	Interfaz Gráfica de Usuario.
HTML	Lenguaje de Marcado de Hipertexto.
XML	Lenguaje de Marcas Extensible.
CSS	Hoja de Estilo en Cascada
XP	Programación Extrema
HTML	Lenguaje de Marcado de Hipertexto
GSAW	Guía Segura de Aplicaciones Web

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

INTRODUCCIÓN

La evolución de la tecnología hoy en día es muy amplia y de igual manera las actividades que se realizan con ayuda del internet, los usuarios buscan nuevas y mejores experiencias en la red, por lo cual muchas empresas, instituciones, etc en su afán de satisfacer estas necesidades ofrecen una gran variedad de servicios atreves de las aplicaciones web. En la actualidad, la seguridad en las aplicaciones web es un concepto que comienza a ser de vital importancia ya que de la misma manera que el avance de la tecnología nos brinda muchos beneficios y es una gran ayuda en diversas actividades, existen usuarios maliciosos que toman ventaja de esto y obteniendo acceso a información sensible, modificando o borrando datos importantes y haciendo que la imagen de la empresa se vea afectada ante sus usuarios, provocando que de los atacantes a las aplicaciones web también se hayan incrementado.

De esta manera es que el presente trabajo de tesis previo a la obtención de título de Ingeniería en Sistemas Informáticos trata sobre la realización de una Guía de Buenas Prácticas de Desarrollo de Aplicaciones Web Seguras se enfocó principalmente en las vulnerabilidades de manejo de sesiones: autenticación de sesiones y validación de sesiones, y la vulnerabilidad de intérprete de inyección: inyección de código e inyección SQL.

El escenario de desarrollo es el Área de Recursos Humanos de la Empresa "GRUPO LAAR" donde se vió la necesidad de implementar una aplicación web que permita llevar un control de los nuevos aspirantes a la empresa.

Para automatizar el proceso se desarrolló una solución informática con la cual se pueda ofrecer un correcto tratamiento de la información brindando un mejor desenvolvimiento del Área de Recursos Humanos.

En el Capítulo I Marco referencial, se detalla los antecedentes, la justificación de la investigación los objetivos generales y específicos y se plantea la hipótesis la cual se comprueba al final de la investigación.

El Capítulo II Marco Teórico, comprende el estudio y definiciones de los conceptos, pilares y principios de seguridad, las herramientas de verificación de vulnerabilidades y las metodologías de riesgos toda esta información sirve para el desarrollo de esta investigación.
En el Capítulo III Guía de buenas prácticas para desarrollo de aplicaciones web seguras presenta el análisis de las metodologías de riesgos, la elaboración de la guía de buenas prácticas GSAW.
En el Capítulo IV comprende el desarrollo del Sistema de Control de Nuevos Aspirantes en la Empresa "Grupo Laar" utilizando buenas prácticas de desarrollo web seguro GSAW.
En el Capítulo V comprende el análisis de resultados que se obtuvieron en la realización de la investigación.
.

CAPÍTULO I
MARCO REFERENCIAL

1.1. Antecedentes

La constante, casi frenética, evolución que está sufriendo el desarrollo de Internet y el aumento de usuarios que lo utiliza, está causando un gran auge en el desarrollo de aplicaciones web.

La mayoría de los problemas de seguridad en las aplicaciones son el resultado de escritura defectuosa de código, debemos entender que desarrollar aplicaciones web seguras no es una tarea fácil, ya que requiere por parte del programador, no únicamente mostrar atención en cumplir con el objetivo funcional básico de la aplicación, sino una concepción general de los riesgos, las vulnerabilidades que puede correr la información contenida, solicitada y recibida por el sistema, existen diversas vulnerabilidades como son: Phishing; autenticación; autorización; manejo de sesiones con ataques en secuestro de sesión, autenticación de sesión, validación de sesión, sesión pre programada; intérprete de inyección sus ataques son en inyección XML, inyección de código, inyección LDPA, inyección SQL; validación de datos sufre ataques de secuencia de comandos en sitios

cruzados XSS, referencia directa insegura a objetos, falsificación de peticiones en sitios cruzados CSRF, defectuosa configuración de seguridad, almacenamiento criptográfico inseguro, falla de restricción de acceso a URL, protección insuficiente en la capa de transporte, redirecciones y reenvíos no validados.

Desgraciadamente, la mayoría de las amenazas son invisibles hasta que es demasiado tarde, ya que manejan activos tan intangibles como los datos y la información que se obtiene de estos, aunque existen muchas publicaciones que permiten formar un criterio sobre el tema, no existen acuerdos básicos sobre lo que se debe o no se debe hacer, y lo que en algunas publicaciones se recomienda, en otras es atacado, lo que se pretende es cerrar ciertas "puertas de entrada" que no desearíamos tenerlas abiertas para usuarios fraudulentos, que pueden afectar el correcto funcionamiento de las aplicaciones web.

Una de las decisiones más importantes a la que nos enfrentamos en el inicio de todo proyecto es decidir ¿cómo vamos a trabajar? No es una decisión trivial, para tener un trabajo exitoso se debe seguir un proceso ordenado y de calidad para que los resultados obtenidos por la investigación sean eficientes, den valor y ayuden a la empresa a la toma de decisiones.

Es necesario que se incorporen buenas prácticas para proteger el entorno de información, y prevenir aún más la posibilidad de ser blanco de cualquiera de las amenazas, que constantemente buscan sacar provecho de las debilidades, para ello es importante conocer los peligros latentes, y cómo detenerlos a través de mecanismos de prevención, es por eso qué hacer de las aplicaciones web seguras una realidad, conlleva la adaptación de "buenas prácticas", que son aquellas medidas de corrección o mejoramiento, son acciones o iniciativas con repercusiones favorables en cuanto a la mejora de calidad de desarrollo web.

Entendiendo la responsabilidad y dedicación para el desarrollo de aplicaciones web seguras de calidad y a pesar de la existencia de mucha información como son: Normativas de seguridad web, Manuales de Desarrollo Web OWASP , Modelos para Desarrollar aplicaciones web; en el mercado aún no existe una guía de buenas prácticas para el desarrollo de aplicaciones web seguras, es importante incorporar en nuestras actividades esquemas de investigación al respecto y normativas que certifiquen la seguridad de

nuestros productos para verlo reflejado desde el punto de vista de un lenguaje específico, como desarrollar una aplicación web segura utilizando PHP y MySQL.

El control de nuevos aspirantes a la empresa “Grupo LAAR” es el acto en el cual el área de Recursos Humanos realiza dos pruebas de ingreso para los aspirantes que desean solicitar trabajo a los diferentes puestos vacantes de las empresas que conforma “Grupo LAAR”.

La empresa “Grupo LAAR” está conformada por LAAR SEGURIDAD, LAAR COM y LAAR COURIER, en el área de Recursos Humanos se encuentran tres personas cada una se encarga de llevar acabo las actividades concernientes a cada empresa, en el control de nuevos aspirantes.

En la actualidad la empresa cuenta con el Sistema Control de Nuevos Aspirantes, el cual posee diversas vulnerabilidades como: inyección SQL, autenticación de sesiones, validación de sesiones, inyección de código, las mismas que fueron estudiadas en el presente proyecto de tesis; además que la aplicación no cumple con todo el proceso de control de nuevos aspirantes, lo cual causa un gran problema para el personal del área de recursos humanos de la empresa.

Para que tenga un buen control de nuevos aspirantes, se mejoró el sistema existente, desarrollando una aplicación web segura que satisfaga las necesidades actuales, se utilizó PHP y MYSQL ya que provee una forma más sencilla y mejor de desarrollar aplicaciones web seguras, así como una estructura más sólida para la misma y con el propósito de obtener buenos resultados en base a la seguridad se ve la necesidad de implementar buenas prácticas para determinar posibles vulnerabilidades en los procesos de desarrollo, permitiendo realizar cambios y mejoras en las aplicaciones web, de una forma continua, sabiendo que las buenas prácticas son formas de garantizar que se está produciendo el menor impacto posible.

1.2. Justificación

1.2.1. Justificación Teórica

Analizar los riesgos existentes en una aplicación web, tomar acciones con respecto a sus causas es un factor muy importante, ya que los mismos constituyen un elemento crítico para el éxito, ignorar las amenazas que acechan a la aplicación web, es como jugar a la ruleta rusa: se puede ser afortunado durante algún tiempo, pero tarde o temprano esa fortuna terminará.

La mayoría de Empresas a veces no cuentan con normas, patrones o estándares de desarrollo de software, las entidades que las poseen no saben cómo utilizarlas de forma óptima, la mejor forma es a través de buenas prácticas de desarrollo de aplicaciones web seguras, ayuda en la obtención de un sistema eficiente, esto se constituye como una herramienta indispensable para la mejora continua, pues provee un sentido de necesidad de cambio, así como también al estar los procesos de desarrollo bajo constante medición se vuelven más eficientes, en conclusión la utilización de estos instrumentos nos conduce a la realización e identificación de mejores prácticas en el desarrollo web, permitiendo así obtener una aplicación web eficiente y segura, pero el desarrollo de la misma es muy amplio ya que requiere realizar un estudio para comprender los puntos vulnerables de la seguridad.

Dentro de la Empresa “Grupo LAAR” la seguridad en el desarrollo de sus aplicaciones es algo esperado, y no un complemento caro o algo dejado de lado, sin embargo las vulnerabilidades de las aplicaciones web abren la puerta a la explotación de información corporativa sensible, a la interrupción del servicio y al robo o alteración de información importante, es por eso que prevenir las vulnerabilidades de manejo de sesiones e intérprete de inyección se vuelve una tarea importante.

En cuanto a manejo de sesiones se pretende prevenir los ataques de:

- ✓ Autenticación de sesiones
- ✓ Validación de sesiones.

De esta manera poder asegurar que los usuarios autenticados tengan una robusta y criptográficamente segura asociación con sus sesiones, se hagan cumplir los controles de autorización.

En cuanto a intérprete de inyección se pretende prevenir los ataques de:

- ✓ Inyección de código
- ✓ Inyección SQL.

De esta manera se desea garantizar que las aplicaciones sean seguras de manipulación de parámetros y contra intérpretes comunes.

1.2.2. Justificación Práctica

En vista de la necesidad del realizar aplicaciones web seguras y de calidad se cree de vital importancia realizar una aplicación web segura que permitió automatizar el Control de Nuevos Aspirantes en el área de Recursos Humanos de la Empresa "Grupo LAAR" ya que, así se tiene un buen manejo de la información que se emplea, permitiendo que el trabajo que realiza el área de recursos humanos en base al control de nuevos aspirantes se reduzca, realizándose de una manera más eficiente y segura, lo que permite que los aspirantes accedan a esta aplicación para rendir las evaluaciones de ingreso y conocer de inmediato el resultado de su proceso, de la misma manera se permite poner en marcha el uso de buenas prácticas para su desarrollo, haciendo que este sea aceptable y a su vez se convierta en un punto de partida para futuras aplicaciones web seguras.

La programación deficiente conduce a tener vulnerabilidades futuras, es por eso que para la realización del sistema de control de nuevos aspirantes utilizaremos la plataforma PHP y como motor de base de datos MYSQL.

PHP es una plataforma robusta y gratuita que cumple con todos los estándares de calidad actuales, utiliza su propio sistema de administración de recursos, en cuanto a la seguridad PHP provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo.ini, lo cual nos permite desarrollar proyectos escalables lo que facilita la tarea del desarrollador.

MYSQL es Open Source lo que representa que el usuario puede usar libremente y hacer ajustes en el código, para maximizar su funcionamiento, anexando mejoras a la base de datos, logrando así una baja probabilidad de corromper datos.

De esta manera el presente proyecto de tesis está enfocado en realizar una guía de buenas prácticas para desarrollar una aplicación web segura del sistema de control de nuevos aspirantes de la Empresa "Grupo LAAR" y lograr de esta manera disminuir las vulnerabilidades de manejo de sesiones e intérprete de inyección

1.3. Objetivos

1.3.1. Objetivo General

- ✓ Realizar una guía de buenas prácticas para el desarrollo de aplicaciones web segura para disminuir vulnerabilidades de manejo de sesiones e intérprete de inyección, aplicado al sistema control de nuevos aspiraste Empresa "Grupo LAAR"

1.3.2. Objetivos Específicos

- ✓ Estudiar los conceptos, recursos, herramientas de verificación y vulnerabilidades de manejo de sesiones e intérprete de inyección de las aplicaciones web seguras.
- ✓ Analizar normativas, tutoriales, referencias que permitan recoger información sobre las aplicaciones web seguras.
- ✓ Proponer buenas prácticas para el desarrollo de aplicaciones web segura
- ✓ Aplicar las buenas prácticas en el desarrollo del sistema control de nuevos aspirantes en la empresa "Grupo LAAR"
- ✓ Verificar si las vulnerabilidades de manejo de sesiones e intérprete de inyección disminuyen al aplicar buenas prácticas en el sistema control de nuevos aspirantes en la empresa "Grupo LAAR"

1.4. Hipótesis

La implementación de una guía de buenas prácticas para el desarrollo de aplicaciones web seguras disminuirá vulnerabilidades de manejo de sesiones e intérprete de inyección del sistema control de nuevos aspirantes en la empresa “Grupo LAAR”.

CAPÍTULO II
MARCO TEÓRICO

2. Aplicación web

Santiago Bandiera [1] en su trabajo nos dice que al inicio el Internet estaba compuesto solamente por sitios web, que estos sitios eran fundamentalmente repositorios de información con documentos estáticos y los navegadores web fueron inventados con el propósito inicial de recuperar esta información para mostrársela gráficamente a los usuarios, nos dice también que la mayoría de los sitios no autenticaban a los usuarios ya que esto no era necesario y que cada uno de los usuarios era tratado de la misma manera y tenía acceso idéntico a la información, por lo cual no existían grandes amenazas de seguridad en el almacenamiento y administración de un sitio web y si algún intruso realizaba un ataque al servidor, por lo general no obtenía acceso a información restringida porque casi toda la información almacenada estaba disponible a la vista pública.

Los ataques más comunes eran modificar los archivos almacenados en el servidor para cambiar el contenido del sitio web, o usar la capacidad de almacenamiento del servidor y el

ancho de banda para distribuir software maligno, en la actualidad, la mayoría de los sitios en la web son aplicaciones con un grado significativo de funcionalidades y flujos de información entre el servidor y el navegador web, estas aplicaciones permiten registro y autenticación, transacciones financieras, búsquedas y niveles de autorización por usuarios, entre otras operaciones.

Actualmente la situación ha cambiado los intrusos maliciosos y los ataques que se realizan a las aplicaciones web son más frecuentes y de diversas maneras haciendo que la seguridad sea un asunto de suma importancia y se lo deba considerar al inicio del desarrollo de las mismas y no dejarlo de lado ya que puede acarrear problemas después.

2.1. Seguridad Informática

La seguridad informática, es el área de la informática que se enfoca en la protección de la infraestructura computacional y todo lo relacionado con ésta (incluyendo la información contenida). Para ello existen una serie de estándares, protocolos, métodos, reglas, herramientas y leyes concebidas para minimizar los posibles riesgos a la infraestructura o a la información, podemos entender como seguridad un estado de cualquier tipo de información (informático o no) que nos indica que ese sistema está libre de peligro, daño o riesgo. Se entiende como peligro o daño todo aquello que pueda afectar su funcionamiento directo o los resultados que se obtienen del mismo. Para la mayoría de los expertos el concepto de seguridad en la informática es utópico porque no existe un sistema 100% seguro [2].

2.2. Seguridad en las Aplicaciones Web

La seguridad en las aplicaciones web está enfocada en la protección de la información contenida o circulante en la web, y con el fin de lograr esto se utiliza una serie de estándares, protocolos, métodos, reglas, herramientas.

El desarrollo de aplicaciones web seguras hoy en días surge por el interés de lograr que las aplicaciones sean confiables y de esta manera logren brindar un servicio eficiente tanto para las empresas que las desarrollas así como para los usuarios que la utilizan.

2.2.1. Tópicos en las Seguridades en las aplicaciones web.

Día a día el desarrollador se encuentra con nuevos retos y desafíos, la seguridad en una aplicación web forma parte de estos ya que en la actualidad se convirtió en una tarea muy significativa y de vital importancia, para explayarnos sobre este tema definiremos otros conceptos ampliamente usados y fundamentales en este trabajo de tesis. [1]

- ✓ **Riegos**

Los riesgos son los incidentes o las fallas que se pueden presentar en la realización de alguna actividad, produciendo preocupación por parte del interesado.

- ✓ **Vulnerabilidad**

Una vulnerabilidad es la fragilidad de algo o alguien de ser atacado, es encontrarse con algún desperfecto ya sea en el diseño, implementación o ejecución de un sistema el mismo que puede ser explotado.

- ✓ **Debilidad**

Inexactitud en el software que en las condiciones apropiadas, puede contribuir a la aparición de una vulnerabilidad del mismo.

- ✓ **Ataque**

El ataque es la irrupción de una aplicación web de una manera deliberada y con la cual se logra violar la seguridad de las mismas, son las técnicas que el atacante utiliza para explotar una vulnerabilidad.

- ✓ **Amenaza**

Una amenaza se manifiesta a partir de las vulnerabilidades, son las acciones que pueden atentar contra la seguridad de una aplicación web.

- ✓ **Contramedida**

Una contramedida son las tareas, maniobras, procedimiento o cualquier otra medida destinada a reducir una vulnerabilidad y de la misma manera los ataques que puedan surgir.

2.2.2. Principios básicos de seguridad

La Organización OWASP menciona que los principios básicos de seguridad están enfocadas en la validación de entradas y salidas de información, en los diseños simples, en la reutilización de componentes, en la defensa en profundidad que la seguridad gracias al desconocimiento no funciona, en la verificación de privilegios y en ofrecer la mínima información siendo estos los principios que cualquier aplicación o servicio web debe cumplir, a continuación se detalla cada uno de estos. [3]

- ✓ **Validación de la entrada y salida de información**

  Son estas el principal mecanismo que un usuario malicioso ocupa para perjudicar nuestra aplicación es por esto que se debe verificar que toda entrada o salida del sistema sea adecuada.

- ✓ **Diseños simples**

  La seguridad que se diseña debe ser lo más sencilla posible sin que los usuarios tengan complicaciones a la hora de emplearla.

- ✓ **Utilización y reutilización de componentes de confianza**

Se debe tener en claro que si tenemos componentes que resuelven problemas de una manera correcta y adecuada lo mejor que podemos hacer es utilizarlo.

- ✓ **Defensa en profundidad**

Contar con la de seguridad necesaria ya que en caso de que algún componente del sistema tenga alguna falencia ante un determinado evento, cuente con otro componente de detectarlo.

- ✓ **La seguridad gracias al desconocimiento no funciona**

Se debe emplear unos mecanismos de seguridad correctos y no solo pretender resolver el problema ocultando las cosas ya que en un plazo largo o corto de tiempo la falencia volverá a aparecer.

- ✓ **Verificación de privilegios**

Los sistemas deben diseñarse para que funcionen con los menos privilegios posibles.

- ✓ **Ofrecer la mínima información**

Se tiene que dar a conocer la mínima información ante una situación de error o una validación negativa.

2.2.3. Pilares de la Seguridad

La organización OWASP ha establecido que todos los principios de seguridad buscan el cumplimiento de los tres pilares básicos de la seguridad de la información que son la base para establecer de una manera adecuada la seguridad en nuestras aplicaciones web, los mismos que se describen a continuación: [3]

- ✓ **Confidencialidad**

  La Organización Internacional de Normalización o ISO (International Organization for Standardization) define confiabilidad como la el acceso a la información por parte únicamente de quienes estén autorizados.

  Se debe tener cuidado con la información que se posee ser reservado y solo permitir el acceso debido y de ser necesario mantener oculta la información que haya sido procesada.

- ✓ **Integridad**

  La integridad puede ser definida como la propiedad de la información que asegura que esta no ha sido alterada o destruida de manera no autorizada.

  La información tiene que ser respetada, ser tratarla de una manera leal, es decir que no se la debe destruir o modificar cuando esta ya fue procesada.

- ✓ **Disponibilidad**

  La disponibilidad es la cualidad o la habilidad de ser accesible cuando es necesario.

  Debe ser utilizable siempre que sea necesario para un usuario autorizado.

Es de esta manera que los tres pilares de seguridad deben ser balanceados de acuerdo a los requerimientos de la información, las tres cualidades son importantes, y nos demuestra que la seguridad requiere ser priorizada.

2.2.4. Tipos de Ataques de Seguridad.

El proceso de explotar vulnerabilidades y realizar ataques en las aplicaciones web como ya se mencionó ha crecido y sigue creciendo por lo cual se han realizado varios estudios sobre estas. El OWASP Top 10 2013 [4], se basa en 8 conjuntos de datos de 7 firmas especializadas en seguridad de aplicaciones, incluyendo 4 empresas consultoras y 3

proveedores de herramientas SaaS. Estos datos abarcan más de 500.000 vulnerabilidades a través de cientos de organizaciones y miles de aplicaciones. Las vulnerabilidades del Top 10 son seleccionadas y priorizadas de acuerdo a estos datos de prevalencia, en combinación con estimaciones consensuadas de explotabilidad, detectabilidad e impacto, la lista de estas vulnerabilidades se la visualiza en la FiguraII.1.

OWASP Top 10 – 2013 (Nuevo)
A1 – Inyección
A2 – Pérdida de Autenticación y Gestión de Sesiones
A3 – Secuencia de Comandos en Sitios Cruzados (XSS)
A4 – Referencia Directa Insegura a Objetos
A5 – Configuración de Seguridad Incorrecta
A6 – Exposición de Datos Sensibles
A7 – Ausencia de Control de Acceso a las Funciones
A8 – Falsificación de Peticiones en Sitios Cruzados (CSRF)
A9 – Uso de Componentes con Vulnerabilidades Conocidas
A10 – Redirecciones y reenvíos no validados
Fusionada con 2010-A7 en la nueva 2013-A6

Figura II. 1Top 10 OWASP
Fuente: Guía de Vulnerabilidades OWASP Top 10

A1 – Inyecciones

Vulnerabilidades de inyección de código, desde SQL o comandos del sistema hasta LDAP, ocurren cuando se envían datos no confiables a un intérprete como parte de un comando o consulta.

A2 – Pérdida de autenticación y gestión de sesiones

Comprende los errores y fallos en las funciones de gestión de sesiones y autenticación. Se produce cuando las funciones de la aplicación relacionadas con la autenticación y la gestión de sesiones no se implementan correctamente, lo que puede permitir a los atacantes comprometer contraseñas, claves, token de sesiones, o explotar otros problemas que podrían permitir asumir la identidad de otros usuarios.

A3 – Cross-Site Scripting (XSS)

Puede ser utilizado para robar información delicada, secuestrar sesiones de usuario, y comprometer el navegador, subyugando la integridad del sistema

A4 – Referencias directas inseguras a objetos

Errores al exponer partes privadas o internas de una aplicación sin control y accesibles públicamente. Ocurre cuando un desarrollador expone al exterior una referencia a un objeto de implementación interno, tal como un fichero, directorio, o base de datos.

A5 – Configuración de seguridad incorrecta

Más que un error en el código se trata de la falta o mala configuración de seguridad de todo el conjunto de elementos que comprende el despliegue de una aplicación web, desde la misma aplicación hasta la configuración del sistema operativo o el servidor web. Es decir, se refiere a la definición e implementación de una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor web, base de datos y plataforma. Todas estas configuraciones deben ser definidas, implementadas y mantenidas, ya que por lo general no son seguras por defecto. Esto incluye mantener todo el software actualizado, incluidas las librerías de código utilizadas por la aplicación.

A6 – Exposición de datos sensibles

Esta categoría surge de la fusión y ampliación de las anteriores A7 y A9. Muchas aplicaciones web no protegen adecuadamente datos sensibles tales como números de tarjetas de crédito o credenciales de autenticación. Los datos sensibles requieren de métodos de protección adicionales tales como el cifrado de datos almacenados mediante técnicas criptográficas adecuadas (p.ej. manteniendo el hash de la contraseña en vez de la

propia contraseña cifrada), así como también de precauciones especiales en el intercambio de datos con el navegador.

A7 – Ausencia de control de acceso a funciones

Surge de la ampliación de la anterior categoría 8, que trataba la falta de validación en el procesamiento de URLs que podrían ser usadas para invocar recursos sin los derechos apropiados o páginas ocultas. Ahora se refiere tanto a URLs como a los datos que se pasan a funciones de la propia aplicación.

A8 – Falsificación de Peticiones en Dominios Cruzados o Cross-site Request Forgery (CSRF).

Consistente en el desencadenamiento de acciones legitimas por parte un usuario autenticado, de manera inadvertida por este último y bajo el control de un atacante.

A9 – Utilización de componentes con vulnerabilidades conocidas

Se refiere al uso de componentes tales como librerías, frameworks y otros módulos de software, que en muchas ocasiones funcionan con todos los privilegios. Si se ataca un componente vulnerable esto podría facilitar la intrusión en el servidor o una pérdida de datos.

A10 – Redirecciones y reenvíos no validados.

Errores en el tratamiento de redirecciones y uso de datos no confiables como destino.

En este ranking establecido por OWASP enumera varias de las vulnerabilidades que se presentan en las aplicaciones web al no contar con una adecuada seguridad y que son de gran preocupación tanto para los desarrolladores como para las empresas, de igual forma

es un punto de partida para dar soluciones a varios de los problemas que acarrean las mismas, la investigación realizada se centra en prevenir las vulnerabilidades de Manejo de Sesiones como son : autenticación de sesiones y validación de sesiones e Intérprete de Inyección como son: inyección SQL e inyección de comandos.

2.3. Herramienta de Verificación

Las herramienta de verificación de aplicaciones web permite realizar una prueba de penetración, atacar a una determinada aplicación en la cual se desee comprobar si su desarrollo se lo ha realizado de una forma seguro y de ser el caso ayudar a determinar las posibles fallas que existan en la misma y de esta manera reparar o mejorar la aplicación web.

2.3.1. Análisis de la Herramienta de Verificación

El trabajo realizado por ToolsWatch Top ten de Herramientas de Seguridad del 2013 para programadores [5] como se muestra en la Tabla.II.I determina que existen diversas herramientas de verificación de vulnerabilidades, gracias al estudio realizado se toma como objeto de análisis las tres primeras las cuales son: OWASP ZAP, BeEF y Burp Suite.

Tabla II. I.Votación de Herramientas Verificación

HERRAMIENTA	VOTOS OBTENIDOS
OWASP(ZAP)	106 votos
BeEF	90 votos
Burp Suite	73 votos
PeStudio	60 votos
OWASP Xenotix	49 votos
Lynis	38 votos

Recon-ng	26 votos
Suricata	24 votos
WPScan	5 votos
O-Saft	5 votos

Fuente: Top10 mejores herramientas 2013

A continuación se analizan cada una de las herramientas seleccionadas en base a sus características, ventajas y desventajas para determinar la que mejor se adapte a la investigación.

- ✓ **Herramienta OWASP ZAP**

ZAP es una poderosa herramienta para realizar ataques de penetración que permite analizar aplicaciones web para buscar sus vulnerabilidades, con muy diversos fines, ZAP se basa en la lista OWASP de las vulnerabilidades web más comunes para su desarrollo, por lo que incluye una gran cantidad de herramientas capaces de detectar casi cualquier vulnerabilidad que pueda existir en un sitio web. [6] A continuación se muestran las características, ventas y desventajas de las herramientas OWASP ZAP como se observa en TablaIII.II

Tabla II. II. Herramienta ZAP

Herramienta	OWASP Attack Proxy (ZAP)
Características	✓ Fácil uso para encontrar vulnerabilidades en aplicaciones web. ✓ Está diseñada para ser utilizada tanto por desarrolladores y probadores funcionales. ✓ ZAP ofrece escáneres automatizados, así como un conjunto de herramientas que le permiten encontrar las vulnerabilidades de seguridad de forma manual.
Ventajas	✓ ZAP es una herramienta gratuita, multiplataforma.

	✓ Capaz de detectar casi cualquier vulnerabilidad que pueda existir. ✓ Posibilidad de generar informes, e interconexión con otras aplicaciones para poder realizar análisis más potentes.
Desventajas	✓ ZAP es un proxy, lo que significa que está diseñado para ser ejecutado por el Tester.

- ✓ **Herramienta BeEF**

La BeEF es la abreviatura de The Browser Exploitation Framework. Es una herramienta de pruebas de penetración que se centra en el navegador web.

En medio de la creciente preocupación por los ataques procedentes de Internet en contra de los clientes, incluyendo clientes móviles, BeEF permite la prueba de intrusión profesional para evaluar la situación de seguridad actual de un entorno de destino mediante el uso de vectores de ataque del lado del cliente. [7] Como se observa en TablaII.III se analizó los aspectos de mayor relevancia acerca de la Herramienta BeEF.

Tabla II. III.Herramienta BeEF

Herramienta	BeEF (The Browser Exploitation Framework)
Características	✓ Es un marco modular que utiliza técnicas pioneras que proveen la posibilidad de realizar pruebas de intrusión. ✓ Se centra en el aprovechamiento de las vulnerabilidades del navegador para evaluar la postura de seguridad de un objetivo
Ventajas	✓ Proporcionar un sencillo pero poderoso vector de ataque contra máquinas remotas de una manera muy camuflada.
Desventajas	✓ Integra una serie de exploits manejados directamente por Mestasploi

	aunque es de fácil instalación se encuentra obsoleto.

- ✓ **Herramienta Burp Suite**

Burp Suite es una plataforma integrada para atacar aplicaciones web. Contiene todas las herramientas Burp con numerosas interfaces entre ellos diseñados para facilitar y acelerar el proceso de atacar a una aplicación. [8] Como se observa en TablaII.IV se analizó las características, ventajas y desventajas de la herramienta.

Tabla II. IV.Burp Suite

Herramienta	Burp Suite
Características	✓ Permite realizar test de intrusión en aplicaciones web ✓ Permite combinar técnicas manuales y automáticas para enumerar, analizar, atacar y explotar aplicaciones web ✓ Funcionar como proxy entre nuestro navegador e Internet
Ventajas	✓ Permite inspeccionar y modificar el tráfico entre el navegador y la aplicación de destino. ✓ Extensibilidad, que le permite escribir fácilmente sus propios plugins, para realizar tareas complejas y altamente personalizado ✓ Detecta de numerosos tipos de vulnerabilidad.
Desventajas	✓ Posee una versión gratuita limitada.

Una vez analizadas las características, ventajas y desventajas que posee cada una de las herramientas se concluye que en la investigación realizada OWASP ZAP ayuda a cumplir con los objetivos planteados ya que permite atacar una aplicación web de una forma

sencilla, posee varias herramientas que son muy útiles a la hora de realizar una penetración en una aplicación web, como lo es su escáner activo, spider ,etc; es gracias a esto que se logra detectar varias vulnerabilidades como autenticación de sesiones ,validación de sesiones, inyección de código, inyección SQL, ataque de fuerza fruta, inyección XSS entre otros de igual manera es importante resaltar que es completamente gratuita.

2.3.2. Herramienta OWASP Zed-ZAP

OWASP (Open Web Application Security Project) es el Proyecto abierto de seguridad de aplicaciones web dedicado a determinar y combatir las causas que hacen que el software sea inseguro, comenzó en el año 2001. Es una organización sin ánimo de lucro, se creó en 2004 para apoyar los proyectos e infraestructura de OWASP.
OWASP ZAP, es uno de los productos, sus siglas ZAP corresponden a Zed Attack Proxy.

Figura II. 2. Logo OWASP –ZAP
Fuente: Top10 mejores herramientas 2013
http://upload.wikimedia.org/wikipedia/commons/7/72/Fases_XP.jpg

OWASP Zed Attack Proxy (ZAP) es una herramienta de fácil uso para encontrar vulnerabilidades en aplicaciones web. Está diseñada para ser utilizada tanto por desarrolladores y probadores funcionales como por personas con una amplia gama de experiencia en seguridad. Permite automatizar las pruebas y también facilita un número de herramientas para hacerlas manualmente". [9] Es una herramienta para realizar ataques de penetración, que permite analizar aplicaciones web para buscar sus vulnerabilidades, con diferentes utilidades.

- ✓ **Versión**

Su última versión es ZAP 2.3.1 publicada el 21/05/2013 la cual contiene el conjunto básico de funcionalidad, y se pueden añadir más funcionalidad en cualquier momento a través del ZAP mercado., con versiones de Windows, Linux y Mac OS

- ✓ **Características**

Se menciona las características de mayor importancia que posee la herramienta.

- ✓ **Proxy de interceptación**

  Permite ver todo el tráfico entre el navegador y el servidor web, dejando ver de forma sencilla las cabeceras y el cuerpo de los mensajes HTTP sin importar el método usado (HEAD, GET, POST).

- ✓ **Escaneo Activo**

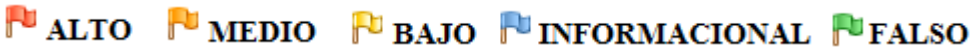
  Intenta encontrar potenciales vulnerabilidades usando ataques conocidos contra objetivos seleccionados. No debería ser utilizado en aquellas aplicaciones webs que no son de su propiedad.

- ✓ **Spider(Araña)**

  Spider es una herramienta que es usada para descubrir automáticamente nuevos recursos (URLs) en un sitio en particular. Spider visita estas URLs, identifica todos los hipervínculos en la página y los agrega a la lista de URLs a visitar y el proceso continuo recursivamente mientras se encuentran nuevos recursos.

- ✓ **Alertas**

  En Alertas se puede observar el resultado del escaneo de active con las vulnerabilidades existentes, la Alertas permiten ver las vulnerabilidades con el riesgo que estas poseen el cual puede ser:



- ✓ **Reportes**

  Genera reportes de alertas seleccionadas y de igual manera genera un reporte general de todas las alertas.

- ✓ **Búsqueda de Vulnerabilidades**

OWASP ZAP permite la búsqueda de vulnerabilidades automáticas, funciona de la siguiente forma:

- ✓ Se hace un recorrido de URL con el Spider
- ✓ Se realiza un escaneo activo de todas las URLs obtenidas en el spider
- ✓ Se analiza el contenido de cada URL y se muestran las alertas en función de la criticidad de la vulnerabilidad

Por todo lo mencionado sobre la herramienta OWASP ZAP se concluye que es el complemento ideal para los desarrolladores que buscan detectar el grado de vulnerabilidad que existen en las aplicaciones desarrolladas ya que sirve de ayuda para identificar las, lo cual permite aprender sobre estas y dar una solución a las mismas. Al pertenecer a los proyectos de OWASP sus versiones está en constante actualización y mejora lo cual hace de la misma una potente herramienta de penetración.

2.4. Metodologías de Riesgos

.

Existen diversas y diferentes metodologías que ayudar en la mitigación del riesgo, brindan ayuda a la hora de desarrollar una aplicación web, se estudiaron las metodologías de riesgos Ace Threat Analysis and Modeling, CORAS y PTA. A continuación se muestra cada una de ellas.

2.4.1. CORAS

CORAS (Consultative Objective Risk Analysis System), es un proyecto creado por la Unión Europea con el objetivo de proporcionar un framework orientado a sistemas donde la seguridad es crítica, facilitando el descubrimiento de vulnerabilidades de seguridad, inconsistencias, y redundancias. Aunque no es exactamente un frameworks para el modelado de amenazas, su uso orientado a tal fin, puede contribuir a la reducción de riesgos y la adopción de unas correctas contramedidas, por lo que me ha parecido interesante mencionarlo. [10]

Según la metodología propuesta por Coras, son siete pasos del proceso como se observa en la Figura II.3.

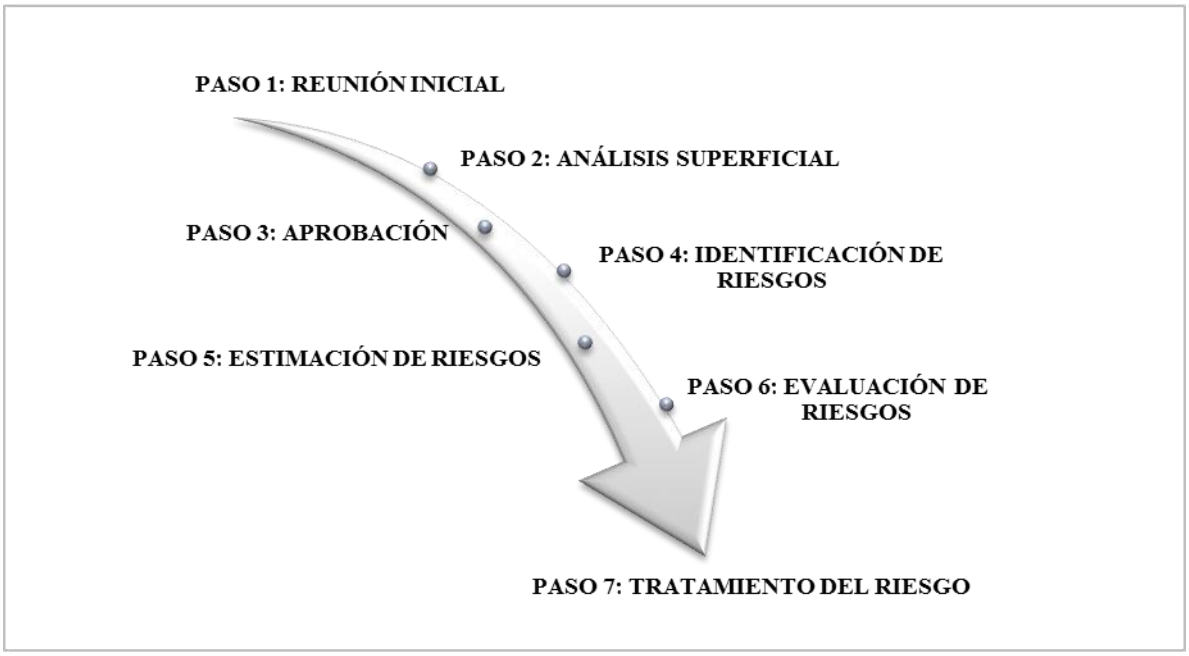


Figura II. 3. Metodología CORAS

PASO 1. Se realiza una entrevista para conocer cuáles son los objetivos principales del análisis. Se recopila la información necesaria en función de los requisitos del cliente.

PASO 2. Una segunda reunión con el cliente para comprobar que la información suministrada al analista ha sido suficiente

PASO 3. Se realiza una descripción más detallada del sistema a analizar, facilitando al cliente la documentación necesaria para su aprobación.

PASO 4. El analista, junto con las personas que mejor conocen el sistema, identifican todos los posibles incidentes no deseados, amenazas, vulnerabilidades y escenarios.

PASO 5. Se estiman las consecuencias y los valores de ocurrencia para cada uno de los posibles incidentes no deseados que se han identificado en los pasos anteriores.

PASO 6. Se proporciona al cliente un borrador del análisis para una primera revisión y corrección.

PASO 7. Se establece el tratamiento del riesgo, es decir, las contramedidas en función del coste/beneficio.

Cada uno de los pasos propuestos por CORAS establecen el tratamiento que se le debe dar a una aplicación para lograr su seguridad, pero muchos de los cuales son redundantes o se los podría ejecutar en uno solo y en cuanto a las contramedidas que se toman se deben estar enfocados no solo en base al coste/beneficio que la empresa tendrá si no también al respaldo de los datos ya que este es el activo intangible más importante que posee una empresa.

2.4.2. Ace Threat Analysis and Modeling

Esta metodología propuesta Microsoft ha desarrollado una metodología de análisis y modelado de amenazas, se basa en el uso de árboles de ataques para luego extrapolar las Amenazas y realizar una clasificación y un ranking de estas con el fin de priorizar las actuaciones necesarias para mitigar el riesgo. [10] Según la metodología Ace Threat Analysis and Modeling, son cinco pasos del proceso como se observa en la Figura II.4

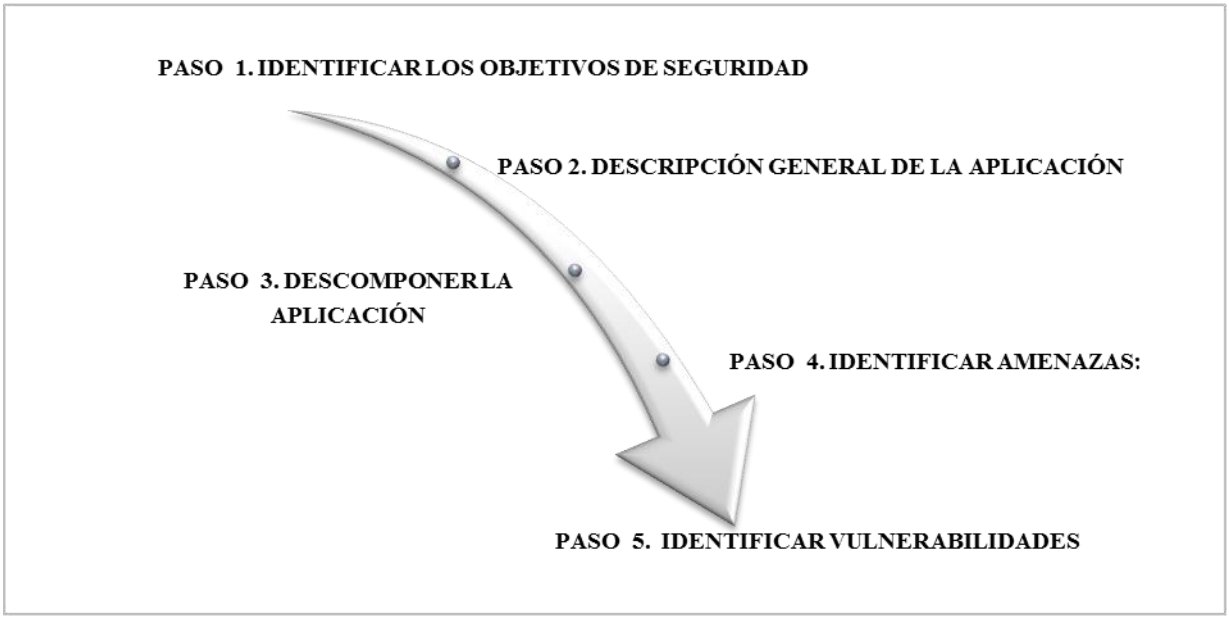


Figura II. 4. Metodología Ace Threat Analysis and Modeling

PASO 1. Determinar cuáles son los objetivos ayudará a cuantificar el esfuerzo se debe dedicar a los siguientes pasos.

PASO 2. Identificar los actores involucrados y las características más importantes de la aplicación facilitará la identificación de las amenazas más importantes.

PASO 3. Una vez que se conoce la arquitectura, es preciso identificar las funcionalidades y los módulos susceptibles de provocar un mayor impacto en la seguridad.

PASO 4. Con la información recopilada, y en función del contexto y el escenario de la aplicación, se procede a la identificación de las amenazas más importantes.

PASO 5. Revisar las diferentes capas de la aplicación para identificar los puntos débiles.

La metodología propuesta por Microsoft determina de una mejor manera cuales son los objetivos de seguridad que debe contener la aplicación web, identifica cuales son las amenazas que la misma tendrá, pero no en ninguno de sus pasos no está establecido las contramedidas que se debe tener.

2.4.3. PTA

La empresa PTA Technologies ha desarrollado su propia metodología que trata de solventar las limitaciones que según ellos tiene la metodología propuesta por Microsoft. Antes de comenzar el proceso de modelado, el analista debe familiarizarse con la aplicación y recopilar información útil para detectar las amenazas. [10] Según la metodología propuesta por PTA, los pasos del proceso son 4, como se observa en la Figura II.5.

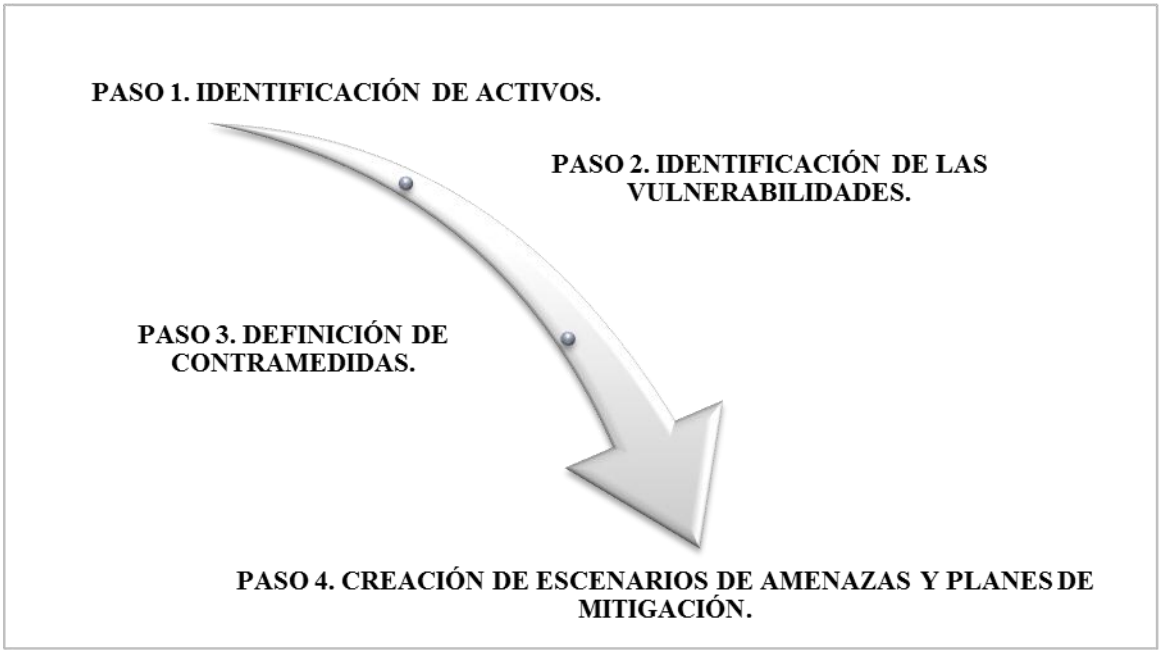


Figura II. 5.Metodología PTA

PASO 1. Se determina cuáles son los activos de mayor valor que deben ser protegidos ante posibles daños, con el fin de determinar las prioridades.

PASO 2. Dependiendo de la arquitectura, funcionalidad y la lógica del negocio se determina de forma iterativa cuales son las vulnerabilidades.

PASO 3. Se establecen las contramedidas a adoptar en función de las vulnerabilidades y del coste que supondrá su implementación.

PASO 4. Se identifican los distintos elementos de las amenazas.

La metodología PTA se enfoca en el estudio de activos que se debe proteger en la empresa en base a su funcionalidad, arquitectura y lógica del negocio con lo cual realiza planes de mitigación pero no trabaja en conjunto con la empresa todo está establecido por la persona que desarrolla la aplicación.

CAPÍTULO III
GUÍA DE BUENAS PRÁCTICAS PARA DESARROLLO DE APLICACIONES WEB SEGURAS

3. Desarrollo de la Guía de buenas Prácticas

Para desarrollar una aplicación web es imprescindible realizar acciones de prevención, corrección o mejoramiento también conocido como buenas prácticas, estas medidas deben ser lo más efectivas posibles para obtener una aplicación web segura y que de esta manera se vean beneficiados la empresa y los usuarios; garantizando el menor riesgo y que brinde eficiencia en el producto final.

Para la realización de esta guía de buenas prácticas se consideró conceptos importantes y concretos sobre seguridad en las aplicaciones web , se estudió metodologías de evaluación de riesgos como: CORAS, PTA y Ace Threat Analysis and Modeling y se determinó una herramientas de verificación de seguridad, con toda esta información se logró determinar

algunos pasos importantes que se deben emplear a la hora de desarrollar una aplicación web para lograr que sea segura, y de así reducir las vulnerabilidades que se pueden presentar a la hora de realizar la aplicación web.

3.1. Análisis de las metodologías de Riesgos

Las metodologías que se estudiaron ayudaron a determinar cuáles son los pasos adecuados para GSAW, En la Tabla.III.V se puede observar cada uno de los pasos que las metodologías de riesgos poseen los cuales de diferentes maneras brindan ayuda a la hora de mitigar las amenazas y los riesgos que se pueden encontrar al desarrollar una aplicación web. Es por esto que las coincidencias que se encontraron entre las tres metodologías fueron de ayuda para la realización de la guía.

Tabla III. V.Concurrencias de las Metodologías

	METODOLOGÍAS		
PASOS	**CORAS**	**Ace Threat Analysis and Modeling**	**PTA**
Identificar vulnerabilidades	✓	✓	✓
Identificar los objetivos de seguridad		✓	
Identificación de Activos	✓		
Descripción de la Aplicación		✓	✓
Creación de Escenario de Amenaza			✓
Aprobación del sistema	✓		
Estimación de Riesgos	✓		

Evaluación de Riesgos	✓		
Tratamiento del Riesgo	✓	✓	✓

Dentro del análisis de las concurrencias encontradas los pasos a considerar incluir en la guía de buenas prácticas de desarrollo web son:

- ✓ Tener una descripción de la aplicación ya que es aquí donde se determina lo que la aplicación realiza.
- ✓ Identificar las vulnerabilidades que posee determinada aplicación
- ✓ Evaluar los riesgos que estas vulnerabilidades presentan y dar un tratamiento a los mismos es decir proponer contramedidas.

Ya que cada uno de los pasos mencionados brindan un gran aporte en la búsqueda de seguridad en las aplicaciones web.

3.2. Descripción de la Guía Segura de Aplicaciones Web

La Guía se nombró a la guía como: GSAW "Guía segura de aplicaciones web".
GSAW es una guía de seguridad para el desarrollo de aplicaciones web inspirada en el análisis de metodologías de riesgos, la presente guía ofrece una visión general sobre las buenas prácticas en lo que se refiere al desarrollo de la aplicación; desde el reconocimiento de diferentes vulnerabilidades, hasta el tratamiento de los mismos.

3.3. Pasos de la Guía Segura de Aplicaciones Web

Los pasos de GSAW consisten en cuatro puntos principales como son: Definir, Modelar, Evaluar, Cuantificar y Validar como se observa en la FiguraIII.6, de la misma manera es importante resaltar que una vez que termina el paso Validar se regresa nuevamente a Evaluar haciendo que la guía sea recursiva.

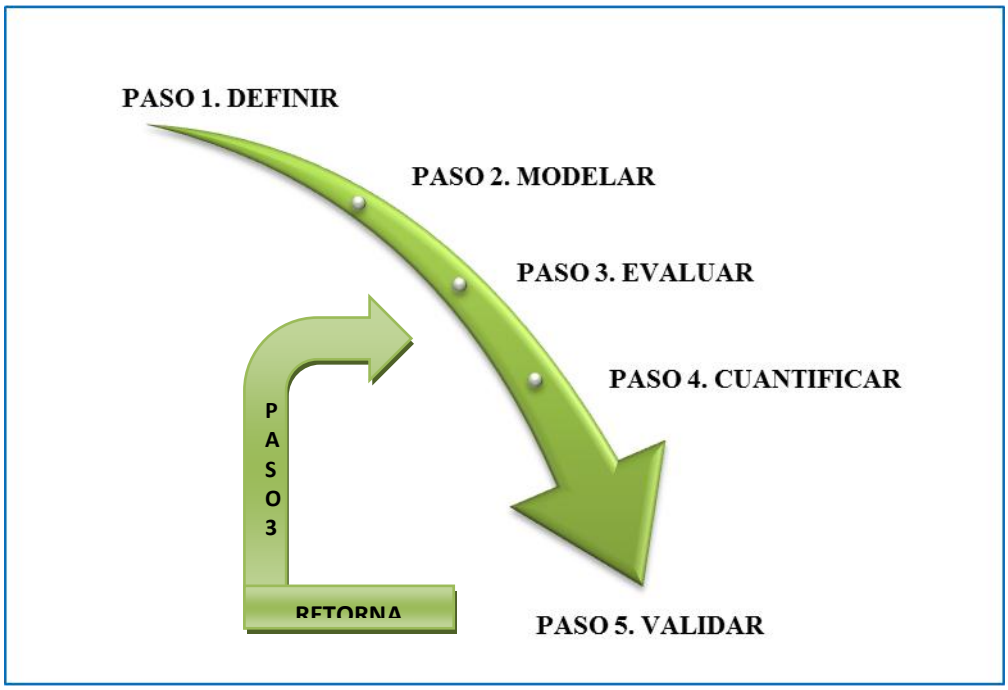


Figura III. 6. Pasos GSAW

Cada uno de los pasos de GSAW cuenta con diferentes actividades que se deben realizar para que la guía se cumpla, todas estas actividades se encuentran detalladas en la Tabla III.VI,

Tabla III. VI. Pasos y Actividades de GSAW

PASOS	ACTIVIDADES
1. Definir	✓ Identificación del Equipo de Trabajo ✓ Funcionalidad del Aplicación
2. Modelar	✓ Diseño de la Arquitectura ✓ Planteamiento de las posibles vulnerabilidades ✓ Planteamiento de seguridad

3. Evaluar	✓ Detección y listado de las Amenazas real
4. Cuantificar	✓ Determinar la severidad de Amenaza
5. Validar	✓ Contramedidas

3.3.1. Paso 1. Definir

En este paso de la guía de buenas prácticas se identifican roles, requerimientos, datos que se obtendrán mediante las entrevistas que se realizan con el cliente.

- ✓ **Identificación del equipo de trabajo**

Para que el desarrollo de una aplicación web se realice de una manera segura y eficiente es necesario tener bien definido cuál será el equipo de trabajo. La conformación del equipo de trabajo se la realiza con la certeza de que las personas que lo integren brinden el aporte necesario con sus conocimientos, habilidades, capacidades, información y así llegar a realizar un trabajo en conjunto acorde a las necesidades de la empresa, a continuación se describe la conformación del equipo de trabajo:

Líder de proyecto: es el responsable del éxito del proyecto es el encargado la planificación y administra su ejecución, efectúa un control estricto el sistema.
Gerente del producto: es la persona responsable de dirigir el sistema, establece la arquitectura del sistema.
Dueño del producto: financia el proyecto o conecta con quien lo financia, es el encargado de aprobar los entregables.

Usuario: son los encargados de transmitir los requerimientos al equipo de trabajo, validan los entregables.

Equipo de especialistas: conformado por el analista, desarrollador, diseñador, teaster quienes son encargados de la realización del sistema.

- ✓ **Analista:** es el encargado de las especificaciones del sistema y brindar seguridad al sistema.
- ✓ **Desarrollador:** es el encargado del código del sistema es quien se encarga de establecer las contramedidas.
- ✓ **Diseñador:** es el encargado del diseño de interfaces
- ✓ **Tester:** es el encargado de realizar las pruebas del sistema.

La misma que se puede observar en la Figura III.7

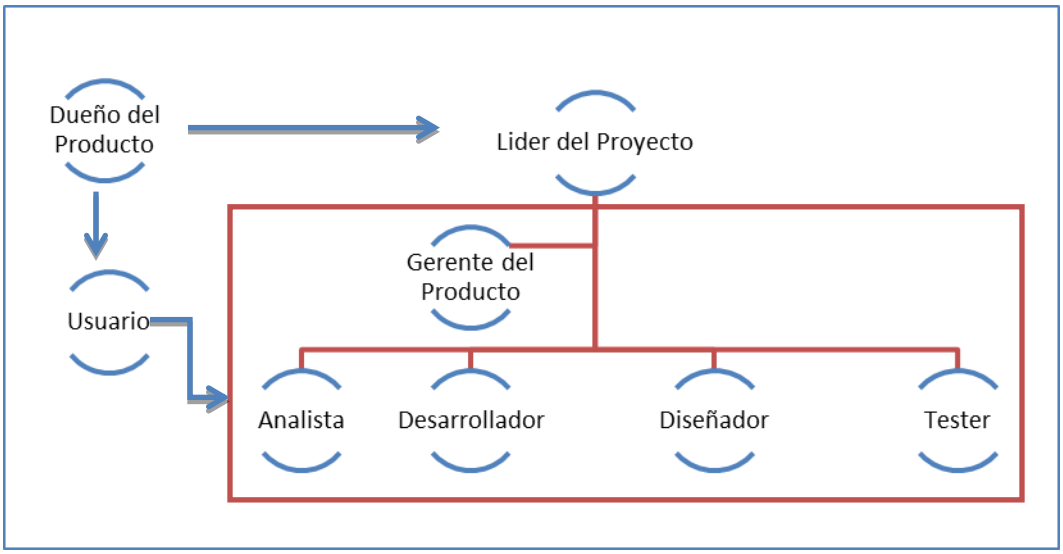


Figura III. 7.Equipo de Trabajo

La Plantilla Equipo de Trabajo que se observa en la Tabla III.VII ayuda a identificar como está conformado el equipo de trabajo, las actividades que realizan y el aporte que brindan para el desarrollo de la aplicación web

Tabla III. VII. Plantilla Equipo de Trabajo

Dueño del Producto:				
Sistema:				
Usuario:				
Líder del Proyecto:		**Gerente del Producto:**		
Equipo de Especialistas				
Analista	**Desarrollador**	**Diseñador**	**Tester**	**OTRAS**
Tarea	Tarea	Tarea	Tarea	Se lo necesita
Responsable	Responsable	Responsable	Responsable	Responsable
Observaciones:				
Resultados:				

.

- ✓ **Funcionalidad del Aplicación**

Se plantea el funcionamiento que tendrá la aplicación en base a los requerimientos, definiendo las entradas y salidas para lo cual se tomó en cuenta el principio básico de validación de las entradas y salidas de información y de igual manera los pilares de seguridad, como se observa en la Tabla III.VIII la plantilla de información la cual nos permite tener un Tratamiento de datos como se observa en la Figura III.8 y así tener una idea más clara sobre la seguridad que se debe implementar.

Tabla III. VIII.Plantilla de Información

Requerimiento #:		
TIPO DE DATO	**ENTRADA**	**SALIDAS**

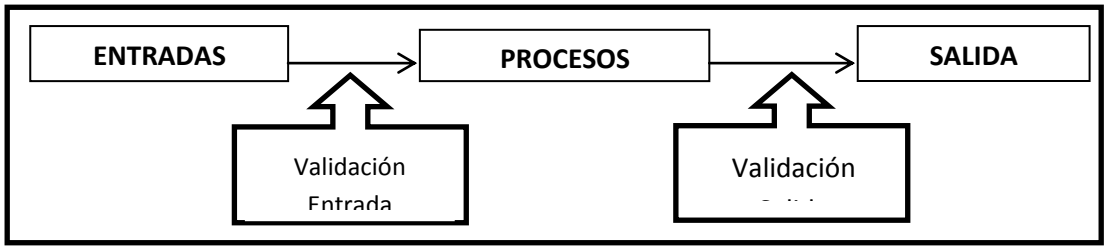


Figura III. 8. Tratamiento de Datos

3.3.2. Paso 2 Modelar

Genera un listado de amenazas, vulnerabilidades así como los posibles incidentes no deseados dependiendo de la funcionalidad de la aplicación y la arquitectura.

- ✓ **Diseño de la Arquitectura**

Una vez que se tiene en claro la funcionalidad de la aplicación se debe analizar los componentes y las tecnologías que están implicadas en el desarrollo de la aplicación web, esto se logra mediante un diagrama de vulnerabilidades en el diseño de la arquitectura, la cual que permite identificar cómo interactúan entre si cada uno de los componentes.

- ✓ **Planteamiento de las posibles vulnerabilidades**

Con la plantilla de información y el diseño de la arquitectura de la aplicación web permite plantear cuáles son las posibles vulnerabilidades que se determinaron que pueden ocurrir en el desarrollo de la aplicación.

- ✓ **Planteamiento de seguridad**

Una vez que se tiene planteado las posibles vulnerabilidades de la aplicación se puede establecer la seguridad que se necesita para mitigar este problema acogiendo los principios de seguridad y estableciendo una solución enfocado a la vulnerabilidad que se esté tratando.

3.3.3. Paso 3. Evaluar

Se identifican las amenazas posibles en la aplicación.

- ✓ **Detección y listado de amenazas reales**

Una vez que se haya finalizado el desarrollo de la aplicación web se analiza ésta para se obtiene un listado de las amenazas que existen en la aplicación.

3.3.4. Paso 4. Cuantificar

Se determina el impacto de los riesgos asociados con cada amenaza además de la probabilidad.

- ✓ **Determinar la severidad de Amenaza**

Para determinar la severidad se lo realiza con ayuda del método de puntuación DREAD, el cual permite determinar:

- ✓ **DAÑO POTENCIAL (D):** ¿Cuál es el daño que puede originar la vulnerabilidad si llega a ser explotada.?
- ✓ **REPRODUCIBILIDAD (R):** ¿Es fácil reproducir las condiciones que propicien el ataque?
- ✓ **EXPLOTABILIDAD (E):** ¿Es sencillo llevar a cabo el ataque?
- ✓ **USUARIOS AFECTADOS (A):** ¿Cuantos usuarios se verían afectados?
- ✓ **DESCUBRIMIENTO (D):** ¿Es fácil encontrar la vulnerabilidad?

La TablaIII.IX es la que se utilizará para determinar la severidad de los riegos según la amenaza encontrada.

Tabla III. IX.Severidad del Riesgo

Amenaza	D	R	E	A	D	Total	Puntuación

Este método permite puntuar el riesgo y de igual forma priorizar las acciones para mitigar el riesgo, el método trata de facilitar el uso de un criterio común; se toma como referencia la Tabla III.X con el método DREAD.

Tabla III. X.Método DREAD.

	PUNTUACIÓN	ALTO (3)	MEDIO(2)	BAJO(1)
D	Damage potential (Daño potencial)	El atacante podría ejecutar aplicaciones con permiso de administrador; subir contenido.	Divulgación de información sensible	Divulgación de información trivial
R	Reproducibility (Reproducibilidad)	El ataque es fácilmente reproducible.	El ataque se podría reproducir, pero sólo en condiciones muy concretas.	Ataque difícil de reproducir, incluso conociendo la naturaleza del fallo.
E	Exploitability (Explotabilidad)	Un programador novel podría implementar el ataque en poco tiempo.	Un programador Experimentado podría implementar el ataque.	Se requieren ciertas habilidades y conocimientos para explotar la vulnerabilidad.
A	Affected users (Usuarios afectados)	Todos los usuarios, configuración por defecto.	Algunos usuarios, no es la configuración por defecto.	Pocos usuarios afectados.
D	Discoverability (Descubrimiento)	Existe información pública que explica el ataque. Vulnerabilidad presente en una parte de la aplicación muy utilizada.	La vulnerabilidad afecta a una parte de la aplicación que casi no se utiliza. No es muy probable que sea descubierta.	El fallo no es trivial, no es muy probable que los usuarios puedan utilizarlo para causar un daño potencial.

Fuente: Análisis-y-Modelado-de-Amenazas Método DREAD

Para determinar el riesgo que una vulnerabilidad posee se debe sumar los parámetros de DREAD y con el valor total obtenido se da una puntuación como se puede observar en la Tabla III.XI

Tabla III. XI. Rango de Puntuación

RANGO	PUNTUACIÓN
5-7	BAJO
8-11	MEDIO
12-15	ALTO

El Total se determina mediante la suma de los valores que se obtienen en DREAD

(D)+(R)+(E)+(A)+(D)=TOTAL

De esta manera se establece el rango para determinar la puntuación.

3.3.5. PASO 5. Validar

Realiza las mejoras oportunas estableciendo el tratamiento para el riesgo.

- ✓ **Contramedidas**

Las contramedidas que se deben tomar están enfocadas en base al ataque que se haya detectado, las mismas que deben brindar una solución la cual ayude en su mayoría a mitigar las vulnerabilidades. Una vez que se hayan establecido las contramedidas en la aplicación web aplicamos la recursividad de los pasos por lo cual retornamos al Paso 3 Evaluar, de esta manera se podrá volverá a analizar la aplicación hasta que las amenazas se hayan mitigado a lo más mínimo posible.

CAPÍTULO IV

DESARROLLO DEL SISTEMA DE CONTROL DE NUEVOS ASPIRANTES EN LA EMPRESA "GRUPO LAAR" UTILIZANDO BUENAS PRÁCTICAS DE DESARROLLO WEB SEGURO GSAW.

4.1. Gestión del Proyecto

La Programación Extrema o XP nace oficialmente hace cinco años fundada por Kent Beck, es el más destacado de los procesos ágiles de desarrollo de software. La programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de la XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

Esta metodología de desarrollo de software posee cuatro características básicas que debe reunir el programador XP que son: la simplicidad, la comunicación y la retroalimentación o reutilización del código desarrollado (reciclado de código).

El ciclo de vida de XP comprende cuatro fases como se lo puede observar en la FiguraIV.9.

- ✓ Fase I: Planificación
- ✓ Fase II: Diseño
- ✓ Fase III: Codificación
- ✓ Fase IV: Pruebas

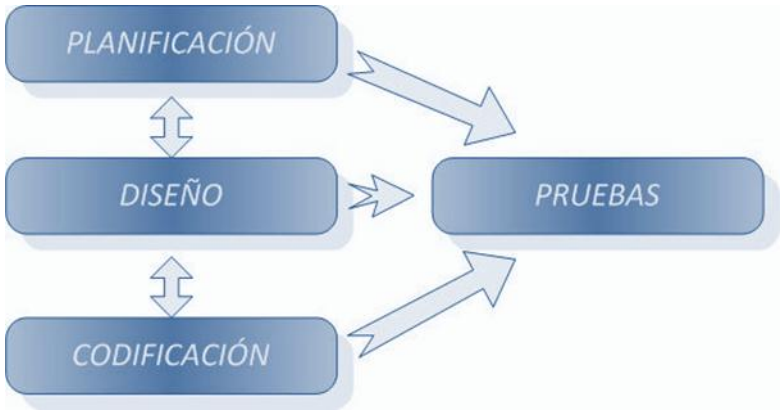


Figura IV. 9. Ciclo de Vida XP
Fuente: Fases de Ciclo de Vida XP

Antes de continuar desarrollando la metodología es necesario aclarar que el desarrollo de la aplicación web se hizo trabajando en conjunto con la con la Metodología XP y la Guía de buenas prácticas GSAW como se puede observar en la Tabla IV.XII.

Tabla IV. XII.Metodología XP y GSAW

FASES METODOLOGÍA XP	PASOS DE GSAW
1. Planificación	✓ Funcionalidad de la aplicación ✓ Identificación del equipo de trabajo
2. Diseño	✓ Diseño de la arquitectura ✓ Planteamiento de las posibles vulnerabilidades ✓ Planteamiento de seguridad
3. Codificación	✓ Contramedidas
4. Pruebas	✓ Detección y listado de las amenazas real ✓ Determinar la severidad del riesgo

4.2. Fase I. Planificación

En esta fase se desarrollarán todas las actividades que permitan conocer, comprender y entender el proceso de CONTROL DE NUEVOS ASPIRANTES EMPRESA "GRUPO LAAR, así como también las historias de usuario del mismo, la planificación inicial e iteraciones necesarias para automatizar este proceso y cumplir con los objetivos planteados.

4.2.1. Descripción de la Aplicación

El sistema de Control de Nuevos Aspirantes a la Empresa "GRUPO LAAR" es la elaboración de una evaluación en base a un cargo o puesto existente, que un aspirante previamente registrado en el sistema puede rendir, también permite un registro de las empresas, cargos y usuarios que existen en la empresa y de la misma manera se registra las pruebas, preguntas y respuestas que conforman la

evaluación, una vez que un aspirante haya rendido una evaluación ,esta es almacenada para determinar la calificación que obtuvo y así dar una respuesta sobre los resultados obtenidos.
El sistema se desarrolló en el lenguaje de programación PHP, se utilizó como gestor de base de datos MySQL, por su rapidez y bajo consumo de recursos.

4.2.2. Definición del flujo del proceso de Sistema.

Para automatizar un proceso es necesario conocer cómo es que se realiza es por esto la necesidad de definirlo y no dejar cabos sueltos en cuanto a requerimientos se refiere.

A continuación se presentan los flujos del proceso que tiene el sistema:
Proceso Administrador: Como se muestra en la Figura IV.10. el usuario administrador debe estar registrado en el sistema para acceder una vez que tenga autorización puede acceder a los todos los módulos como son:

- ✓ Usuario: es aquí donde registra a los usuarios de la empresa y de la misma manera puede modificarlos.
- ✓ Empresa: aquí puede registrar o modificar una empresa.
- ✓ Cargos: permite que pueda registrar o modificar un cargo.
- ✓ Aspirante: se lista a los aspirantes registrados en las empresas y una vez que haya rendido una evaluación se puede dar una respuesta sobre el resultado obtenido.
- ✓ Evaluación: permite el registro y modificación de las pruebas, preguntas y respuestas que conforman la evaluación.

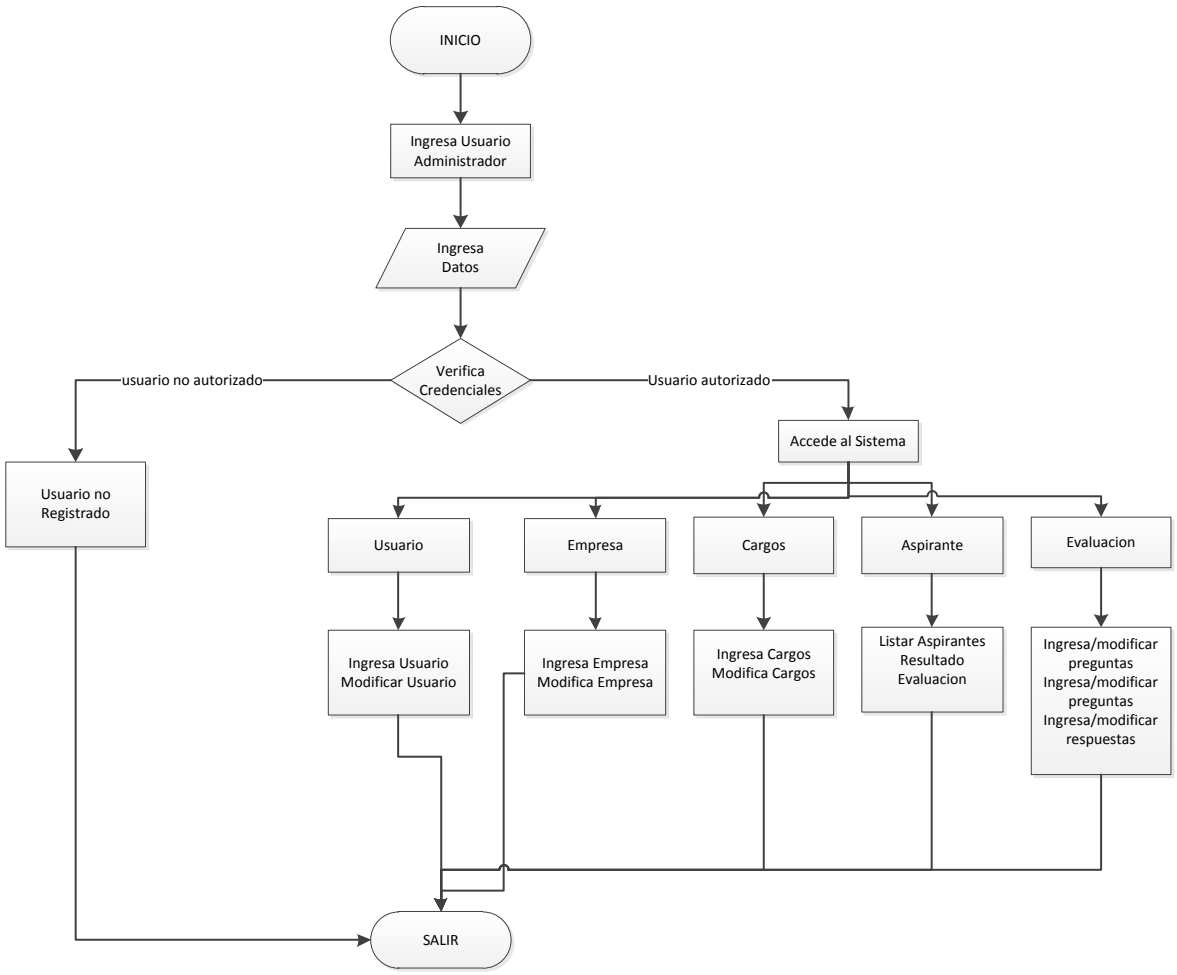


Figura IV. 10. Proceso Administrador

Proceso Usuario GRUPO LAAR: se puede observar en la Figura IV.11 el usuario registrado por el administrador en el sistema una vez que esté autorizado puede acceder a los siguientes módulos:

- ✓ Cargos: permite que pueda registrar o modificar un cargo.
- ✓ Aspirante: se lista los aspirantes registrados en las empresas y una vez que haya rendido una evaluación se puede dar una respuesta sobre el resultado obtenido.
- ✓ Evaluación: permite el registro y modificación de las pruebas, preguntas y respuestas que conforman la evaluación

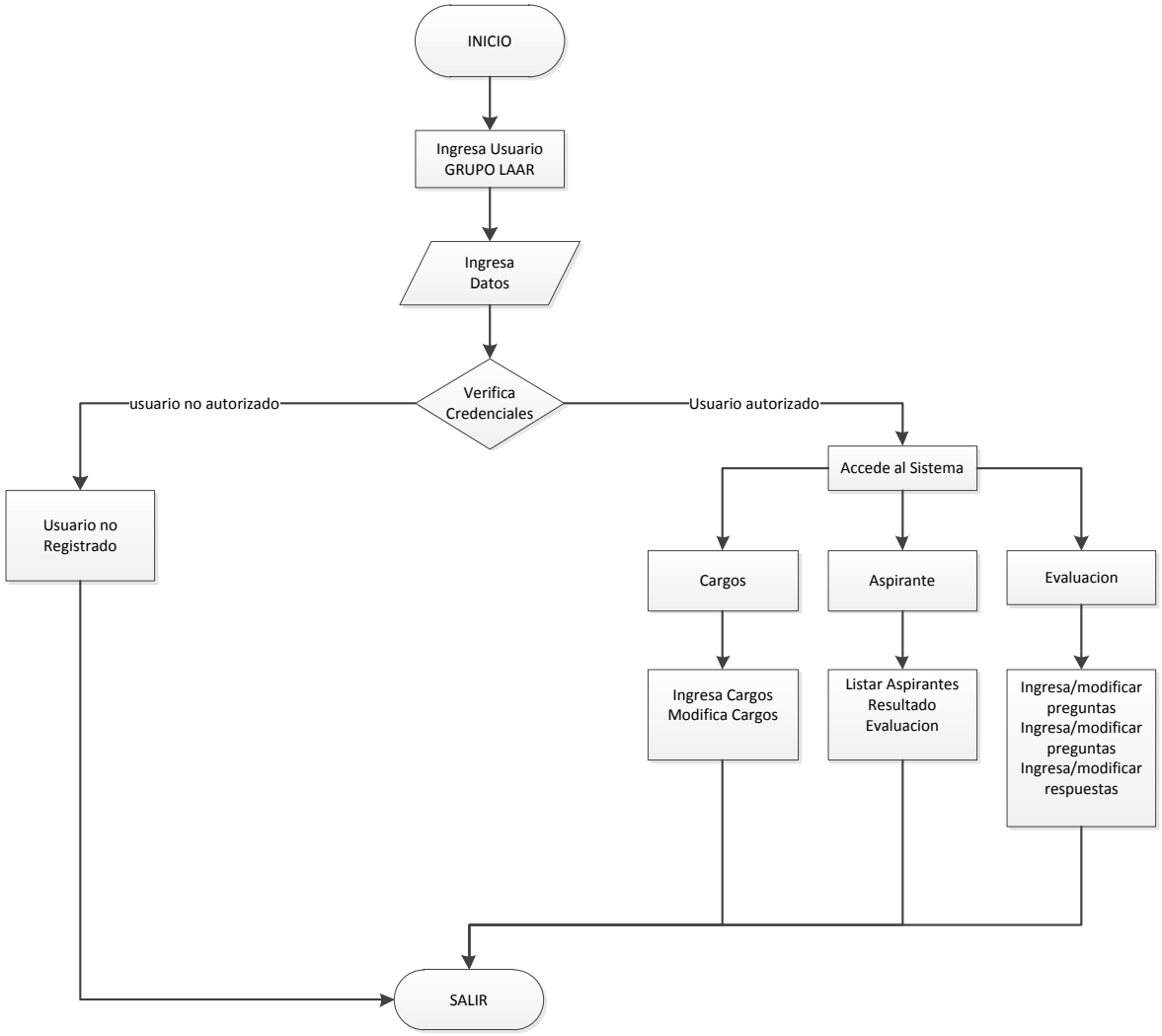


Figura IV. 11. Proceso Usuario GRUPO LAAR

Proceso Usuario Aspirante: como se observa en la Figura IV.12 una vez que el usuario se haya registrado en el sistema y esté autorizado para acceder al sistema puede acceder a los siguientes módulos:

- ✓ Evaluación: le permite al usuario rendir la o las prueba del cargo (puesto) que se oferte en la empresa.
- ✓ Respuesta: el usuario puede ver una respuesta que emite la empresa sobre resultado obtenido en la evaluación.

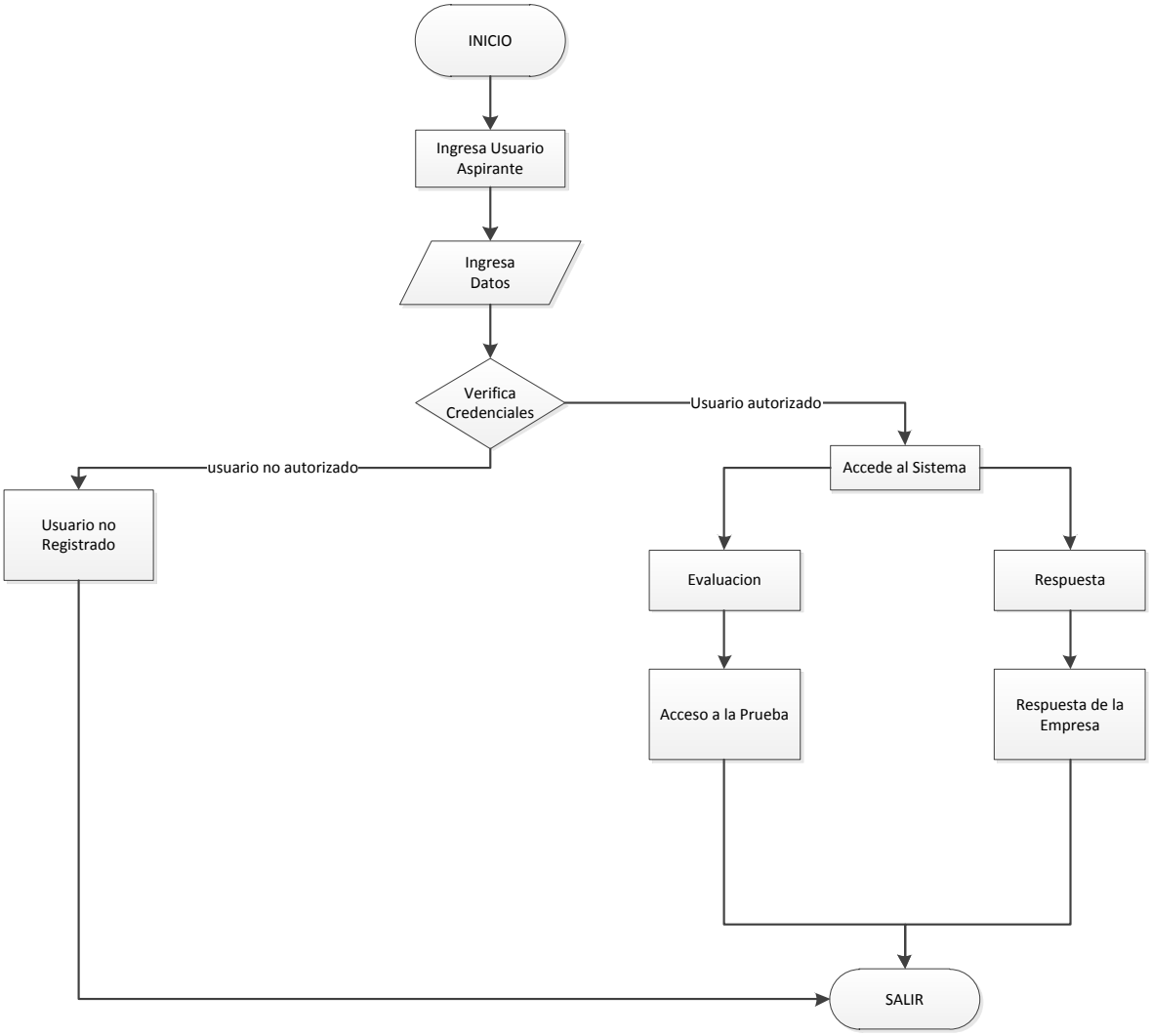


Figura IV. 12.Proceso Aspirante

4.2.3. Integrantes y Roles

Teniendo en cuenta la participación tanto de los Jefes de Proyecto, como de los usuarios y desarrolladores, se formará así el equipo encargado de la implementación de software. Esto implicará que los diseños deberán ser claros y sencillos, los usuarios deberán disponer de versiones operativas cuanto antes para poder participar en el proceso creativo mediante sus

sugerencias y aportaciones, el potenciar al máximo el trabajo en equipo es fundamental para el desarrollo del sistema, dicho equipo de trabajo se ve ilustrado en la Tabla definiendo Roles:

Tabla IV. XIII. Integrantes y Roles

Miembro	Grupo	Roles XP	Metodología
Elvira Yánez	Tesista	Rastreador, Testeador, Programador	Xp
Ing. Gloria Arcos Ing. Diego Almachi	Consultor	Entrenador	

- ✓ **Identificación del Equipo de Trabajo (GSAW)**

La identificación del equipo de trabajo se encuentra especificada en la TablaIV.XIV

Tabla IV. XIV.Plantilla Equipo de Trabajo

Dueño del Producto: Empresa "GRUPO LAAR"			
Sistema: Control de Nuevos Aspirantes a la Empresa "GRUPO LAAR"			
Usuarios: Área de Sistemas/Área de Recursos Humanos /Aspirantes			
Líder del Proyecto: Ing. Pablo Montufar			**Gerente del Producto:** Ing. Diego Almachi
Equipo de Especialistas			
Analista	**Desarrollador**	**Diseñador**	**Tester**
TAREAS			
Encargado de cumplir con los	Encargado de construir la aplicación en base a los	Encargado del diseño de	Comprobar el correcto

pasos (1-4) de la guía GSAW	requerimientos que se planten y se encarga de paso 5 de la guía GSAW	las interfaces de la aplicación	funcionamiento de la aplicación
RESPONSABLE: Elvira Yánez			
Observaciones: ✓ El análisis de requerimientos se lo realizo en conjunto con el Área de Recursos Humanos y el Área de Sistemas, ya que ellos son quienes utilizan la aplicación y conocen lo que debe tener la misma. ✓ El diseño de la aplicación se trabajó en conjunto con el área de márquetin de la empresa para respetar los colores y diseño de las pantallas.			
Resultados. ✓ Conformación de Equipo de Trabajo ✓ Se estableció la lista de requerimientos			

- ✓ **Funcionalidad del Aplicación (GSAW)**

La funcionalidad de la Aplicación se la realizó identificando las entradas y salidas de cada requerimiento como se puede observar en la Tabla IV.XV y se determinó la validación de entrada y salida que se debe emplear como se puede observar en la Figura IV.13

Tabla IV. XV.Plantilla de Información Requerimiento 1

Requerimiento 1: Gestionar usuarios (Ingreso, Modificación)		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres	Datos por parte de los Usuario	Envió a la base de datos

Números		

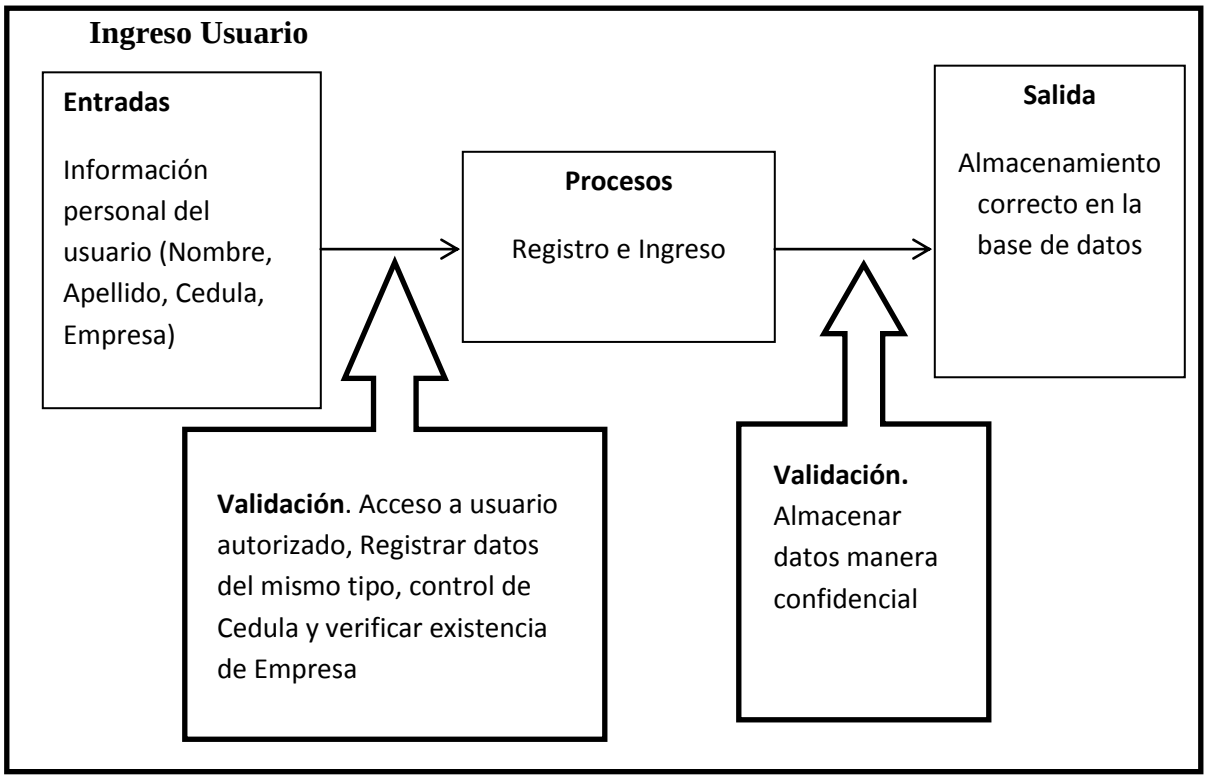


Figura IV. 13.Tratamiento de Datos Ingreso Usuario

4.2.4. Historias de Usuario

El Propósito de las historias de usuario es determinar las necesidades del sistema; por este motivo es que su descripción se la realiza de una manera corta y escrita en lenguaje del usuario esto quiere decir que no se utiliza terminología técnica, se proporciona los detalles sobre cuánto tiempo conllevará la implementación de dicha historia de usuario, la iteración que tendrá, se la considerará como tiempo de trabajo 2 horas por cada día laborable, se lo puede observar en la TablaIV.XVI.

Tabla III. XVI.Historia de Usuarios

N° ITERAC	N°	NOMBRE DE LA HISTORIA	FECHA INICIO	FECHA FIN	PUNTOS
1	1	El sistema permite un registro de Aspirantes	26/07/2013	06/08/2013	8
1	2	El sistema permite un registro de Usuarios GRUPO LAAR	07/08/2013	16/08/2013	8
1	3	El sistema permite autenticación de usuarios	19/08/2013	26/08/2013	6
2	4	El sistema permite el registro de empresas	27/08/2013	04/09/2013	7
2	5	El sistema permite el registro de cargos	05/09/2013	13/09/2013	7
2	6	El sistema permite el registro de pruebas	16/09/2013	26/09/2013	9
2	7	El sistema permite el registro de preguntas	27/09/2013	09/10/2013	9
2	8	El sistema permite el registro de respuestas	10/10/2013	22/10/2013	9
3	9	El sistema permitiré modificación del Usuario GRUPO LAAR	23/10/2013	30/10/2013	6
3	10	El sistema permite modificación de la empresa	31/10/2013	06/11/2013	6
3	11	El sistema permite modificación del cargo	13/11/2013	21/11/2013	6
3	12	El sistema permite modificación del pruebas	25/11/2013	02/11/2013	6
3	13	El sistema permite modificación del preguntas	03/12/2013	10/12/2013	6

3	14	El sistema permite modificación del respuestas	11/12/2013	18/12/2013	6
4	15	El sistema permite rendir la evaluación al aspirante	26/12/2013	08/01/2014	10
4	16	El sistema permite dar respuesta aspirantes	09/01/2014	16/01/2014	6
4	17	El sistema permite reporte los aspirantes	17/01/2014	24/01/2014	6
4	18	El sistema permite reporte de la Evaluación de un aspirante	27/01/2014	03/02/2014	6

4.2.5. Plan de Entregas

Se usará el plan de entregas para crear los planes de cada iteración.

El plan de entrega se puede planificar en función de estos dos parámetros: tiempo de desarrollo ideal y grado de importancia para el usuario. La planificación de las iteraciones individuales se lo efectúa justo antes de que comience cada iteración como se puede apreciar en las Tablas IV (XVII-XX) y en las Figuras IV (14-17)

Iteración 1

Tabla IV. XVII. Iteración 1 Historial Usuarios

HISTORIAL DE USUARIO	Puntos Estimados
El sistema permite un registro de Aspirantes	8
El sistema permite un registro de Usuarios GRUPO LAAR	8
El sistema permite autenticación de usuarios	6

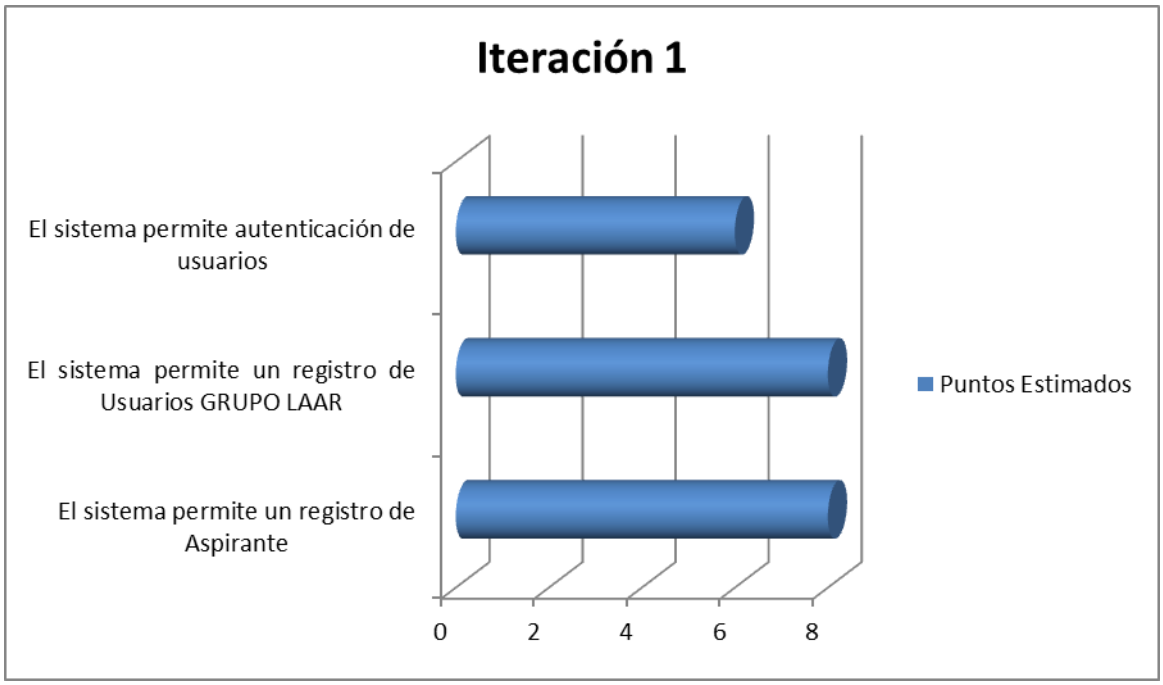


Figura IV. 14. Iteración 1 Historial de Usuarios

ITERACIÓN 2

Tabla IV. XVIII. Iteración 2 Historial Usuario

HISTORIAL DE USUARIO	PUNTOS ESTIMADOS
El sistema permite el registro de empresas	7
El sistema permite el registro de cargos	7
El sistema permite el registro de pruebas	9
El sistema permite el registro de preguntas	9
El sistema permite el registro de respuestas	9

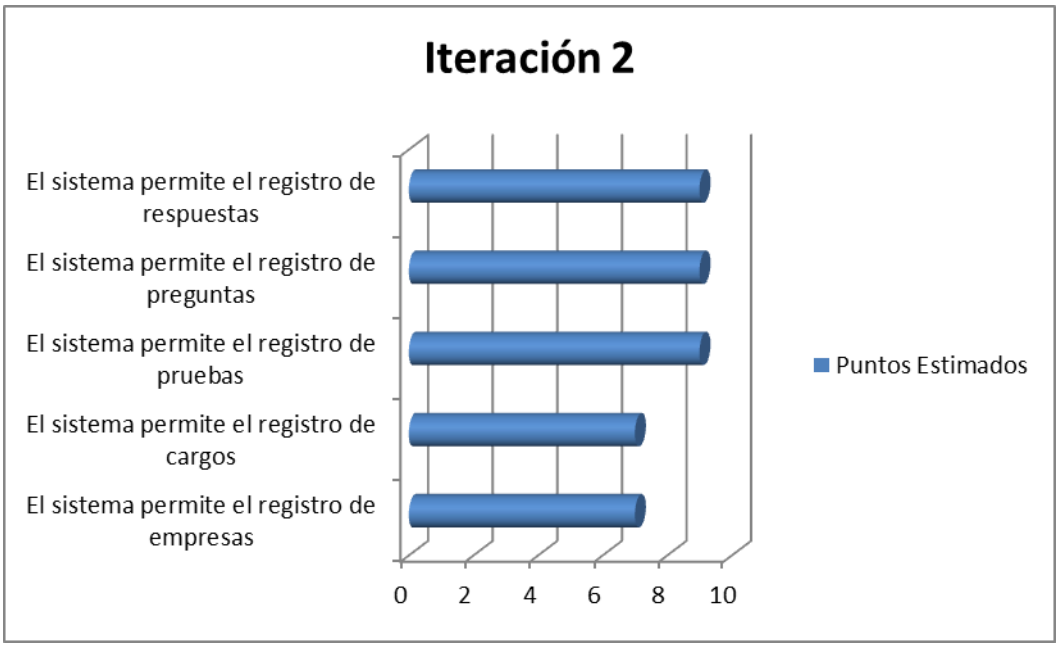


Figura IV. 15. Iteración 2 Historial Usuario

ITERACIÓN 3

Tabla IV. XIX. Iteración 3 Historial Usuario

HISTORIAL DE USUARIO	PUNTOS ESTIMADOS
El sistema permitiré modificación del Usuario GRUPO LAAR	6
El sistema permite modificación del Empresa	6
El sistema permite modificación del Cargo	6
El sistema permite modificación del Pruebas	6
El sistema permite modificación del Preguntas	6
El sistema permite modificación del Respuestas	

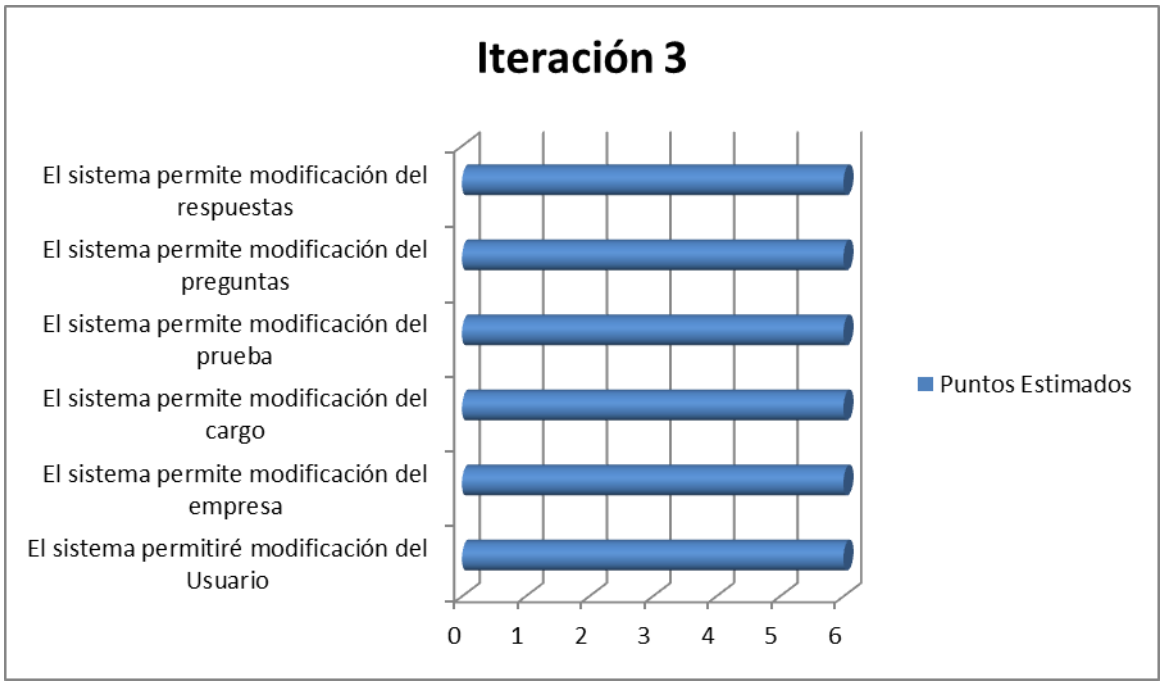


Figura IV. 16. Iteración 3 Historial Usuario

ITERACIÓN 4

Tabla IV. XX. Iteración 4 Historial Usuario

HISTORIAL DE USUARIO	PUNTOS ESTIMADOS
El sistema permite listar aspirantes	5
El sistema permite rendir la evaluación al aspirante	10
El sistema permite dar respuesta aspirantes	6
El sistema permite reporte los aspirantes	6
El sistema permite reporte de la Evaluación de un aspirante	6

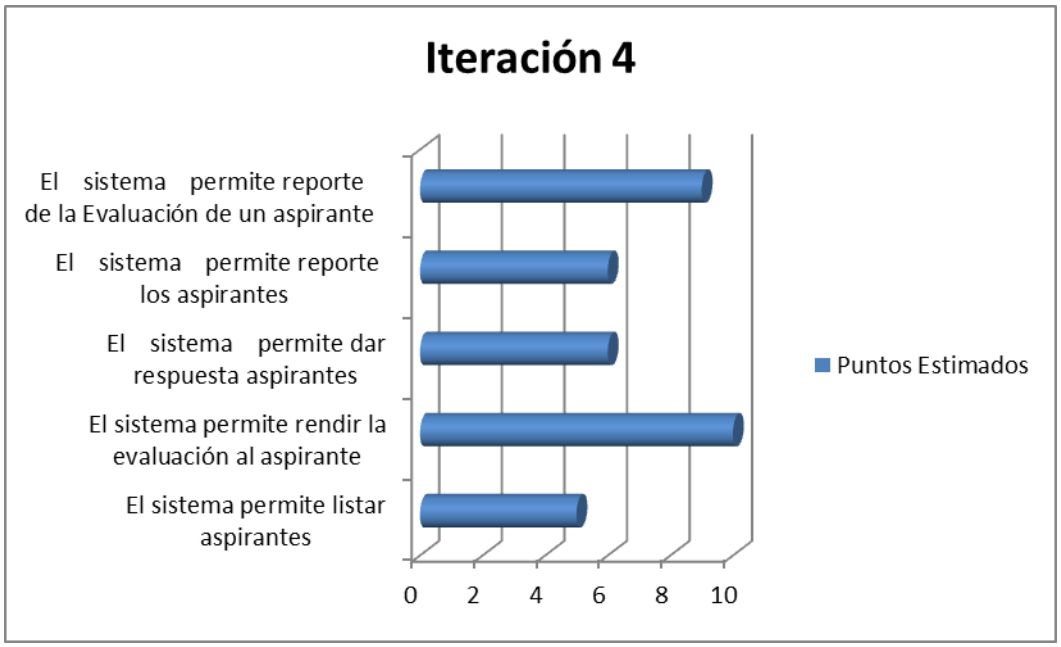


Figura IV. 17. Iteración 2 Historial Usuario

ITERACIÓN 1

La Tabla IV.XXI. Muestra la primera historia de usuario de la Iteración 1 que se implementó dentro de la aplicación, la que contiene sus respectivas tareas de ingenierías como se lo puede observar en las Tablas IV. (XXII-XXVI)

Tabla IV. XXI.Iteración 1. Historia 1.

Historia de Usuario	
Número: 1	Nombre de la historia: Registro aspirante
Creación de historia de usuario	
Usuario: Aspirante	Iteración Asignada: 1
Prioridad en el Negocio: Alta	Puntos Estimados:12
Riesgo en el Desarrollo: Bajo	Puntos Reales:16
Descripción: El sistema permitirá el registro de aspirantes.	
Observaciones: Se registrara la siguiente información de cada aspirante: empresa, cargo, cedula, nombre, apellido, dirección, teléfono, estado, mail, password	
Pruebas de Aceptación: ✓ Al ingresar un código (cedula) de un aspirante ya existente, se emitirá un mensaje de error. ✓ Al ingresar los datos erróneo se emitirá un mensaje de confirmación que los datos Inserción del Aspirante fallida ✓ Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.	

Tabla IV. XXII.Tarea de Ingeniería 1.

Tarea de Ingeniería		
Historia de Usuario: Registro aspirante		
Número de Tarea: 1	Nombre de Tarea: Creación de los métodos necesarios en clase **ADatos** de la capa de **Acceso_Datos** y la creación de la clase aspirante	
Tipo de Tarea: Desarrollo.		Puntos Estimados: 6
Fecha Inicio:14/10/2013		Fecha Fin:21/10/2013
Programador Responsable: Elvira Yánez		
Descripción: Crearemos el método web **insertarregistroaspirante** en la clase **ADatos** en la capa de **Acceso_Datos**		

Pruebas de Aceptación. ✓ Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos. ✓ Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.

Tabla IV. XXIII.Tarea de Ingeniería 2.

Tarea de Ingeniería		
Historia de Usuario: Registro aspirante		
Número de Tarea: 2	Nombre de Tarea: Creación del método **insertarregistroaspirante** en clase **LNegocios** de la capa de **Logica_Negocio.**	
Tipo de Tarea: Desarrollo.		Puntos Estimados: 5
Fecha Inicio: 22/10/2013		Fecha Fin: 28/10/2013
Programador Responsable: Elvira Yánez		
Descripción: Crearemos el método web **insertarregistroaspirante** en la clase **LNegocios** en la capa de **Logica_Negocio.**		
Pruebas de Aceptación. ✓ Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos. ✓ Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.		

Tabla IV. XXIV.Tarea de Ingeniería 3.

<table>
<tr><td colspan="3">Tarea de Ingeniería</td></tr>
<tr><td colspan="3">Historia de Usuario: Registro de aspirante</td></tr>
<tr><td>Número de Tarea: 4</td><td colspan="2">Nombre de Tarea: Creación de las páginas web en la capa Presentación necesarias para ingresar un aspirante</td></tr>
<tr><td colspan="2">Tipo de Tarea: Desarrollo.</td><td>Puntos Estimados: 5</td></tr>
<tr><td colspan="2">Fecha Inicio:29/10/2013</td><td>Fecha Fin:04/11/2013</td></tr>
<tr><td colspan="3">Programador Responsable: Elvira Yánez</td></tr>
<tr><td colspan="3">Descripción: Creación de las paginas RegistroAspirante.php en la capa Presentación, las mismas que permiten capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.</td></tr>
<tr><td colspan="3">Pruebas de Aceptación.
✓ Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos.
✓ Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.</td></tr>
</table>

En el Anexo 3 se describe las demás, iteraciones y cada una de las historias de usuario y las tareas de ingeniería.

4.3. Fase II. Diseño

Se diseña la base de Datos del sistema y las interfaces de usuarios finales.

4.3.1. Base de Datos

A continuación en la Figura IV.18 se visualiza el diseño de la base de datos para el desarrollo del sistema de Control de Nuevos Aspirantes a la Empresa “GRUPO LAAR”.

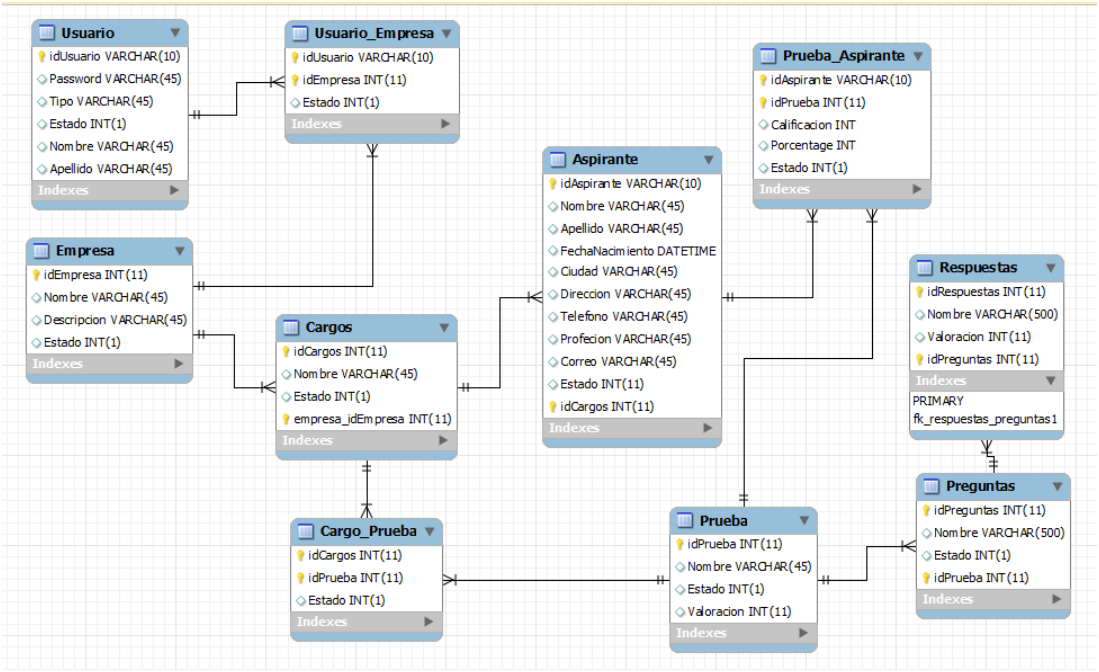


Figura IV. 18. Diseño de la Base de Datos

✓ Diseño de la Arquitectura (GSAW)

La arquitectura que se utilizó para el desarrollo de la aplicación web es Cliente Servidor .
De igual manera podemos identificar los componentes que se utilizaron y las tecnologías que nos ayudaron al desarrollo de la aplicación. Como se puede observar en la Figura IV.19 Se empleó un Servidor web Win2008 server con IIS (Internet Información Server), como motor de Base de datos se empleó es MySQL y navegador web (Firefox, Chrome).
Se puede observar de igual manera que el Servidor de Bases de Datos será quien permita el acceso a la Aplicación Web y realizar la autentificación y autorización de usuarios y de ésta manera se pueda realizar la validación de datos. Es importante resaltar que la validación de datos se efectúa en la capa de negocio.

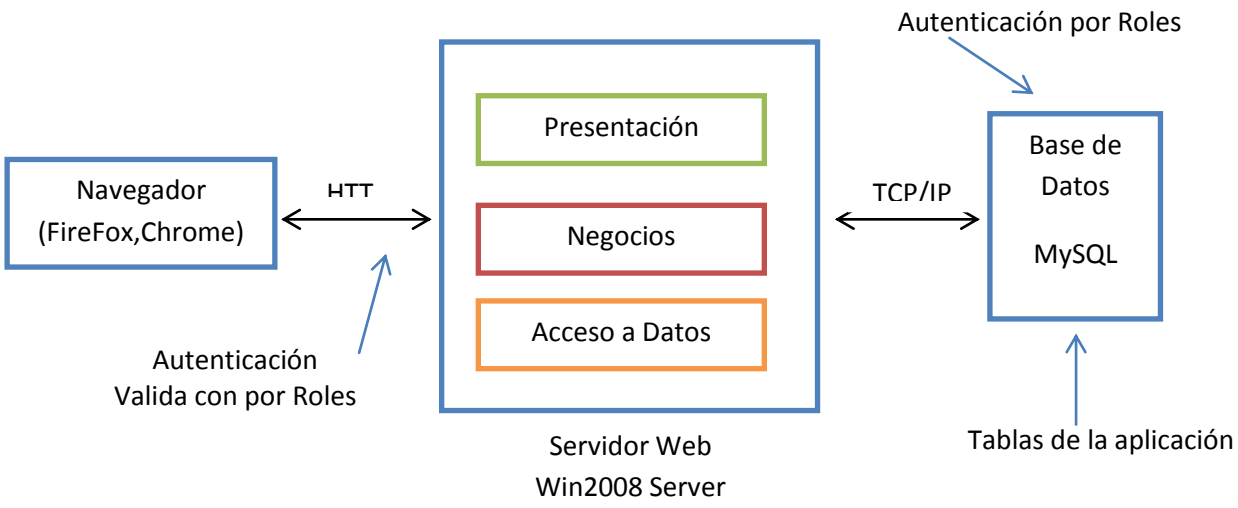


Figura IV. 19.Diagrama de Arquitectura y Componentes

En la Figura IV.20 Se puede observar que se han identificado algunas vulnerabilidades que la aplicación podrá sufrir, ya que la misma aún no cuenta con la seguridad necesaria entre los ataques identificados tenemos: robo de sesiones, intercepción de autenticación y datos, inyección de código, ataques de fuerza bruta por acceso no autorizado.

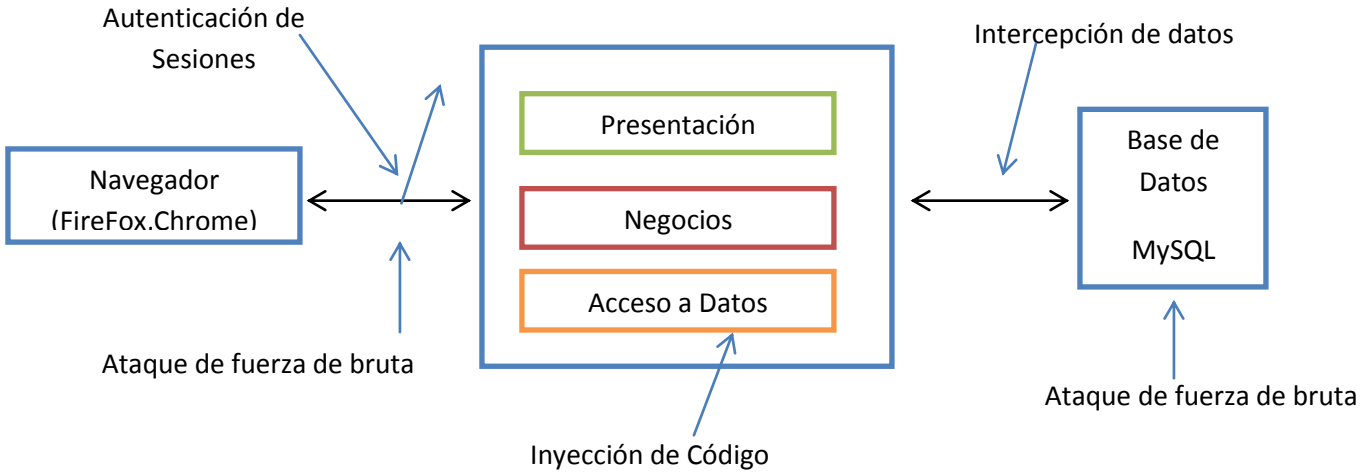


Figura IV. 20.Diagrama de Arquitectura y Componentes

4.3.2. Diseño de Interfaces

Con la descripción detallada de las historias de usuario y con los diagramas de procesos podemos definir las interfaces de usuario finales.

En la Figura IV.21 se puede observar la interfaz de usuario de Autenticación de Usuarios la cual nos permite acceder al Sistema del Control de Nuevos Usuarios de la Empresa Grupo Laar, los colores escogidos para todas las pantallas son los establecidos por el Área de Marketing de la empresa porque son los colores representativos de la misma de igual forma el tipo de letra para las pantallas y su tamaño es de acuerdo a su ubicación.

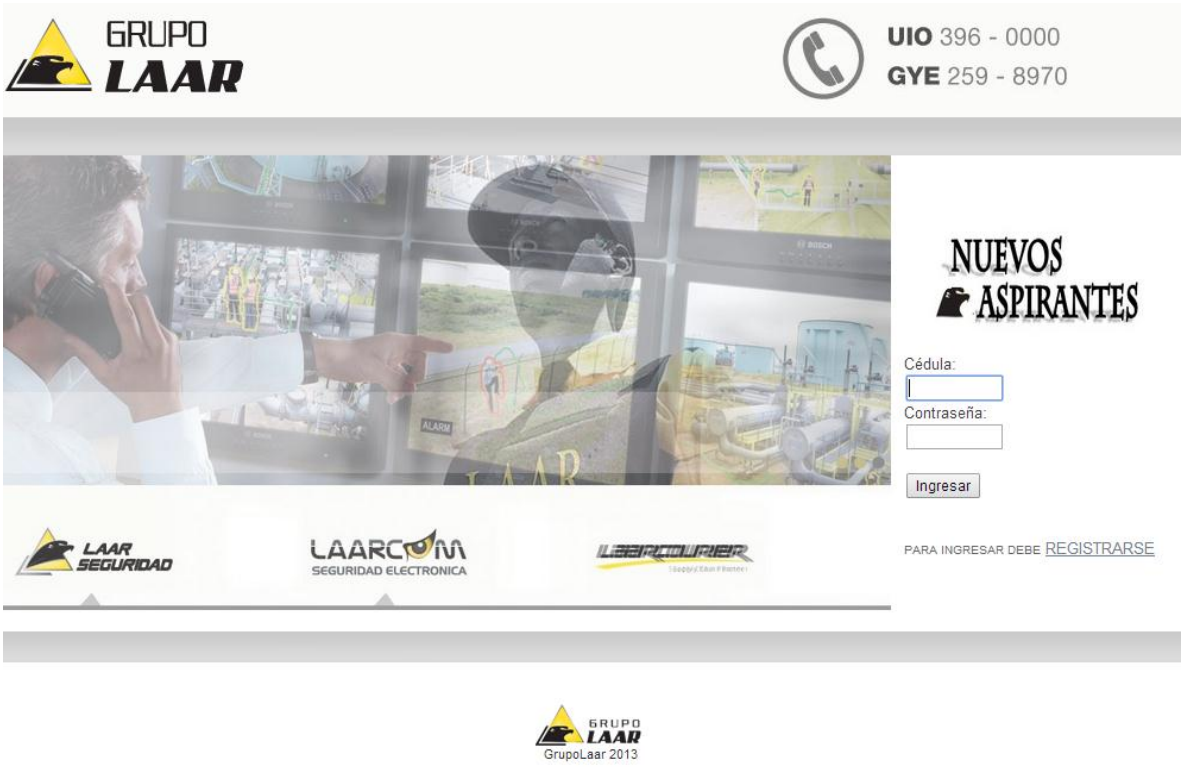


Figura IV. 21. Autenticación de Usuario

En el Anexo 4. Se puede visualizar el resto de las interfaces que conforman el sistema.

- ✓ **Planteamiento de las posibles vulnerabilidades (GSAW)**

El resultado obtenido en los pasos anteriores arroja como resultado las posibles vulnerabilidades.

En la Tabla IV.XXV podemos observar la Vulnerabilidad 1 Autenticación de Sesiones

Tabla IV. XXV.Vulnerabilidad 1

Vulnerabilidad:	Autenticación Sesiones
Atacante:	Considerar atacantes anónimos externos, además de usuarios con sus propias cuentas, que podrían intentar robar cuentas de otros. Considerar también a trabajadores que quieran enmascarar sus acciones.
Razón:	Utiliza filtraciones o vulnerabilidades en las funciones de autenticación o gestión de las sesiones Este tipo de vulnerabilidad podría permitir que algunas o todas las cuentas sean atacadas.

En la Tabla IV.XXVI podemos observar la Vulnerabilidad 2 Inyección de Código

Tabla IV. XXVI. Vulnerabilidad 2

Vulnerabilidad:	Inyección de Código
Atacante:	Considerar cualquier persona que pueda enviar datos no confiables al sistema, incluyendo usuarios externos, internos
Razón:	Ocurren cuando una aplicación envía datos no confiables a un intérprete la cual puede resultar en pérdida o corrupción de datos.

En la Tabla IV.XXVII podemos observar la Vulnerabilidad 3 Ataque de fuerza bruta

Tabla IV. XXVII.Vulnerabilidad 1

Vulnerabilidad:	Ataque de fuerza bruta
Atacante:	Se puede considerar a cualquier persona que desee obtener información usuarios externos, internos
Razón:	Encontrando claves, copias de datos no cifradas o accediendo por canales que automáticamente descifran la información. Esta vulnerabilidad normalmente compromete todos los datos que deberían haber estado cifrados

En la Tabla IV.XXVIII podemos observar la Vulnerabilidad 3 Intercepción de Datos

Tabla IV. XXVIII.Vulnerabilidad 4

Vulnerabilidad:	Intercepción de Datos
Atacante:	Considerar atacantes anónimos externos, además de usuarios con sus propias cuentas,
Razón:	Sucede al realizar un envío de las credenciales a través de la red, desde el Navegador de Internet al Servidor de Aplicaciones y desde el Servidor de Aplicaciones hasta el servidor de Bases de Datos, para un atacante sería muy sencillo interceptar las credenciales si estas viajan en texto en claro por la red.

- ✓ **Planteamiento de seguridad (GSAW)**

Una vez que se determinaron las posibles vulnerabilidades es necesario plantear medidas de seguridad necesarias para mitigar las vulnerabilidades detectadas.

- ✓ La validación de E/S se utilizó para eliminar caracteres sospechosos y así se evitó que usuarios malintencionados alteren la información que ingresa o sale de la aplicación.

- ✓ Se estableció roles con lo cual los usuarios tendrán acceso y privilegios mínimos con el objeto de proteger la aplicación.

- ✓ La autorización se comprobó en acceso al sistema con la cual los usuarios pueden modificar y acceder a la información.

- ✓ Se utilizó un algoritmo criptográfico de generación de credenciales.

- ✓ Se implementó una función de cierre de sesión para la aplicación

- ✓ La aplicación se conecta a la base de datos con credenciales diferentes y así tener una distinción de confianza (por ejemplo, usuario aspirante, usuario grupolaar, usuario administradores)

4.4. Fase III. Codificación

Es necesario tener una idea clara de la estructura y cómo van a interactuar el servidor web con los clientes y de igual manera con la base de datos, por lo que la Figura IV.22 se muestra el diagrama que se ha diseñado para de esta manera se tenga una mejor solución.

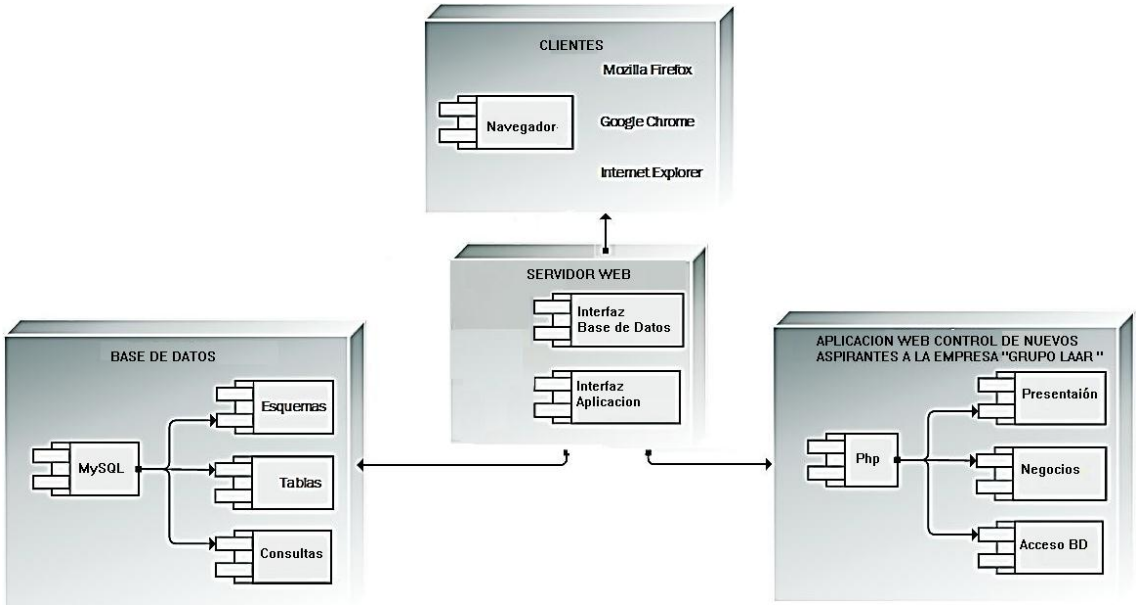


Figura IV. 22. Diagrama de Bloques y Despliegue del Sistema

4.4.1. Lenguaje

El sistema de Control de Nuevos Aspirantes a la Empresa "GRUPO LAAR" se desarrolla en lenguaje PHP, pues provee de grandes ventajas para cumplir con los objetivos planteados, como motor de base de datos se utilizó MySQL se utiliza una arquitectura en tres capas la cual permite independencia y una mejor organización del sistema como se puede observar en la Figura IV.23.

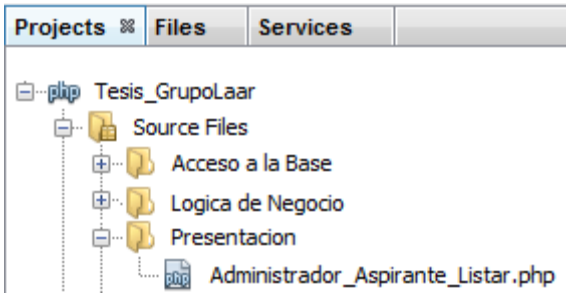


Figura IV. 23.Arquitectura del Sistema

- ✓ **Contramedidas**

Las contramedidas que se realizaron se las implementó en la codificación de la aplicación web son soluciones que se emplean con el lenguaje de programación PHP.

FALLA INYECCIÓN SQL

Para evitar fallas de Inyección de código SQL se utiliza:

- ✓ La función mySQL_real_escape_string, la cual se añade en las variables que introducimos en la consulta.

  mySQL_real_escape_string() llama a la función mySQL_real_escape_string de la biblioteca de MySQL, la cual antepone barras invertidas a los siguientes caracteres: *\x00*, *\n*, *\r*, **, *'*, *"* y *\x1a*;Esta función siempre debe usarse (con pocas excepciones) para hacer seguros los datos antes de enviar una consulta a MySQL.

  Sintaxis:

```
string mysql_real_escape_string ( string $unescaped_string [, resource $link_identifier =
NULL ] )
```

  Parámetros:

  - ✓ **unescaped_string**

    El string que va a ser escapado.

  - ✓ **link_identifier**

    La conexión MySQL. Si el identificador de enlace no se especifica, el último enlace abierto por mySQL_connect() es asumido. Si no se encuentra dicho enlace, la función intentará establecer un nuevo enlace como si mySQL_connect() fuese invocado sin parámetros.

Ejemplo:

A continuación se muestra la función utilizada en la consulta al encriptar la contraseña

```
//Encriptacion Contraseña
    $salt = substr($clave, 0, 2);
    $clave_crypt = crypt($clave, $salt);
    echo "CLAVEEE: $usuario";
    echo "CLAVEEE: $clave";
    echo "CLAVEEE: $clave_crypt";
    $instruccion = "select * from usuario where Usuario = '" . mySQL_real_escape_string($usuario) . "'" . " and Password = '$clave_crypt'";
    $consulta = mySQL_query($instruccion, $conexion) or die("Ingrese Correctamente");
    $datos = mySQL_fetch_array($consulta);
    $nfilas = mySQL_num_rows($consulta);
```

- ✓ Se usó la función htmlspecialchars()

Htmlspecialchars(): Convierte caracteres especiales en entidades HTML

Sintaxis:

```
string htmlspecialchars ( string $string [, int $flags = ENT_COMPAT | ENT_HTML401 [, string $encoding = "UTF-8" [, bool $double_encode = true ]]] )
```

Parámetros:

- ✓ **string**

  El string a convertir.

- ✓ **flags**

  Una máscara de bits de una o más de los siguientes indicadores, los cuales especifican cómo manejar las comillas, las secuencias de unidad de código inválidas y el tipo de documento utilizado.

- ✓ **double_encode**

Cuando double_encode se desactiva, PHP no codificará las entidades HTML existentes. El valor predeterminado es convertirlo todo.

Ejemplo:

Se utilizó en el inicio de sesiones

```
// Iniciar sesión
session_start();
/* @var $usuario type */
if ((isset($_POST['Ingreso']))) {
    $usuario = htmlspecialchars($_POST['cedula'] , ENT_QUOTES);
    /* @var $clave type */
    $clave = htmlspecialchars($_POST['clave'] , ENT_QUOTES);
```

✓ Validar todos los datos que se ingresan a la base de datos

Una vez que se ingrese los datos en un formulario se valida su correcto ingreso antes de almacenar, verificando que no estén vacíos

```
if (isset($_POST['accion'])) {
        if ((($_POST['empresa']) == "") && (isset($_POST['estadoempresa'])) && (($_POST['descripcion']) == "")) {
            print "<script>alert('LLENE TODOS LOS CAMPOS SON OBLIGATORIOS')</script>";
```

✓ Se mantuvo register_globals=off

```
// session_register ("usuario_valido");
// Con register_globals Off
        $_SESSION['usuario_valido'] = $usuario_valido;
        $_SESSION['Tipo'] = $datos['Tipo'];
```

```
$_SESSION['Cedula'] = $datos['Usuario'];
$_SESSION['Nombre'] = $datos['Nombre'];
$_SESSION['Apellido'] = $datos['Apellido'];
$_SESSION['Empresa'] = $datos2['idEmpresa'];
$em = $datos2['idEmpresa'];
echo strip_tags('<input type="hidden" name="empresa" id="empresa" value=' . "$em" . ' >');
```

FALLA COMANDOS

Para evitar fallas de Comandos se utiliza:

- ✓ Se previene la inclusión de código por medio de la función de strip_tags()

  strip_tags(): esta función intenta devolver un string con todos los bytes NULL y las etiquetas HTML y PHP retirados de un str dado

  Sintaxis:

  ```
  string strip_tags ( string $str [, string $allowable_tags ] )
  ```

  Parámetros:

  - ✓ **str**

    El string de entrada.
  - ✓ **allowable_tags**

    Se puede usar el segundo parámetro opcional para especificar cuáles etiquetas no deben ser retiradas.

  Ejemplos:

  Se lo utilizo en los campos ocultos

  ```
  echo strip_tags('<input type="hidden" name="user" id="user" value=' . "$usuario" . ' >');
  ```

- ✓ Limpiar los métodos get y post

Se lo utiliza al inicio de nuestros archivos .php que se van a invocar y lo que hace es recorrer todos los valores recibidos por GET y POST y limpiarlos de cadenas peligrosas

```
$input_arr = array();
foreach ($_POST as $key => $input_arr)
{
        $_POST[$key] = addslashes(limpiarCadena($input_arr));
}

$input_arr = array();
foreach ($_GET as $key => $input_arr)
{
        $_GET[$key] = addslashes(limpiarCadena($input_arr));
}
```

- ✓ Validar todos los datos para incluir y ejecutar archivos

- ✓ Puede ayudar si el php.ini de Allow_url_fopen=off

VALIDACIÓN DE SESIONES

- ✓ Todas las sesiones deben tener un tiempo de expiración por inactividad y absoluto para esto se establecer un timeout de sesión. Se puede utilizar un código similar al siguiente para controlar este timeout:

```
if (!isset($_SESSION['timeout_idle']))
```

- ✓ Sistema de logout: dar a los usuarios una forma de salir de su cuenta y destruir la sesión.

```
<?PHP
  if (isset($_SESSION["usuario_valido"])) {
```

```
    session_destroy ();
    echo '<BR>';
    print "<script>alert('SESION FINALIZADA')</script>";
    print("<script>window.location.replace('PaginaPrincipal.php');</script>");
  } else {
    print "<script>alert('NO EXISTE UNA CONEXION')</script>";
    print("<script>window.location.replace('PaginaPrincipal.php');</script>");
  }
  ?>
```

AUTENTICACIÓN DE SESIONES

- ✓ Siempre crear una nueva sesión id al recibir los datos de autenticación.

```
<?php
session_start();
?>
```

- ✓ Realizar un correcto filtrado de variables

```
$fono = filter_var($_POST["fono"]);
$profesion = strtoupper($_POST['profesion']);
$ciudad = strtoupper($_POST['ciudad']);
 $nombre1 = strtoupper($nombre);
$cedula = validarCI($_POST['cedula']);
```

- ✓ Establecer sistema de criptografía para las contraseñas

```
//Encriptacion Contraseña
      $salt = substr($clave, 0, 2);
      $clave_crypt = crypt($clave, $salt);
      echo "CLAVEEE: $usuario";
      echo "CLAVEEE: $clave";
```

```
echo "CLAVEEE: $clave_crypt";
```

- ✓ Solo Usuarios Autenticados y debidamente autorizados pueden acceder a la aplicación

```
<?php
        if (isset($_SESSION['usuario_valido']) && ($_SESSION['Tipo'] ==
"Administrador")) {
            require_once( "Usuario_Administrador.php" );
           echo '<BR>';
        } else
        if (isset($_SESSION['usuario_valido']) && ($_SESSION['Tipo'] ==
"UsuarioLaar")) {
           require_once ( "Usuario_GrupoLaar.php" );
           echo '<BR>';
        } else
        if (isset($_SESSION['usuario_valido']) && ($_SESSION['Tipo'] ==
"User")) {
           require( "Usuario_Aspirante.php" );
           echo '<BR>';
        }// Intento de entrada fallido
        else
        if (isset($usuario) && isset($clave)) {
           print "<script>alert('Cédula o Contraseña Incorrecta')</script>";

print("<script>window.location.replace('PaginaPrincipal.php');</script>");
        }
```

4.5. Fase IV. Pruebas

En ésta fase se verifica el correcto funcionamiento del sistema, ingresando los datos necesarios para su utilización.

A continuación se muestran las pruebas de aceptación realizadas de la primera historia de usuarios y tareas de ingeniería como se puede observar en las Tablas IV. (29-31), de igual manera se observan los CRC en las Tablas IV. (XXXII-XXXIV).

Tabla IV. XXIX.Prueba de Aceptación 1

Prueba de Aceptación:	
Código:1	**Historia de Usuario:** Registro aspirante
Nombre: Al ingresar un código (cédula) de un aspirante ya existente, se emite un mensaje de error.	
Responsable: Elvira Yánez	**Fecha:** 04/11/2013
Descripción: Al ingresar una cédula que ya existe, se mostrará un mensaje donde dirá La Cédula ya existe.	
Condiciones de Ejecución: Insertar un aspirante con una CEDULA inexistente	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón guardar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje	
Evaluación de la prueba: satisfactorio	

Tabla IV. XXX.Prueba de Aceptación 2

Prueba de Aceptación:	
Código:2	**Historia de Usuario:** Registro aspirante
Nombre: Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	
Responsable: Elvira Yánez	**Fecha:** 04/11/2013
Descripción: Verificar el Despliegue del mensaje “Inserción Exitosa!” del método insertarregistroaspirante implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un aspirante	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón enviar. 3. Se visualiza el mensaje “Inserción Exitosa!”.	
Resultado esperado: Emitir el mensaje “Inserción Exitosa!”.	
Evaluación de la prueba: Satisfactorio	

Tabla III. XXXI.Prueba de Aceptación 3

Prueba de Aceptación:	
Código:3	**Historia de Usuario:** Registro aspirante
Nombre: Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos.	
Responsable: Elvira Yánez	**Fecha:** 04/11/2013
Descripción: Verificar el Despliegue del mensaje "Falla al Insertar Aspirante" del método insertarregistroaspirante implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un aspirante , con error en el campo Cédula	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón enviar. 3. Se visualiza el mensaje "Falla al Insertar Aspirante" en caso que la Cédula este mal o el aspirante ya exista	
Resultado esperado: Emitir el mensaje de "Falla al Insertar Aspirante	

Evaluación de la prueba: satisfactorio

Tabla IV. XXXII. Tarjeta CRC Acceso Datos

Nombre de la Clase: Acceso Datos	
AccesoDatos() Conectar() Desconectar() EjecutarSelect() EjecutarUpdate()	**Colaboradores:**

Tabla IV. XXXIII. Tarjeta CRC ADatos

Nombre de la Clase: ADatos	
Responsabilidades: empresa. cargo. cedula, nombre, apellido, dirección, teléfono, mail, estado, password, insertarregistroaspirante ()	**Colaboradores:** AccesoDatos

Tabla IV. XXXIV. Tarjeta CRC Aspirante

Aspirante	
Responsabilidad	**Colaboración**
empresa, cargo, cedula,nombre, apellido, dirección, teléfono, mail, estado, password, Get(), Set()	

En el Anexo 3 se puede observar el resto de las pruebas de aceptación y CRC realizadas por cada historia de usuario y tarea de ingeniería.

✓ **Detección y Listado de Amenazas Reales (GSAW)**

Para detectar cuáles son las amenazas que existen en la aplicación se emplea la herramienta de verificación OWASP ZAP de la siguiente manera:
Iniciaremos el ataque a la aplicación escribiendo el URL atacar como se Observa en la Figura IV.24

Url: Nombre de la página atacar

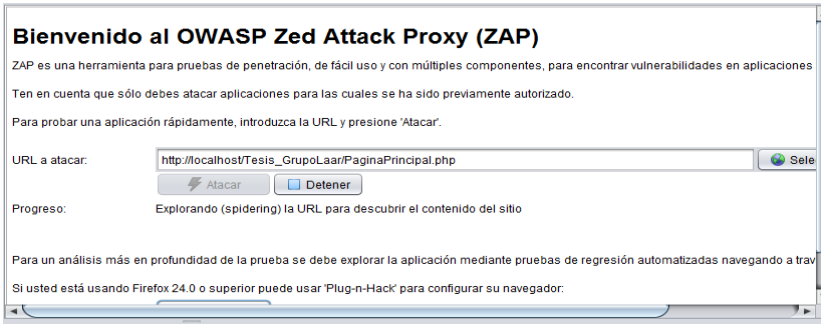


Figura IV. 24.Ataque a la aplicación

Se inicia el Escaneo de Activos hasta completar el 100% de escaneado como se observa en la Figura IV.25 es aquí donde el Spider de la herramienta va detectando uno a uno las posibles amenazas.

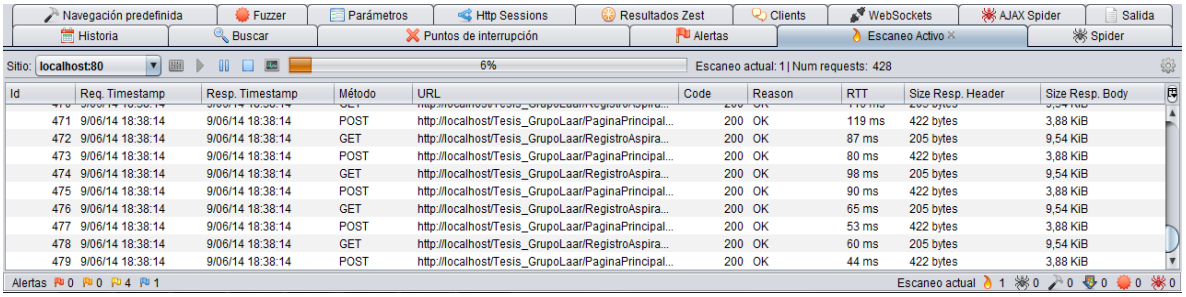


Figura IV. 25.Escaneo Activo

Una vez finalizado el Escaneo se emite un resultado sobre las alertas encontradas como se observa en la Figura IV.26 la herramienta arroja como resultado que se han encontrado 8 alertas de las cuales Falla de Comandos y Falla de inyección SQL son de

alto riesgos y Autenticación de Sesiones y Validación de Sesiones posee un riego mediano, de igual manera nos muestra que existe una Manipulación de parámetros, Dominio Cruzado JavaScript y por último un aviso de que las opciones de X-frame no se establecen en la cabecera.

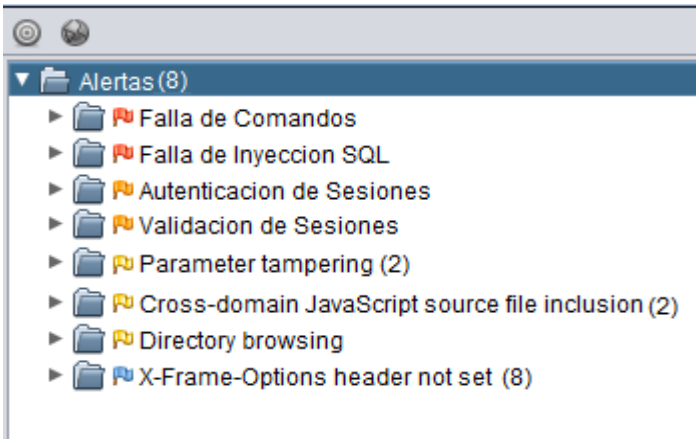


Figura IV. 26. Alertas encontradas

La herramienta OWASP ZAP detectó 8 alertas de las cuales se analizarán las 4 vulnerabilidades como un riesgo alto y medio las mismas que son: Falla de Comandos o Inyección de Comandos, Falla por Iyección SQL, Validación de Sesiones y Autenticacion de Sesiones siendo éstas la lista de vulnerabilidades reales.

- ✓ **Determinar la Severidad del Riesgo (GSAW)**

La severidad de Riesgo se lo realizó con el método DREAD en cada una de las Amenazas que se obtuvo en el paso anterior las cuales fueron: Falla de Comandos, Falla de Inyección SQL, Validación de Sesiones y Autenticación de Sesiones.

Tabla IV. XXXV.Severidad de Riesgo Amenaza 1

Amenaza	D	R	E	A	D	Total	Puntuación
Falla de Comandos	2	2	2	2	2	10	Medio

Como se puede observar en la Tabla IV.XXXV al aplicar el método DREAD en la amenaza haya de Comandos arroja como resultado un total de 10 por lo que el Riesgo que se obtiene es de MEDIO.

Tabla IV. XXXVI. Severidad de Riesgo Amenaza 2

Amenaza	D	R	E	A	D	Total	Puntuación
Falla Inyección SQL	3	2	2	2	3	13	Alta

Al aplicar el método DREAD en la amenaza Falla de Inyección SQL se obtienen como resultado un total de 13 y el Riesgo que se obtiene es ALTO como se puede observar en la Tabla IV.XXXVI

Tabla IV. XXXVII.Severidad de Riesgo Amenaza 3

Amenaza	D	R	E	A	D	Total	Puntuación
Validación de Sesiones	2	2	1	2	2	9	Medio

Al aplicar el método DREAD en la amenaza Falla de Validación de Sesiones se obtiene como resultado un total de 9 y el Riesgo que se obtiene es MEDIO como se observa en la Tabla IV.XXXVII.

Tabla IV. XXXVIII.Severidad de Riesgo Amenaza 4

Amenaza	**D**	**R**	**E**	**A**	**D**	**Total**	**Puntuación**
Autenticación de Sesiones	3	2	3	1	1	10	Medio

Se observa en la Tabla IV.XXXVIII que al aplicar el método DREAD en la amenaza autenticación de Sesiones arroja como resultado un total de 10 por lo que el Riesgo que se obtiene es de MEDIO.

CAPÍTULO V
ANÁLISIS DE RESULTADOS

El siguiente capítulo está enfocado en analizar los resultados obtenidos en el trabajo de tesis, para lo cual se analizó la aplicación de Control de Nuevos Aspirantes a la Empresa "GRUPO LAAR" en cada periodo de desarrollo, con ayuda de la Herramienta OWASP ZAP y el método de puntuación DREAD

5.1. Escenarios de Prueba

Se emplearon dos escenarios de pruebas diferentes es aquí donde se comprueba si al utilizar la guía de buenas prácticas GSAW ayuda en la reducción de vulnerabilidades, lo cual se logra atacando cada uno de los escenarios con la herramienta OWASP ZAP y el método de puntuación DREAD, a continuación se describe cada uno de los escenarios de prueba:

- ✓ **Escenario 1**

Es la aplicación en su primera instancia, como se lo estuvo utilizando en la empresa, el cual no cumplía con la seguridad adecuada en la misma se detectaron las vulnerabilidades de Falla de Comandos, Falla de Inyección SQL, Autenticación de Sesiones y Validacion de Sesiones con un riesgo de severidad ALTA

- ✓ **Escenario 2**

Se le aplicó la guía de buenas práctica GSAW en la cual se detectaron las vulnerabilidades de Falla de Comandos, Falla de Inyección SQL, Autenticacion de Sesiones y Validacion de Sesiones con una sevridad de riesgo MEDIA. Con esos resultados se aplica la recursividad de la guía y se obtiene que las vulnerabilidades hayan disminuido, la única detectada es Fallo de Inyección SQL con una severidad de riesgo BAJA.

5.2. Análisis Escenario 1.

Se atacó el Escenario 1 con la herramienta OWASP ZAP obteniendo como resultado 9 Alertas en las cuales se enumeran las vulnerabilidades encontradas y las alertas de información, entre las mas perjudiciales se tiene: Falla de Comandos, Falla de Inyección SQL, Autenticacion de Sesiones, Validacion de Sesiones.

Con el método de Puntuación DREAD podremos determinar el porcentaje y la severidad de los riesgos que tienen cada una de las vulnerabilidades de Falla de Comandos, Falla de Inyección SQL, Autenticación de Sesiones y Validacion de Sesiones, así que el análisis se lo realiza de manera individual. La vulnerabilidad de Falla de comandos en el cálculo de DREAD obtiene como total de riesgo el valor de 12 como se puede observar en la Tabla V.XXXIX

Tabla V. XXXIX.Severidad de Riesgo Amenaza 1

Amenaza	D	R	E	A	D	Total
Falla de Comandos	3	2	2	2	3	12

Se puede observar en la Tabla V.XL que el total obtenido con el cálculo de DREAD es como total de riesgo el valor de 15 en cuanto a la vulnerabilidad de Falla de inyección SQL.

Tabla V. XL. Severidad de Riesgo Amenaza 2

Amenaza	D	R	E	A	D	Total
Falla Inyección SQL	3	3	3	3	3	15

En la Tabla V.XLI el resulta obtenido de aplicar el método DREAD en la vulnerabilidad de Validación de Sesiones se obtiene como total de riesgo el valor de12.

Tabla V. XLI.Severidad de Riesgo Amenaza 3

Amenaza	D	R	E	A	D	Total
Validación de Sesiones	3	3	1	2	3	12

La vulnerabilidad de Autenticación de Sesiones en el cálculo de DREAD obtiene como total de riesgo el valor de 10 como se puede observar en la Tabla V.42

Tabla V. XLII.Severidad de Riesgo Amenaza 4

Amenaza	D	R	E	A	D	Total
Autenticación de Sesiones	3	2	3	1	1	10

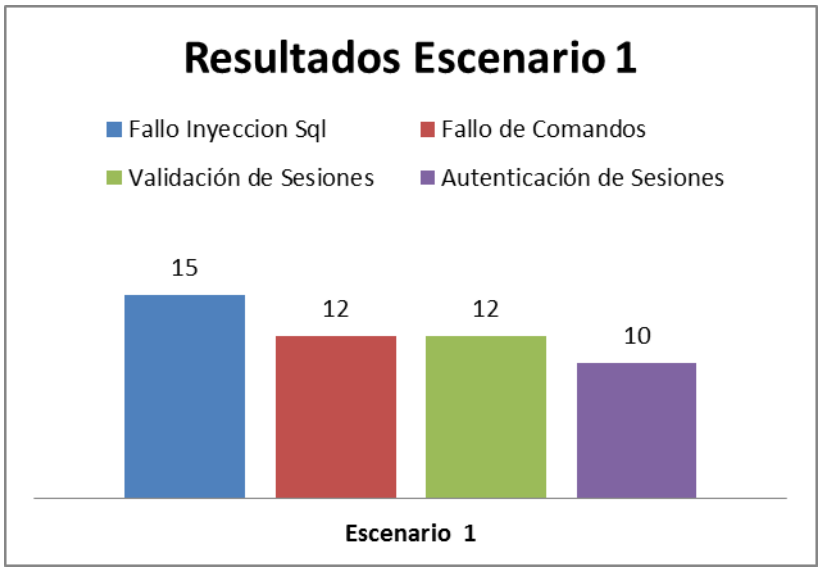


Figura V. 27.Resultado Escenario 1

La gráfica de la Figura V.27 muestra los resultados obtenidos en cada una de las vulnerabilidades, en la cual el Fallo de Inyección SQL obtuvo una severidad de riesgo más alta que el resto. De igual manera se puede apreciar en la Tabla V.XLIII con el método de puntuación Dread se determinó que en el escenario 1 existen amenazas en la aplicación con una severidad de Riesgo de Alto y Medio por lo que la valoración final que se tiene es de ALTA.

Tabla V. XLIII. Resultado DREAD Escenario 1

Amenaza	Puntuación
Falla de Comandos	Alto
Fallo de Inyección SQL	Alto
Validación de Sesiones	Alto
Autenticación de Sesiones	Medio
VALORACIÓN	ALTA

5.3. Análisis del Escenario 2

Al aplicar la herramienta OWASP ZAP en el escenario 2 se obtiene como resultado 8 Alertas en las cuales se enumeran las vulnerabilidades encontradas y las alertas de informacion, entre las mas perjudiciales se tiene: Falla de Comandos, Falla de Inyección SQL, Autenticacion de Sesiones, Validación de Sesiones

Se aplicó el método de Puntuación DREAD para determinar el porcentaje y la severidad de los riesgos que tienen cada una de las vulnerabilidades de Falla de Comandos, Falla de Inyección SQL, Autenticacion de Sesiones y Validación de Sesiones se las analiza una a una.

Se puede observar en la Tabla V.XLIV que el total de riesgo obtenido con el cálculo de DREAD es de 10 en cuanto a la vulnerabilidad de Falla de Comandos.

Tabla V. XLIV.Severidad de Riesgo Amenaza 1

Amenaza	**D**	**R**	**E**	**A**	**D**	**Total**
Falla de Comandos	2	2	2	2	2	10

Se obtuvo como valor de riesgo un total 13 al aplicar el método DREAD en la vulnerabilidad Falla de Inyección SQL como se muestra en la Figura V.XLV

Tabla V. XLV. Severidad de Riesgo Amenaza 2

Amenaza	**D**	**R**	**E**	**A**	**D**	**Total**
Falla Inyección SQL	3	2	2	2	3	13

En la Tabla V.XLVI el resultado obtenido de aplicar el método DREAD en la vulnerabilidad de Validación de Sesiones es de un riesgo total de 9.

Tabla V. XLVI.Severidad de Riesgo Amenaza 3

Amenaza	D	R	E	A	D	Total
Validación de Sesiones	2	2	1	2	2	9

Como se observa en la Tabla V.XLVII el resulta obtenido de aplicar el método DREAD en la vulnerabilidad de Autenticación de Sesiones se dé un riesgo total de 10.

Tabla V. XLVII.Severidad de Riesgo Amenaza 4

Amenaza	D	R	E	A	D	Total
Autenticación de Sesiones	3	2	3	1	1	10

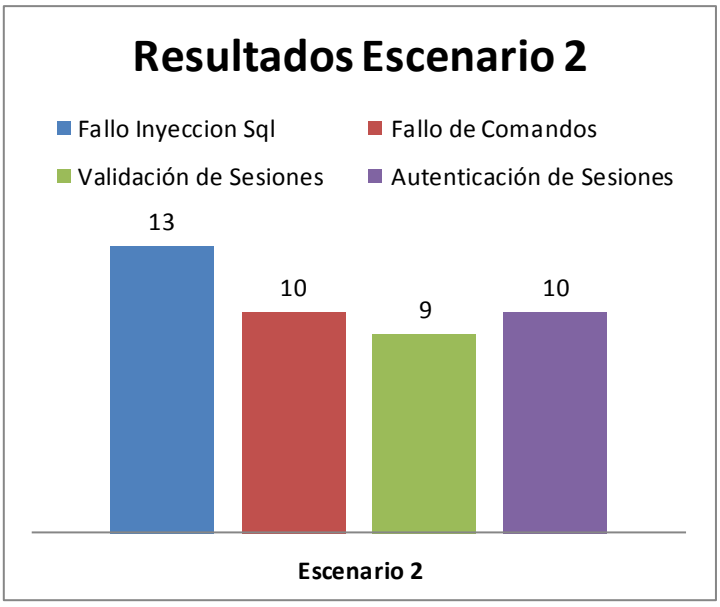


Figura V. 28.Resultado Escenario 2

En la FiguraV.28 se puede observar el resultado obtenido en el Escenario 2 donde el riesgo en las vulnerabilidades ha disminuido y de igual manera se puede decir que el Fallo de Inyección SQL es la que ocupa un riego alto en comparación de las demás.

En la Tabla V.XLVIII se puede observar que la severidad del riesgo que posee el escenario 2 está entre Medio y Alto por lo que se obtiene una valoración de riesgo de MEDIA.

Tabla V. XLVIII. Resultado DREAD Escenario 2

Amenaza	Puntuación
Falla de Comandos	Medio
Fallo de Inyección SQL	Alto
Validación de Sesiones	Medio
Autenticación de Sesiones	Medio
VALORACIÓN	MEDIA

Como en la aplicación aún existen vulnerabilidades se aplica la recursividad de la guía GSAW una vez que se termina de aplicar los pasos, se utiliza la herramienta OWASP ZAP y se obtuvo como resultado 3 Alertas, se enumeran la vulnerabilidad encontrada y las alertas de información. La vulnerabilidad Falla de Inyección SQL, es la que persiste en este escenario ya que con ayuda de GSAW se logró mitigar las otras 3 vulnerabilidades. Con el método de Puntuación DREAD se podrá determinar el porcentaje y la severidad de los riesgos que tienen la vulnerabilidad de Falla de Inyección SQL ya que las anteriores se lograron disminuir; como se observa en la Tabla IV.XLIX. se ha logrado un valor total de 5.

Tabla V. XLIX. Severidad de Riesgo Amenaza 2

Amenaza	D	R	E	A	D	Total
Falla Inyección SQL	1	1	1	1	1	5

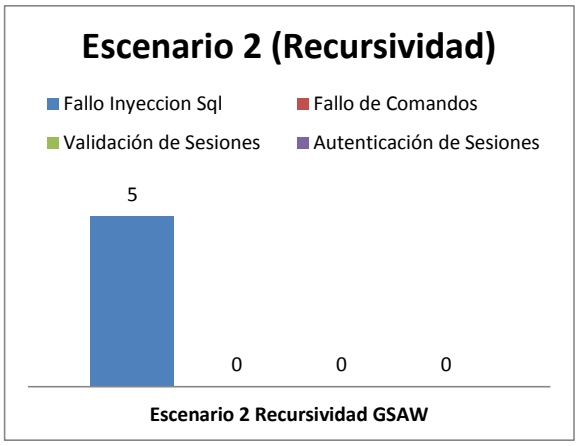


Figura V. 29. Resultado Escenario 2 Aplicando Recursividad de GSAW

Como se observa en la Figura V.29 el resultado de haber aplicado el método DREAD en la cual la vulnerabilidad que aún persiste es la de Fallo de Inyección SQL, y el resto de las vulnerabilidades se ha logrado mitigar.

Tabla V. L. Resultado DREAD Escenario 2 Aplicando Recursividad de GSAW

Amenaza	Puntuación
Falla de Inyección SQL	Bajo
VALORACIÓN	BAJA

Como se puede apreciar en la Tabla V.L la severidad de riesgo que se obtuvo en el escenario 2 aplicando la recursividad de GSAW es bajo con lo cual se tiene una severidad de riesgo BAJA.

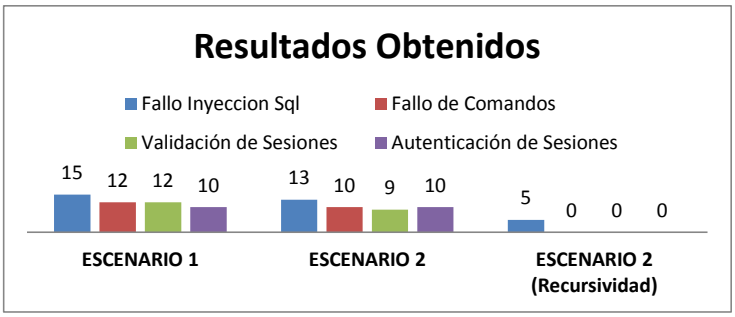


Figura V. 30. Resultados Obtenidos

Una vez analizado los escenarios podemos observar los resultados obtenidos de los mismos en la FiguraV.30 y se puede indicar que:

Escenario 1: Al utilizar la herramienta OWASP ZAP y el Método de puntuación DREAD en la aplicación de Control de Nuevos Aspirantes en la Empresa "GRUPO LAAR" se obtiene que el riesgo que existe en la aplicación es Alto por lo que las vulnerabilidades que existen representan un gran peligro.

Escenario 2: Al utilizar la herramienta OWASP ZAP y el Método de puntuación DREAD en la aplicación de Control de Nuevos Aspirantes en la Empresa "GRUPO LAAR" una vez que se aplicó la guía de buenas prácticas GSAW se obtiene que el riesgo que existe en la aplicación es Bajo y aunque en la aplicación aún existen la vulnerabilidad de Fallo de Inyección SQL no representan un riesgo alto.

Para el resultado final se toma la vulnerabilidad que persiste en el Esenario 2(Recursividad GSAW) y se la compara con las de los otros esenarios como se muestra en la TablaV.LI.

Tabla V. LI. Resultado Final

AMENAZA	ESCENARIO 1	ESCENARIO 2	ESCENARIO 2 (Recursividad)
Fallo de Inyección SQL	15	13	5

Como se observa en la FiguraV.LI el Fallo de Inyección SQL en el Esenario 2 (Rescursividad GSAW) disminuyó considerablemente a comparacion del Esenario 2 (GSAW) de MEDIO a BAJO y de igual manera el Escenario 2(GSAW) disminuyó a comparación del Escenario 1 de ALTO a MEDIO.

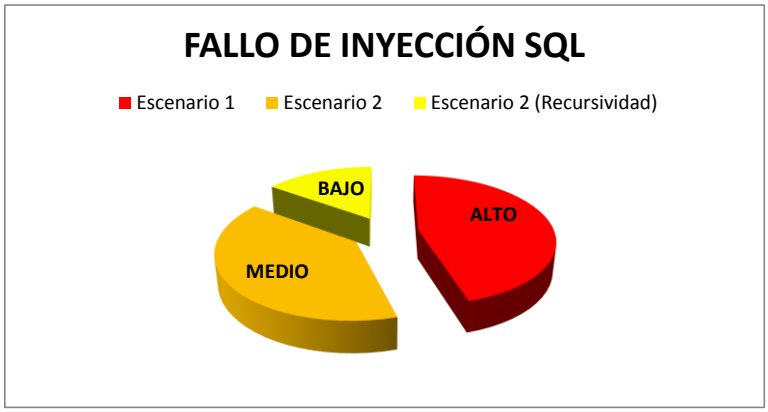


Figura V. 31. Resultado Final

De esta manera podemos decir que una vez que se empleara la guía de buenas prácticas GSAW en la aplicación de Control de Nuevos Aspirantes en la Empresa "GRUPO LAAR" se ha logrado reducir las vulnerabilidades de Manejo de Sesions e Interprete de Inyección. Por lo que se se acepta la hipótesis de la investigación.

CONCLUSIONES

- ✓ La realización de la guía de buenas prácticas GSAW se la realizó con el estudio de conceptos de seguridad, el análisis de las metodologías de riesgos y herramientas de verificación.

- ✓ La herramienta de verificación OWASP ZAP cumple con las características y funcionalidades necesarias para implementarla en la guía de buenas prácticas GSAW.

- ✓ Al implementar la guía de buenas prácticas en desarrollo en el Sistema de Control de Nuevos Aspirantes en la Empresa "GRUPO LAAR" se puede disminuir las vulnerabilidades de manejo de sesiones e intérprete de inyección.

- ✓ El riesgo de las vulnerabilidades de "Falla de Inyección SQL", "Falla de comandos", " Validación de Sesiones " y "Autenticación de Sesiones" en el Escenario 2 aplicando GSAW disminuyó a MEDIA

- ✓ El riesgo de las vulnerabilidades de "Falla de comandos", " Validación de Sesiones " y "Autenticación de Sesiones" en el Escenario 2 aplicando la recursividad de GSAW desapareció, aunque la vulnerabilidad de "Falla de Inyección SQL" persiste en la aplicación el riesgo de esta vulnerabilidad disminuyó a BAJA.

RECOMENDACIONES

- ✓ Incorporar principios y pilares de seguridad en el desarrollo de aplicaciones web, para dar un tratamiento adecuado a la seguridad que se desea implementar.

- ✓ Se recomienda que la elección de la herramienta de verificación se la debe realizar en base a su funcionalidad ya que no todas cuentan con las mismas características ni todas las penetraciones que estas realizan son satisfactorias.

- ✓ Para iniciar un análisis de rendimiento se las metodologías de riesgos se debe seleccionar cuidadosamente las mismas ya que existen diversas metodologías y cada una posee diferentes pasos en su desarrollo.

- ✓ La conformación del equipo de trabajo debe brindar el aporte necesario en base a sus conocimientos, habilidades, capacidades, información y así llegar a realizar un trabajo en conjunto acorde a las necesidades de la empresa.

- ✓ Se sugiere que para futuros trabajos investigativos se debe contemplar amenazas que no se trate en esta investigación, como son: Redirecciones y reenvíos no validados, falsificación de peticiones en dominios cruzados, entre otras.

- ✓ Aplicar la Guía Segura de Aplicaciones Web (GSAW) en el desarrollo de las aplicaciones web en la empresa GRUPO LAAR

RESUMEN

Se elaboró una guía de buenas prácticas para desarrollar aplicaciones web y reducir vulnerabilidades de manejo de sesiones e intérprete de inyección en el Sistema de Control de Nuevos Aspirantes en Empresa GRUPO LAAR de la ciudad Quito.

Se aplicó el método científico para elaborar la guía denominada Guía Segura de Aplicaciones Web (GSAW), que contiene principios y pilares de seguridad; se analizó metodologías de riesgos CORAS, PTA y ATAM en la elaboración, con el método experimental se desarrolló dos escenarios de pruebas, el Primero no contiene la guía (GSAW) y el otro si contiene la guía, en ellos se analizó vulnerabilidades existentes con la herramienta OWASP ZAP y el método de puntuación de riesgos DREAD.

Con estadística descriptiva se obtuvo los siguientes resultados: en escenario 1 existieron vulnerabilidades de Inyección de Código con riesgo 10, Inyección SQL con riesgo 13, Validación de Sesiones con riesgo 9 y Autenticación de Sesiones con riesgo 10 observándose que la severidad de riesgo es alta y mientras que escenario 2 se detectó solo la vulnerabilidad de Inyección SQL con riesgo 5 con una severidad de riesgo baja.

Al implementar la guía (GSAW) en el Sistema de Control de Nuevos Aspirantes en Empresa se logró disminuir vulnerabilidades en el manejo de sesiones e intérprete de inyección.

Se recomienda la utilización de la guía GSAW en el desarrollo de aplicaciones web.

Palabras clave:

/DESARROLLO DE APLICACIONES WEB/

/SEGURIDADES EN LA WEB/

/APLICACIONES WEB SEGURAS/

.

SUMARY

A guideline of good practice was elaborated to develop web applications and reduce session management vulnerabilities and interpreter of injection in Control System for New Business Aspirants in LAAR GROUP of the city Quito.

The scientific method was used to develop the guide called Secure Guide for Web Applications (SGWA) containing principles and security pillars it was using the experimental method test two scenarios were developed, the first contains the guidance (SGWA) and the other whether it contains the guide in them, existing vulnerabilities are analyzes, with the OWASP ZAP tool and risk scoring method DREAD.

With descriptive statistics the following results were obtained: in scenario 1 existed Injection Vulnerabilities Code risk 10 SQL Injection risk 13 Validation Sessions at risk 9 and Authentication Sessions at risk 10 and we observed that the severity of risk is high, while in scenario 2 was detected only SQL Injection vulnerability risk 5, with a low risk a low risk severity

By implementing the guide (SGWA) in the Control System for New Applicants for the Company was able to reduce vulnerabilities in the handling of sessions and interpreter injection.

The use of the guide SGWA is recommended in the development of web applications.

Keywords:
/SECURITIES ON THE WEB/
/WEB APPLICATION DEVELOPMENT/
/SECURE WEB APPLICATIONS/

GLOSARIO

AMENAZA: Un evento de posible ocurrencia que podría dañar o comprometer un activo o un objetivo estratégico.

VULNERABILIDAD: Una vulnerabilidad es un punto débil que de explotarse con éxito puede llegar a originar la consecución de una amenaza.

ATAQUE: Una acción que sirve de una o varias vulnerabilidades para materializar una amenaza.

RIESGO: la probabilidad (cuantificable) de sufrir una pérdida debido a una amenaza materializada. Su valor depende de dos factores: la frecuencia con la que ocurre la amenaza; el impacto del daño que puede causar.

INCIDENTE NO DESEADO: Evento que podría dañar o reducir el valor de los activos.

CONTRAMEDIDA: Se establece para hacer frente a las vulnerabilidades y reducir la probabilidad de sufrir un ataque o el impacto que pueda originarse de la consecución de una amenaza.

PRINCIPIO DE MENOR PRIVILEGIO: Se basa en no conceder más privilegios de los absolutamente necesarios.

Separación de privilegios: Evitar que las operaciones se basen en una única condición.

CONFIDENCIALIDAD: Sólo los usuarios debidamente autorizados deben tener acceso a los datos y/o los recursos de un modo a

INTEGRIDAD: Sólo los usuarios debidamente autorizados pueden modificar los datos y/o los recursos de modo apropiado y controlado.

DISPONIBILIDAD: Los recursos deben estar disponibles cuando son necesarios y deben funcionar a un nivel aceptable.

NO REPUDIO: Se basa en el hecho de que las acciones realizadas por los usuarios deben quedar registradas de un modo tal que estos no puedan negar posteriormente el haberlas realizado.

API (Interfaz de programación de aplicaciones): es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usados generalmente en las bibliotecas.

APLICACIONES WEB: aplicación web es aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

DOM: es esencialmente una interfaz de programación de aplicaciones (API) que proporciona un conjunto estándar de objetos para representar documentos HTML y XML.

HISTORIAS DE USUARIO: Una historia de usuario es una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario.

HTML: es el lenguaje de marcado p para la elaboración de páginas web. Es usado para describir la estructura y do en forma de texto, así como para complementar el texto con objetos tales como imágenes.

METODOLOGÍA XP: Es una metodología de desarrollo de la ingeniería de software, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.

XML: (Lenguaje de Marcas Extensible), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específico.

OPEN SOURCE: Código abierto es la expresión con la que se conoce al software distribuido y desarrollado libremente. Se focaliza más en los beneficios prácticos (acceso al código fuente) que en cuestiones éticas o de libertad que tanto se destacan en el software libre.

NAVEGADOR: Un navegador o navegador web, o browser, es un software que permite el acceso a Internet, interpretando la información de archivos y sitios web para que éstos puedan ser leídos

SERVIDOR WEB: Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente.

GESTOR DE BASE DE DATOS: son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

ONLINE: El término en línea (online)] cia a un estado de conectividad, frente al término fuera de línea (offline) que indica un estado de desconexión.

DEBUG: define depuración, que es un proceso de leer todos los comandos que se involucran al presentar una página de internet o un programa instalado.

HACKERS: Gente apasionada por la seguridad informática. Esto concierne principalmente a entradas remotas no autorizadas por medio de redes de comunicación como Internet (*"Black hats"*). Pero también incluye a aquellos que depuran y arreglan errores en los sistemas (*"White hats"*) y a los de moral ambigua como son los *"Grey hats"*.

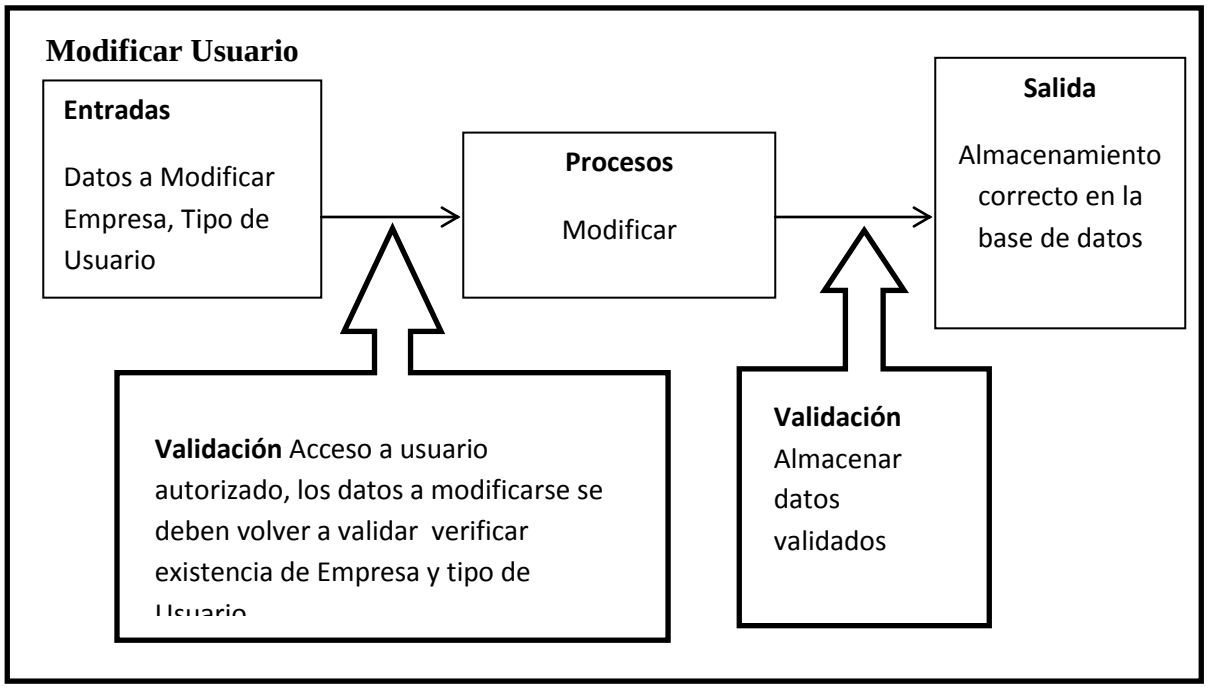
TOKENS: componente léxico es una cadena de caracteres que tiene un significado coherente en cierto lenguaje de programación. Son los elementos más básicos sobre los cuales se desarrolla toda traducción de un programa, surgen en la primera fase, llamada análisis léxico, sin embargo se siguen utilizando en las siguientes fases (análisis sintáctico y análisis semántico) antes de perderse en la fase de síntesis.

ANEXOS

Anexo 1.

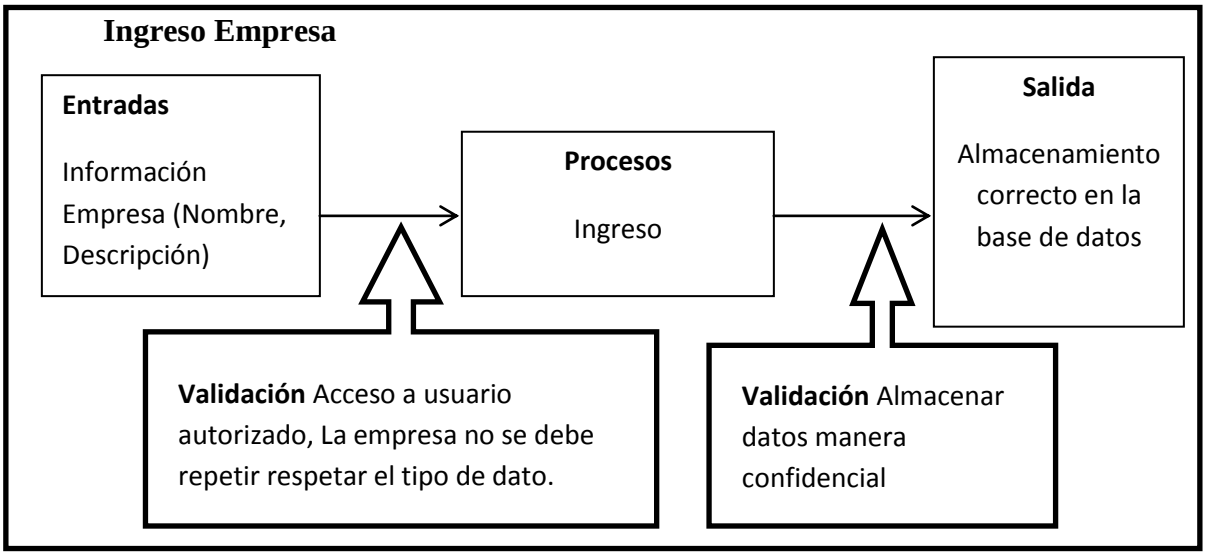
FUNCIONALIDAD DE LA APLICACIÓN

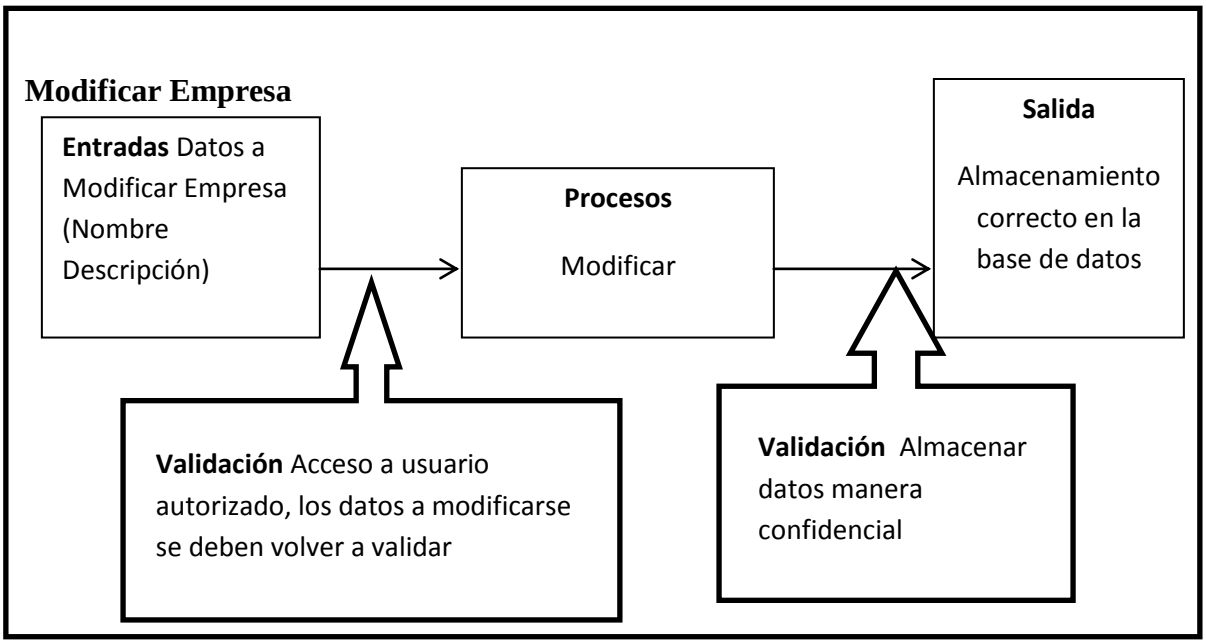
Se determinó las entradas, salidas y los procesos de cada requerimiento como se puede observar en las Tablas () y Figuras ()



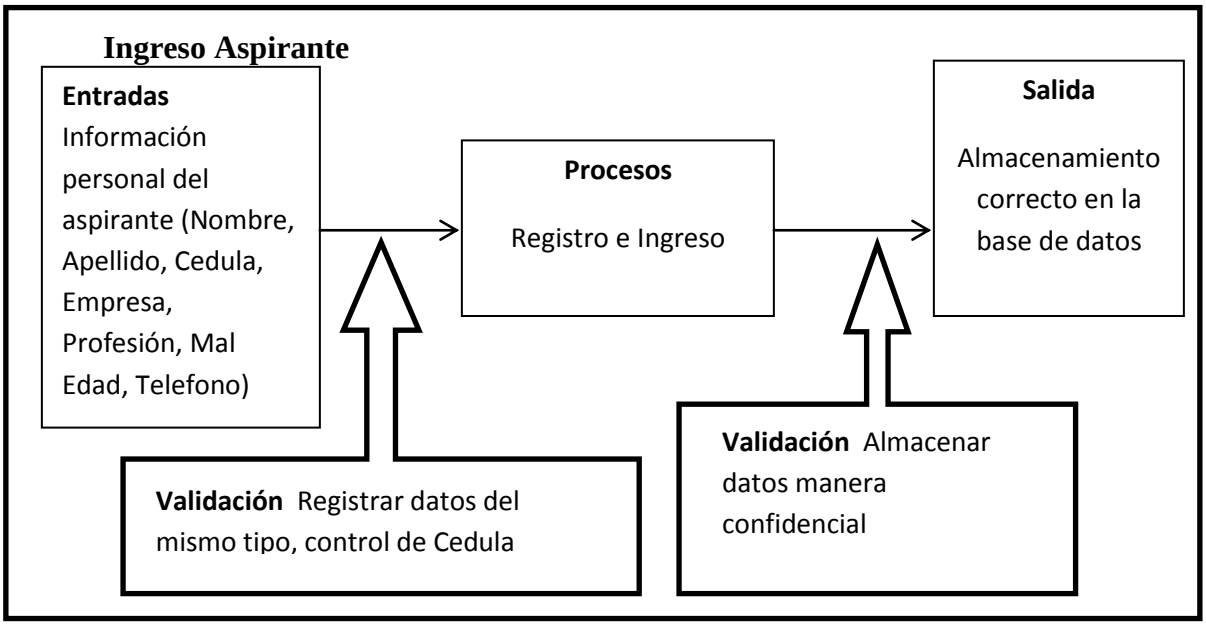
Requerimiento 2: Gestionar empresas (Ingreso, Modificación)		
TIPO DE DATO	**ENTRADA**	**SALIDAS**

Cadena de Caracteres	Datos por parte de la Empresa	Envió a la base de datos



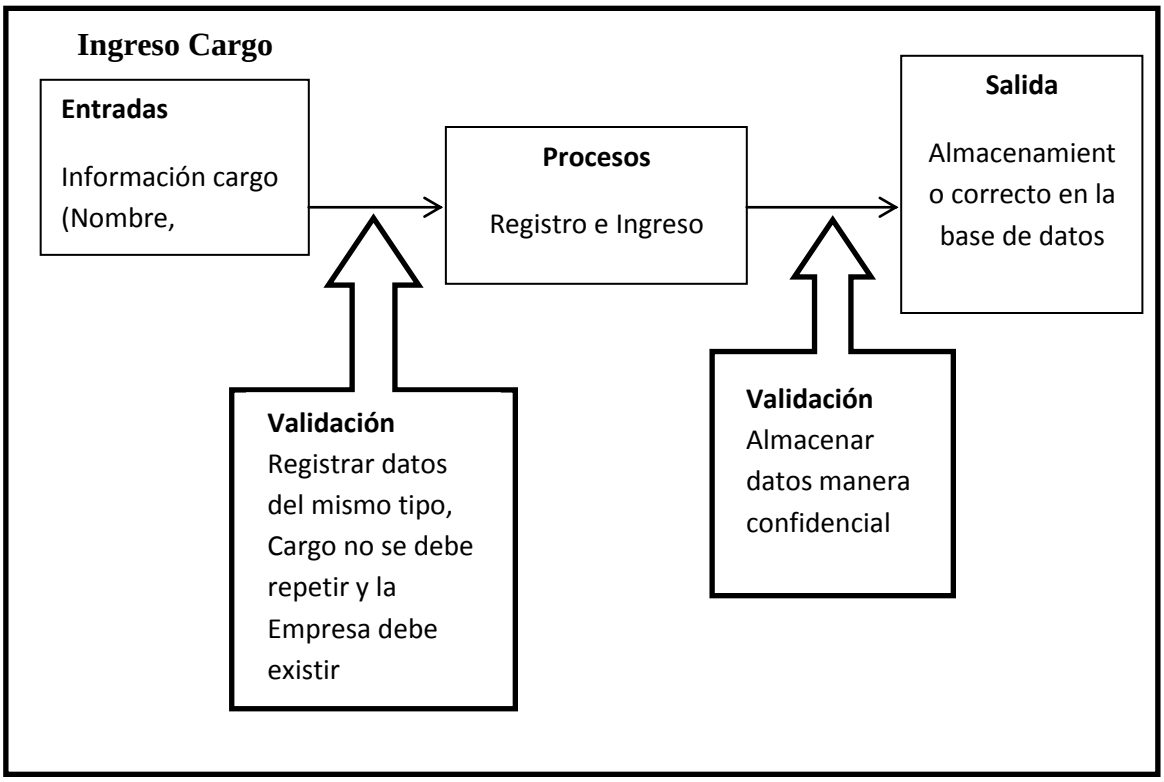


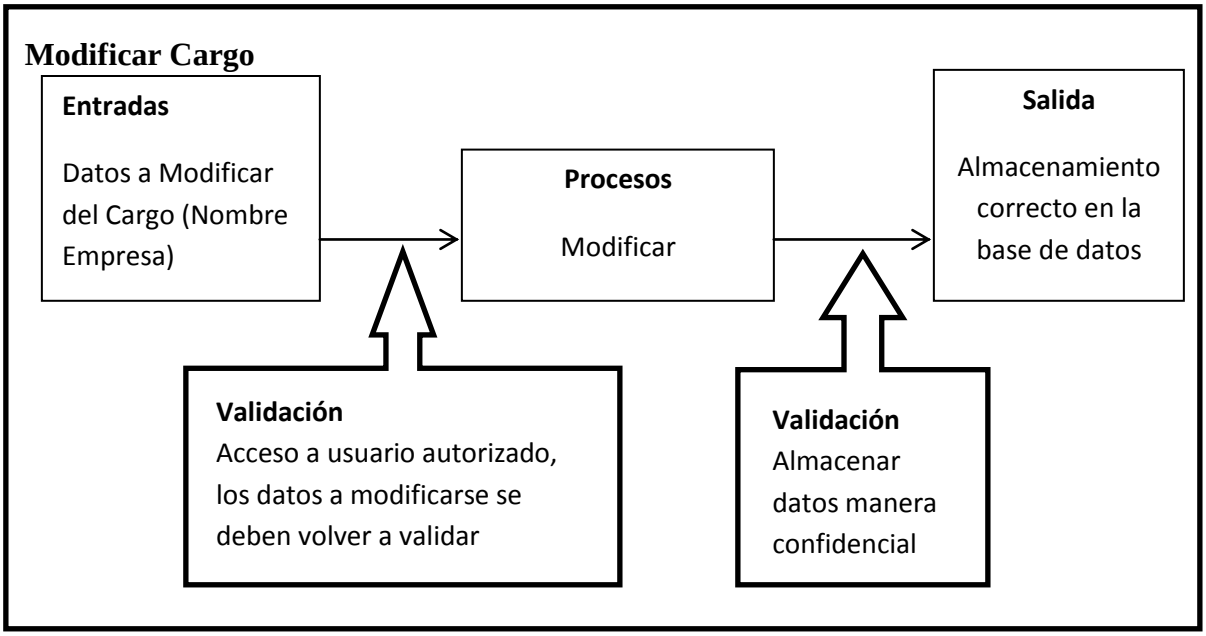
Requerimiento 3: Gestionar Aspirante (Ingreso)		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres Números	Datos por parte de los Aspirantes	Envió a la base de datos



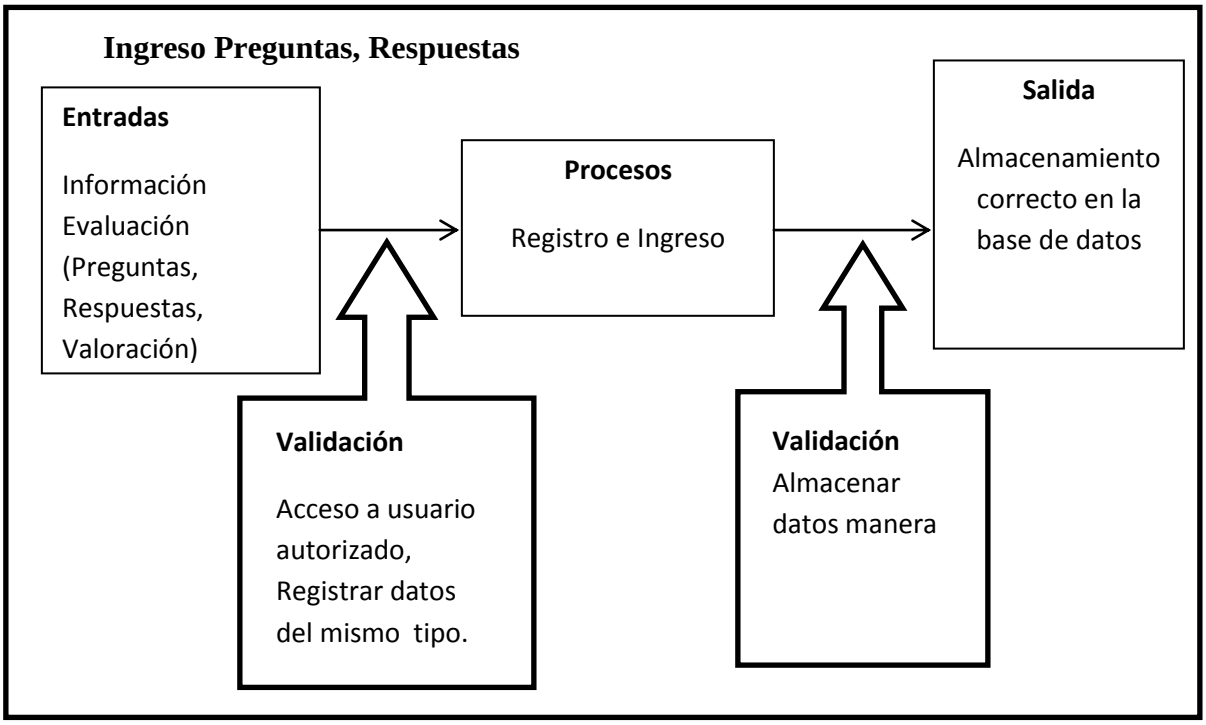
Requerimiento 4: Gestionar cargos a evaluar.		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres	Datos por parte de los Cargos	Envió a la base de datos

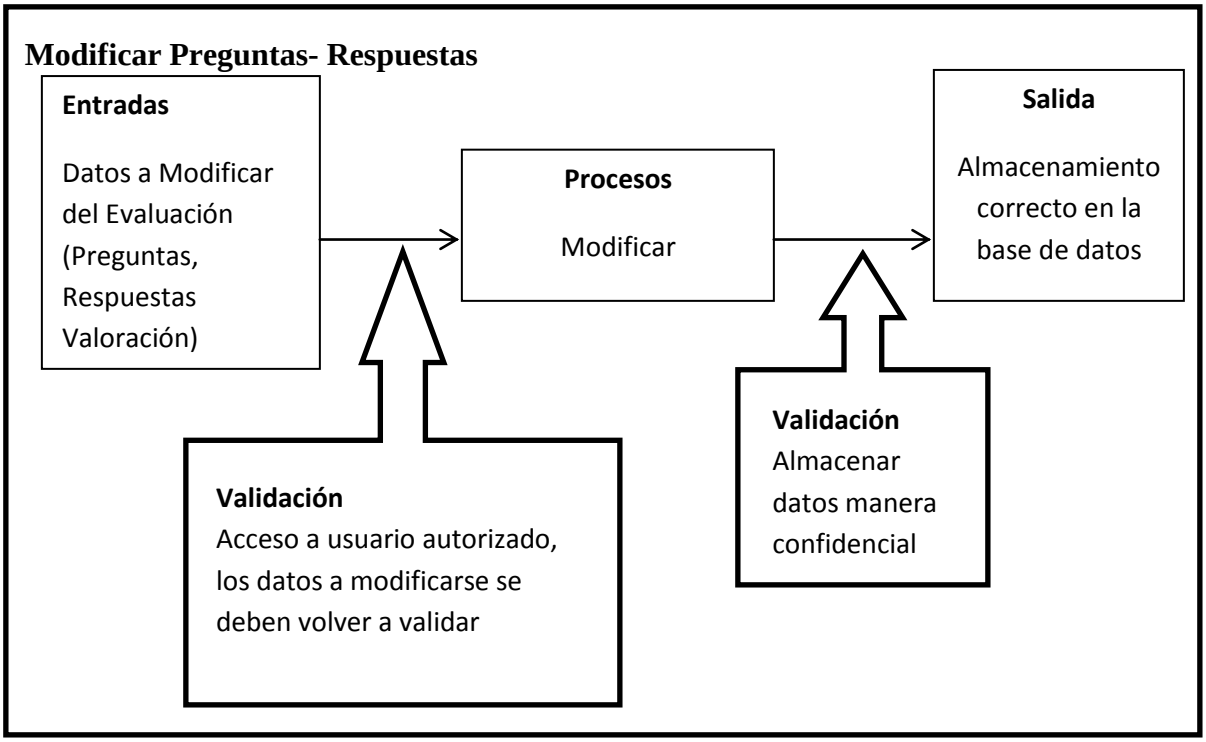
Números		


Ingreso Cargo
Entradas
Información cargo (Nombre,
Procesos
Registro e Ingreso
Salida
Almacenamiento correcto en la base de datos
Validación
Registrar datos del mismo tipo, Cargo no se debe repetir y la Empresa debe existir
Validación
Almacenar datos manera confidencial

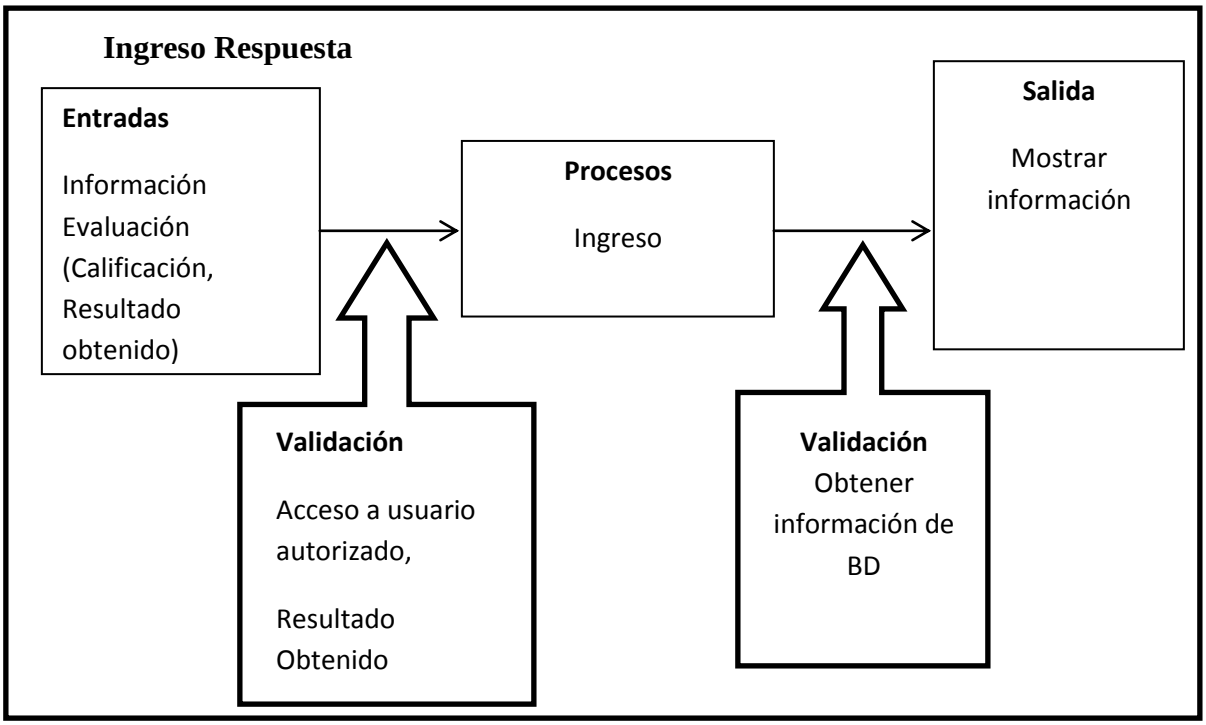

Modificar Cargo
Entradas
Datos a Modificar del Cargo (Nombre Empresa)
Procesos
Modificar
Salida
Almacenamiento correcto en la base de datos
Validación
Acceso a usuario autorizado, los datos a modificarse se deben volver a validar
Validación
Almacenar datos manera confidencial

Requerimiento 5: Permitir el ingreso de preguntas, respuestas y calificación de las mismas para elaborar las evaluaciones.		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres Números	Datos por parte de la Evaluación, Preguntas, respuestas.	Envió a la base de datos Generación de la Evaluación



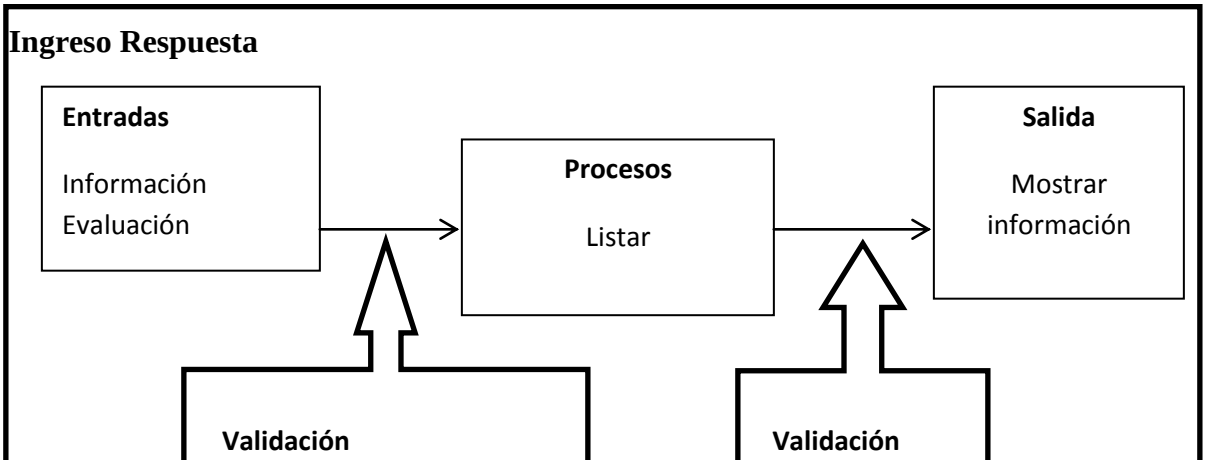


Requerimiento 6: Permitir dar respuesta a los aspirantes sobre las evaluaciones rendidas		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres	Datos por parte de la Evaluación,	Mostrar Respuesta

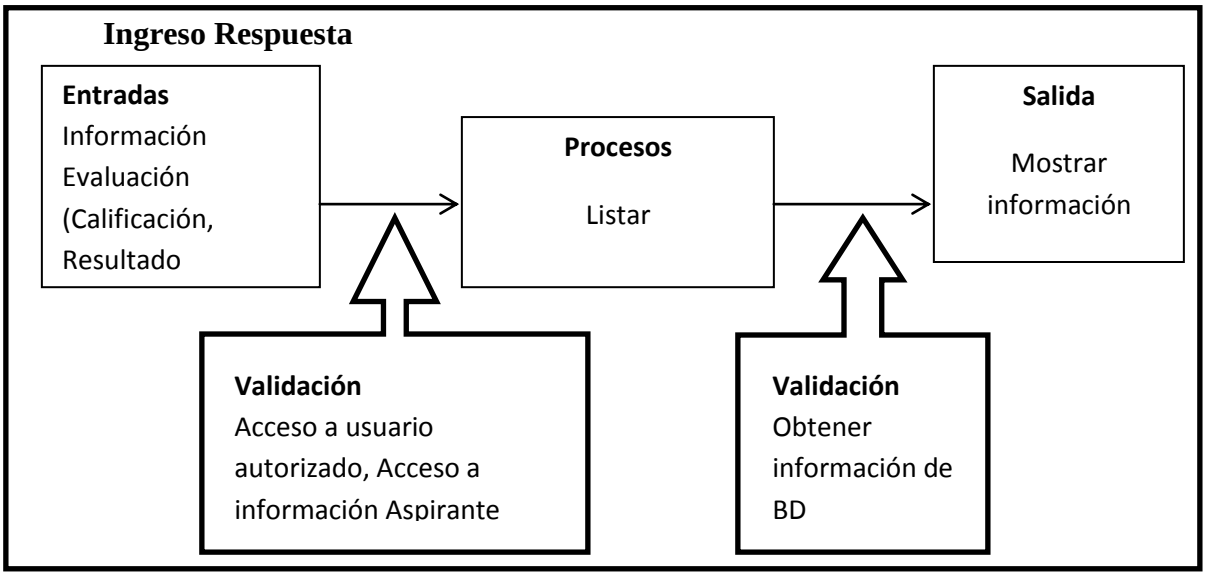


7

Requerimiento 7: El sistema debe permitir realizar reportes de evaluaciones		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres	Datos por parte de la Evaluación,	Mostrar Reporte



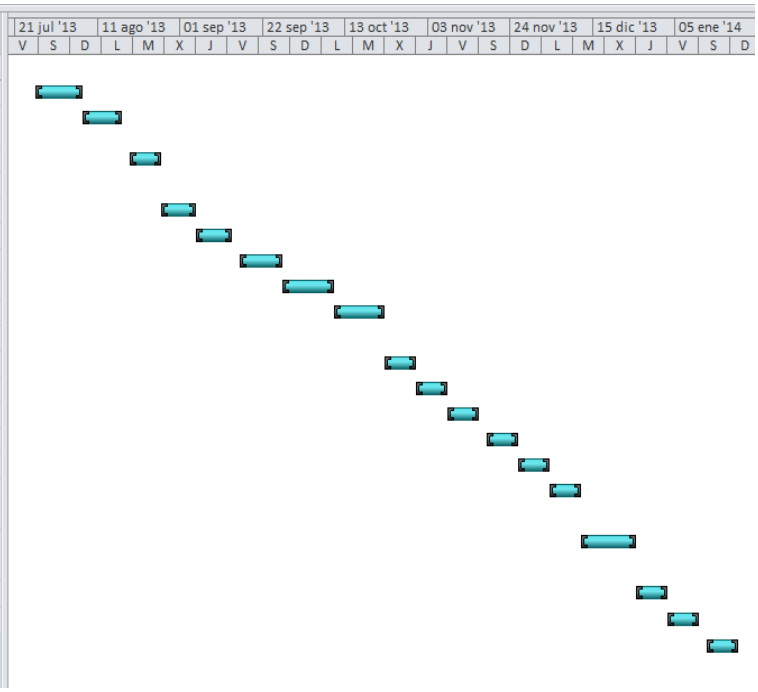
Requerimiento 8: El sistema debe permitir realizar reportes los aspirantes		
TIPO DE DATO	**ENTRADA**	**SALIDAS**
Cadena de Caracteres	Datos por parte de la Aspirantes	Mostrar Reporte



Anexo 2

DISEÑO DE LA PLANIFICACIÓN

Nombre de tarea	Duración	Comienzo	Fin
ITERACIÓN 1			
Registro de Aspirantes	8 días	vie 26/07/13	mar 06/08/13
Registro de Usuarios GRUPO LAAR	8 días	mié 07/08/13	vie 16/08/13
Autenticación de usuarios	6 días	lun 19/08/13	lun 26/08/13
ITERACIÓN 2			
Registro de empresas	7 días	mar 27/08/13	mié 04/09/13
Registroo de cargos	7 días	jue 05/09/13	vie 13/09/13
Registro de pruebas	9 días	lun 16/09/13	jue 26/09/13
Registro de preguntas	9 días	vie 27/09/13	mié 09/10/13
Registro de respuestas	9 días	jue 10/10/13	mar 22/10/13
INTERACIÓN 3			
Modificar Usuario	6 días	mié 23/10/13	mié 30/10/13
Modificar empresas	6 días	jue 31/10/13	jue 07/11/13
Modificar cargos	6 días	vie 08/11/13	vie 15/11/13
Modificar pruebas	6 días	lun 18/11/13	lun 25/11/13
Modificar preguntas	6 días	mar 26/11/13	mar 03/12/13
Modificar respuestas	6 días	mié 04/12/13	mié 11/12/13
ITERACIÓN 4			
Rendir la evaluación al aspirante	10 días	jue 12/12/13	mié 25/12/13
Dar respuesta aspirantes	6 días	jue 26/12/13	jue 02/01/14
Reporte los aspirantes	6 días	vie 03/01/14	vie 10/01/14
Reporte de la Evaluación de un aspirante	6 días	lun 13/01/14	lun 20/01/14



Anexo 3

HISTORIAL DE USUARIOS

REGISTRO ASPIRANTE

Historia de Usuario	
Número: 1	Nombre de la historia: Registro aspirante
Creación de historia de usuario	
Usuario: Aspirante	Iteración Asignada: 1
Prioridad en el Negocio: Alta	Puntos Estimados:8
Riesgo en el Desarrollo: Bajo	Puntos Reales:10
Descripción: El sistema permitirá el registro de aspirantes.	
Observaciones: Se registrara la siguiente información de cada aspirante: empresa, cargo, cedula, nombre, apellido, dirección, teléfono, estado, mail, password	
Pruebas de Aceptación: Al ingresar un código (cedula) de un aspirante ya existente, se emitirá un mensaje de error. Al ingresar los datos erróneo se emitirá un mensaje de confirmación que los datos Inserción del Aspirante fallida Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Registro aspirante	
Número de Tarea: 1	Nombre de Tarea: Creación de los métodos necesarios en clase ADatos de la capa de Acceso_Datos y la creación de la clase aspirante

Tipo de Tarea: Desarrollo.	Puntos Estimados: 3
Fecha Inicio: 26/07/2013	Fecha Fin:30/07/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarregistroaspirante en la clase ADatos en la capa de Acceso_Datos	
Pruebas de Aceptación. Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	

Tarea de Ingeniería		
Historia de Usuario: Registro aspirante		
Número de Tarea: 2	Nombre de Tarea: Creación del método insertarregistroaspirante en clase LNegocios de la capa de Logica_Negocio.	
Tipo de Tarea: Desarrollo.		Puntos Estimados: 2
Fecha Inicio: 31/07/2013		Fecha Fin: 01/08/2013
Programador Responsable: Elvira Yánez		
Descripción: Crearemos el método web insertarregistroaspirante en la clase LNegocios en la capa de Logica_Negocio.		
Pruebas de Aceptación. Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.		

Tarea de Ingeniería	
Historia de Usuario: Registro de aspirante	
Número de Tarea: 4	Nombre de Tarea: Creación de las páginas web en la capa Presentacion necesarias para ingresar un aspirante

Tipo de Tarea: Desarrollo.	Puntos Estimados: 3
Fecha Inicio:02/08/2013	Fecha Fin:06/08/2013
Programador Responsable: Elvira Yane2	
Descripción: Creación de las paginas RegistroAspirante.php en la capa Presentación, las mismas que permiten capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Registro aspirante
Nombre: Al ingresar un código (cédula) de un aspirante ya existente, se emite un mensaje de error.	
Responsable: Elvira Yánez	Fecha: 04/11/2013
Descripción: Al ingresar una cédula que ya existe, se mostrara un mensaje donde dirá La Cedula ya existe	
Condiciones de Ejecución: Insertar un aspirante con una CEDULA inexistente	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón guardar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: Registro aspirante
Nombre: Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	
Responsable: Elvira Yánez	Fecha: 04/11/2013
Descripción: Verificar el Despliegue del mensaje “Inserción Exitosa!” del método insertarregistroaspirante implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un aspirante	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón enviar. 3. Se visualiza el mensaje “Inserción Exitosa!”.	
Resultado esperado: Emitir el mensaje “Inserción Exitosa!”.	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:3	Historia de Usuario: Registro aspirante
Nombre: Despliegue del mensaje "Falla al Insertar Aspirante" al intentar ingresar un aspirante con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 04/11/2013
Descripción: Verificar el Despliegue del mensaje "Falla al Insertar Aspirante" del método insertarregistroaspirante implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un aspirante , con error en el campo Cédula	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón enviar. 3. Se visualiza el mensaje "Falla al Insertar Aspirante" en caso que la Cédula este mal o el aspirante ya exista	
Resultado esperado: Emitir el mensaje de "Falla al Insertar Aspirante	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Acceso Datos	
AccesoDatos() Conectar() Desconectar() EjecutarSelect() EjecutarUpdate()	Colaboradores:

Nombre de la Clase: ADatos	
Responsabilidades: Empresa, cargo. cedula, nombre, apellido, dirección, teléfono, mail, estado, password, insertarregistroaspirante ()	Colaboradores: AccesoDatos

Aspirante	
Responsabilidad	Colaboración
Empresa, cargo. cedula, nombre, apellido, dirección, teléfono, mail,	

estado, password, Get() Set()	

REGISTRO DE USUARIO GRUPO LAAR

Historia de Usuario	
Número: 2	Nombre de la historia: Registro usuario GRUPO LAAR
Creación de historia de usuario	
Usuario: Administrador	Iteración Asignada: 1
Prioridad en el Negocio: Alta	Puntos Estimados:8
Riesgo en el Desarrollo: Bajo	Puntos Reales:10
Descripción: El sistema permitirá el registro de usuario GRUPO LAAR	
Observaciones: Se registrara la siguiente información de cada usuario GRUPO LAAR: empresa, cédula, nombre, apellido, estado, password	
Pruebas de Aceptación: Al ingresar un código (cedula) de un usuario GRUPO LAAR ya existente, se emitirá un mensaje de error. Al ingresar los datos erróneo se emitirá un mensaje de confirmación que los datos Inserción del Usuario fallida Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.	

TAREA DE INGENIERÍA.

Tarea de Ingeniería
Historia de Usuario: Registro usuario GRUPO LAAR

Número de Tarea: 1	Nombre de Tarea: Creación de los métodos necesarios en clase ADatos de la capa de Acceso_Datos y la creación de la clase Usuario
Tipo de Tarea: Desarrollo.	Puntos Estimados: 3
Fecha Inicio:07/08/2013	Fecha Fin:09/08/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarusuario en la clase ADatos en la capa de Acceso_Datos	
Pruebas de Aceptación. Despliegue del mensaje "Falla al Insertar Usuario" al intentar ingresar un aspirante con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	
Tarea de Ingeniería	
Historia de Usuario: Registro usuario GRUPO LAAR	
Número de Tarea: 2	Nombre de Tarea: Creación del método insertarusuario en clase LNegocios de la capa de Logica_Negocio.
Tipo de Tarea: Desarrollo.	Puntos Estimados: 3
Fecha Inicio: 12/08/2013	Fecha Fin: 14/08/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarusuario en la clase LNegocios en la capa de Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Falla al Insertar Usuario" al intentar ingresar un aspirante con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	

Tarea de Ingeniería
Historia de Usuario: Registro usuario GRUPO LAAR

Número de Tarea: 4	Nombre de Tarea: Creación de las páginas web en la capa Presentación necesarias para ingresar un aspirante
Tipo de Tarea: Desarrollo.	Puntos Estimados: 2
Fecha Inicio:15/10/2013	Fecha Fin:16/11/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación de las pagina uario.php en la capa Presentación, las mismas que permiten capturar, p.ocesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Falla al Insertar Usuario" al intentar ingresar un aspirante con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa!” al ingresar un aspirante con datos correctos.	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Registro usuario GRUPO LAAR
Nombre: Al ingresar un código (cédula) de un usuario GRUPO LAAR ya existente, se emite un mensaje de error.	
Responsable: Elvira Yánez	Fecha: 05/11/2013
Descripción: Al ingresar una cédula que ya existe, se mostrara un mensaje donde dirá La Cedula ya existe	
Condiciones de Ejecución: Insertar un usuario con una CEDULA inexistente	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón guardar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:

Código:2	Historia de Usuario: Registro usuario GRUPO LAAR
Nombre: Despliegue del mensaje “Inserción Exitosa” al ingresar un usuario con datos correctos.	
Responsable: Elvira Yánez	Fecha: 05/11/2013
Descripción: Verificar el Despliegue del mensaje “Inserción Exitosa” del método insertarusurio implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un usuario	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón enviar. 3. Se visualiza el mensaje “Inserción Exitosa!”.	
Resultado esperado: Emitir el mensaje “Inserción Exitosa!”.	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:3	Historia de Usuario: Registro usuario GRUPO LAAR
Nombre: Despliegue del mensaje "Falla al Insertar Usuario" al intentar ingresar un aspirante con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 05/11/2013
Descripción: Verificar el Despliegue del mensaje "Falla al Insertar Aspirante" del método insertarrusuario implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un usuario , con error en el campo Cédula	
Pasos de ejecución: 1. Ingresar los datos de un aspirante. 2. Clic en el botón enviar. 3. Se visualiza el mensaje "Falla al Insertar Usuario" en caso que la Cédula este mal o el usuario ya exista	
Resultado esperado: Emitir el mensaje de "Falla al Insertar Usuario	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Acceso Datos	
AccesoDatos() Conectar() Desconectar()	Colaboradores:

Nombre de la Clase: ADatos	
Responsabilidades: empresa, cedula, nombre, apellido, estado,password, insertarusuario ()	Colaboradores: AccesoDatos

Aspirante	
Responsabilidad	Colaboración
empresa, cedula, nombre, apellido, estado, password, Get() Set()	

REGISTRO DE EMPRESAS

Historia de Usuario		
Número: 3	Nombre de la historia: Registro de Empresa	
Creación de historia de usuario		
Usuario: Administrador		Iteración Asignada: 2
Prioridad en el Negocio: Alta		Puntos Estimados: 7
Riesgo en el Desarrollo: Bajo		Puntos Reales: 8

Descripción: El sistema permitirá el registro de empresa.
Observaciones: Se registrara la siguiente i de cada empresa: id; nombre; descripción y estado
Pruebas de Aceptación: Al ingresar un nombre de empresa ya existente o dejar campos vacíos, se emitirá un mensaje de error. Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.

TAREA DE INGENIERIA

Tarea de Ingeniería		
Historia de Usuario: Registro de empresa		
Número de Tarea:1	Nombre de Tarea: Creación de los métodos en clase ADatos de la capa de Acceso_Datos y la creación de la clase Empresa	
Tipo de Tarea: Desarrollo		Puntos Estimados: 2
Fecha Inicio: 27/08/2013		Fecha Fin: 28/08/2013
Programador Responsable: Elvira Yánez		
Descripción: Se implementará el método insertarempresa en la clase ADatos en la capa Acceso_Datos, el cual recibe como parámetro el código del producto para realiza su respectiva consulta por medio de una Sentencia SQL.		
Pruebas de Aceptación El tipo de datos sean correctos Los datos estén completos.		

Tarea de Ingeniería		
Historia de Usuario: Registro de empresa		
Número de Tarea:2	Nombre de Tarea: Creación del método insertarempresa en la clase LNegocio de la capa de Logica_Negocio.	
Tipo de Tarea: Desarrollo		Puntos Estimados: 2

Fecha Inicio: 29/08/2013	Fecha Fin: 30/08/2013
Programador Responsable: Elvira Yánez	
Descripción: Se implementará método insertempresa en la clase LNegocio de la capa Logica_negocio la cual recibe como parámetro el Id de la Empresa para realiza su respectiva consulta atreves del servicio que consume de la capa Acceso_Datos.	
Pruebas de Aceptación Correcta conexión de la base. Verificar que los datos se hayan guardado correctamente. Despliegue del mensaje "Inserción de la Empresa Exitosa!" al ingresar un empresa con datos correctos.	

Tarea de Ingeniería		
Historia de Usuario: Registro de empresa		
Número de Tarea:3	Nombre de Tarea: Creación de las páginas web en la capa wsinterfaces, necesarias para ingresar un producto	
Tipo de Tarea: Desarrollo		Puntos Estimados: 2
Fecha Inicio: 02/09/2013		Fecha Fin: 04/09/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación de las paginas IngresarEmpresa.php capa Presentación, las mismas que permitirán capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.		
Pruebas de Aceptación Enviar correctamente los datos al método en la capa de acceso de datos. Gestionar los datos recibidos correctamente de la capa de aplicación.		

PRUEBAS DE ACEPTACIÓN

Prueba de Aceptación: Ingresar una empresa y emitir el mensaje correspondiente, " Inserción exitosa"	
Código: PA1	Historia de Usuario: Ingreso de empresa

Nombre: Ingresar una empresa y emitir correspondiente, “Inserción exitosa”.
Responsable: Elvira Yánez | Fecha: 05/11/2013
Descripción: Al momento de guardar los datos de la empresa debe emitir un mensaje satisfactorio.
Condiciones de Ejecución: no existir en la base de datos la misma información registrada
Pasos de ejecución: 1. Ingresar los datos de la empresa. 2. Clic en el botón enviar. 3. Se visualiza mensaje satisfactorio de ingreso o de error en caso de ya existir el producto en la base de datos.
Resultado esperado: Emitir el mensaje “ Inserción exitosa ”,
Evaluación de la prueba: satisfactorio.

Prueba de Aceptación: Emitir un mensaje de advertencia “Faltan campos por llenar” al momento de guardar los datos de la empresa con algunos campos en blanco.	
Código: PA2	Historia de Usuario: Ingreso de empresa.
Nombre: Emitir un mensaje de advertencia “Faltan campos por llenar” al momento de guardar	
Responsable: Elvira Yánez	Fecha: 05/11/2013
Descripción: Al momento de guardar el producto con algunos campos en blanco se debe emitir un mensaje de error “Faltan campos por llenar”.	
Condiciones de Ejecución: no existir en la base de datos la misma información registrada	
Pasos de ejecución: 1. Vamos a la interfaz de usuario de en el menú ingreso de empresa 2. Ingresar los datos de la empresa con algunos campos vacíos 3. clic en el botón enviar	
Resultado esperado: Emitir el mensaje de advertencia “Faltan campos por llenar”.	
Evaluación de la prueba: satisfactorio.	

Tarjeta CRC

Nombre de la Clase: AccesoDatos	
Métodos: AccesoDatos() get() set() Conectar() Desconectar() EjecutarSelect() EjecutarUpdate()	Colaboradores:

Nombre de la Clase: ADatos	
Responsabilidades: Atributos: id; nombre; descripción estado; Métodos: Set() Get() insertarempresa()	Colaboradores: AccesoDatos

Nombre de la Clase Empresa	
Responsabilidad	Colaboración
id; nombre; descripción estado; Set() Get()	

REGISTRO CARGO

Historia De Usuario	
Número: 5	Nombre de la Historia: Registro cargo
Creación de la Historia de Usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 2
Prioridad en el Negocio: Baja	Puntos Estimados: 7
Riesgo de Desarrollo: Media	Puntos Reales: 10
Descripción: Se desea registrar los cargos de la empresa	
Observaciones: Se registrará la siguiente información: id, nombres, empresa,, estado	
Pruebas de Aceptación: Al dejar campos vacíos en el formulario de ingreso nos emitirá un mensaje de error indicándonos los campos vacíos Al ingresar correctamente los campos del formulario de ingreso e intentar guardar se emitirá un mensaje de error “Inserción exitosa”	

TAREA DE INGENIERIA

Tarea de Ingeniería	
Historia de Usuario: Registro cargo	
Número de Tarea: 1	Nombre de Tarea: Creación de los métodos necesarios en la clase ADatos de la capa Acceso_Datos. y creación de la clase cargo
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Fecha Inicio: 05/09/2013	Fecha Fin: 06/09/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación del método insertarcargo en la clase ADatos de la capa Acceso_Datos y la clase cargos.	

Pruebas de Aceptación. Despliegue del mensaje “Inserción Exitosa” al ingresar un cargo con datos correctos. Despliegue del mensaje “Inserción Fallida” al intentar ingresar un cargo con datos incompletos o erróneos.

Tarea de Ingeniería	
Historia de Usuario: Registro cargo	
Número de Tarea: 2	Nombre de Tarea: Creación de los métodos en la clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Fecha Inicio: 09/09/2013	Fecha Fin: 10/09/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación del método insertarcargo en la clase LNegocio), de la capa Logica_Negocio	
Pruebas de Aceptación. Despliegue del mensaje “Inserción Exitosa” al ingresar un cargo con datos correctos. Despliegue del mensaje “Inserción Fallida” al intentar ingresar un cliente con datos incompletos o erróneos.	

Tarea de Ingeniería	
Historia de Usuario: Registro cargo	
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web en la capa Presentación , necesarias para ingresar un cargo
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Fecha Inicio: 11/09/2013	Fecha Fin: 13/09/2013
Programador Responsable: Carlos Mejía	
Descripción: Creación de las paginas IIngresarCargo.php en la capa Presentación las mismas que permitirán capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.	

Pruebas de Aceptación. Despliegue del mensaje “Inserción Exitosa” al ingresar un cargo con datos correctos. Despliegue del mensaje “Inserción Fallida” al intentar ingresar un cliente con datos incompletos o erróneos.

PRUEBAS DE ACEPTACIÓN

Prueba de Aceptación :	
Código:PA1	Historia de Usuario: Registro cargo
Nombre: Despliegue del mensaje “Inserción Exitosa!” en la clase Presentación.	
Responsable: Elvira Yánez	Fecha: 12/11/2013
Descripción: Verificar el despliegue del mensaje “Inserción Exitosa! “ del método insertarcargos implementado en la clase ADatos.	
Condiciones de Ejecución: Insertar un cargo.	
Pasos de ejecución: 1. Ingresar los datos de un cargo 2. Clic en el botón enviar. 3. Se visualiza el mensaje “Inserción Exitosa! en caso de no existir el cliente en la base de datos.	
Resultado esperado: Presentación del mensaje de error “Inserción Exitosa!	
Evaluación de la prueba: Satisfactoria.	

Prueba de Aceptación :	
Código: PA2	Historia de Usuario: Registro cargo
Nombre: Despliegue del mensaje “Inserción Fallida!” en la clase Presentación.	
Responsable: Elvira Yánez	Fecha: 12/11/2013
Descripción: Verificar el Despliegue del mensaje “Inserción Fallida “del método insertarcargo implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un cargo.	

Pasos de ejecución: 1. Ingresar los datos de un cargo. 2. Clic en el botón enviar. 3. Se visualiza el mensaje “Inserción Fallida” en caso de existir un cargo en la base de datos.
Resultado esperado: Presentación del mensaje de error “Inserción Fallida”
Evaluación de la prueba: Satisfactoria.

TARJETAS CRC

Cargo	
Responsabilidades	Colaboración
Cargo() Set() Get() id nombres; estado empresa	

ADatos	
Responsabilidades	Colaboración
insertarcargo ()	AccesoDatos Empresa

REGISTRO PRUEBA

Historia de Usuario		
Número: 6	Nombre de la historia: Registro prueba	
Modificación de historia de usuario:		
Usuario: Usuario GRUPO LAAR		Iteración Asignada: 2
Prioridad en el Negocio: Alta		Puntos Estimados:9
Riesgo en el Desarrollo: Bajo		Puntos Reales:10
Descripción: El sistema permitirá el registro de las pruebas para elaborar la evaluación de los usuarios aspirantes.		
Observaciones: Se registrara la siguiente información de cada prueba: id, nombre, valoración, estado; El cargo debe existir para que se cree la prueba		
Pruebas de Aceptación: Al ingresar los datos erróneo se emitirá un mensaje de confirmación que los datos Inserción fallida Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.		

TAREA DE INGENIERÍA.

Tarea de Ingeniería		
Historia de Usuario: Registro prueba		
Número de Tarea: 1	Nombre de Tarea: Creación de los métodos necesarios en clase ADatos de la capa de Acceso_Datos y la creación de la clase Prueba	
Tipo de Tarea: Desarrollo.		Puntos Estimados: 3
Fecha Inicio:16/09/2013		Fecha Fin:18/09/2013
Programador Responsable: Elvira Yánez		
Descripción: Crearemos el método web insertarprueba en la clase ADatos en la capa de Acceso_Datos		
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una prueba con datos correctos.		

Tarea de Ingeniería	
Historia de Usuario: Registro de prueba	
Número de Tarea: 2	Nombre de Tarea: Creación del método insertarprueba en clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Fecha Inicio:19/09/2013	Fecha Fin:23/09/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarprueba en la clase LNegocio en la capa de Logica_Negocio	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallid ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una prueba con datos correctos.	
Tarea de Ingeniería	
Historia de Usuario: Registro prueba	
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web en la capa Presentación, necesarias para ingresar una prueba
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Fecha Inicio:24/09/2013	Fecha Fin:26/09/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación de las paginas IngresoPrueba.php en la capa Presentación, las mismas que permitirán capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una prueba con datos correctos.	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:2	Historia de Usuario: Registro de prueba
Nombre: Despliegue del mensaje “Inserción Exitosa” al ingresar un proveedor con datos correctos.	
Responsable: Elvira Yánez	Fecha: 27/11/2013
Descripción: Verificar el despliegue del mensaje “Inserción Exitosa” del método insertarprueba implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un prueba	
Pasos de ejecución: 1. Ingresar los datos de una prueba. 2. Clic en el botón enviar. 3. Se visualiza el mensaje “Inserción Exitosa!”.	
Resultado esperado: Emitir el mensaje “Inserción Exitosa!”.	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:3	Historia de Usuario: Registro de prueba
Nombre: Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 27/11/2013
Descripción: Verificar el despliegue del mensaje "Inserción fallida” del método insertarprueba implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar una prueba siempre que exista un cargo	
Pasos de ejecución: Ingresar los datos de una prueba. Clic en el botón enviar. Se visualiza el mensaje “Inserción fallida” ! la prueba ya exista o los campos estén incompletos	
Resultado esperado: Emitir el mensaje de	allida”
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: AccesoDatos	
Métodos: AccesoDatos() get() set() Conectar() Desconectar()	Colaboradores:

Nombre de la Clase: ADatos	
Responsabilidades: Atributos: id, nombre, valoración, estado. Métodos: insertapruebar () Set() Get()	Colaboradores: AccesoDatos

Prueba	
Responsabilidad	Colaboración
id, nombre, valoración, estado. Set() Get()	

REGISTRO PREGUNTA

Historia de Usuario	
Número: 7	Nombre de la historia: Registro pregunta
Modificación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 2
Prioridad en el Negocio: Alta	Puntos Estimados:9
Riesgo en el Desarrollo: Bajo	Puntos Reales:11
Descripción: El sistema permitirá el registro de las preguntas por cada prueba para elaborar la evaluación de los usuarios aspirantes.	
Observaciones: Se registrara la siguiente información de cada pregunta: id, nombre, , estado; la prueba debe existir para que se cree la pregunta	
Pruebas de Aceptación: Al ingresar los datos erróneo se emitirá un mensaje de confirmación que los datos Inserción fallida Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Registro pregunta	
Número de Tarea: 1	Nombre de Tarea: Creación de los métodos necesarios en clase ADatos de la capa de Acceso_Datos y la creación de la clase Pregunta
Tipo de Tarea: Desarrollo.	Puntos Estimados: 3
Fecha Inicio:27/09/2013	Fecha Fin:01/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarpregunta en la clase ADatos en la capa de Acceso_Datos	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una prueba con datos correctos.	

Tarea de Ingeniería	
Historia de Usuario: Registro de pregunta	
Número de Tarea: 2	Nombre de Tarea: Creación del método insertarpregunta en clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Fecha Inicio:02/10/2013	Fecha Fin:04/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarpregunta en la clase LNegocio en la capa de Logica_Negocio	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallid ır ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje "Inserción Exitc sar una prueba con datos correctos.	

Tarea de Ingeniería	
Historia de Usuario: Registro pregunta	
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web en la capa Presentación, necesarias para ingresar una pregunta
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Fecha Inicio:07/10/2013	Fecha Fin:09/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación de las paginas IngresoPregunta.php en la capa Presentación, las mismas que permitirán capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje "Inserción Exitosa" al ingresar una prueba con datos correctos.	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Registro de pregunta
Nombre: Despliegue del mensaje “Inserción Exitosa” al ingresar un proveedor con datos correctos.	
Responsable: Elvira Yánez	Fecha: 28/11/2013
Descripción: Verificar el despliegue del mensaje “Inserción Exitosa” del método insertarpregunta implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un pregunta si en la base de datos existe una prueba registrada	
Pasos de ejecución: Ingresar los datos de una pregunta Clic en el botón enviar. Se visualiza el mensaje “Inserción Exitosa!”.	
Resultado esperado: Emitir el mensaje “Inserción Exitosa!”.	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: Registro de pregunta
Nombre: Despliegue del mensaje "Inserción fallida" al intentar ingresar una pregunta con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 28/11/2013
Descripción: Verificar el despliegue del mensaje "Inserción fallida” del método insertarpregunta implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar una prueba siempre que exista un prueba	
Pasos de ejecución: Ingresar los datos de una pregunta. Clic en el botón enviar. Se visualiza el mensaje “Inserción fallida” ! la pregunta ya exista o los campos estén incompletos	
Resultado esperado: Emitir el mensaje de "Inserción fallida”	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: AccesoDatos	
Métodos: AccesoDatos() get() set() Conectar()	Colaboradores:

Nombre de la Clase: ADatos	
Responsabilidades: Atributos: id, nombre, valoración, estado. Métodos: insertapruebar () Set() Get()	Colaboradores: AccesoDatos

REGISTRO RESPUESTAS

Historia de Usuario	
Número: 8	Nombre de la historia: Registro respuestas
Modificación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 2
Prioridad en el Negocio: Alta	Puntos Estimados:9
Riesgo en el Desarrollo: Bajo	Puntos Reales:10
Descripción: El sistema permitirá el registro de las respuestas para elaborar la evaluación de los usuarios aspirantes.	
Observaciones: Se registrara la siguiente información de cada prueba: id, nombre, valoración, estado; El cargo debe existir para que se cree la prueba	
Pruebas de Aceptación: Al ingresar los datos erróneo se emitirá un mensaje de confirmación que los datos Inserción fallida Al ingresar correctamente los datos se emitirá un mensaje de confirmación que los datos se han guardado correctamente.	

TAREA DE INGENIERÍA

.

Tarea de Ingeniería	
Historia de Usuario: Registro respue[illegible]	
Número de Tarea: 1	Nombre de [illegible]ción de los métodos necesarios en clase ADatos de la capa de Acceso_Datos y la creación de la clase Respuesta
Tipo de Tarea: Desarrollo.	Puntos Estimados: 3
Fecha Inicio:10/10/2013	Fecha Fin:14/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarrespuesta en la clase ADatos en la capa de Acceso_Datos	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una respuesta con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una respuesta con datos correctos.	

Tarea de Ingeniería	
Historia de Usuario: Registro de respuesta	
Número de Tarea: 2	Nombre de Tarea: Creación del método insertarrespuesta en clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Fecha Inicio:15/10/2013	Fecha Fin:17/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web insertarrespuesta en la clase LNegocio en la capa de Logica_Negocio	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una respuesta con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una respuesta con datos correctos.	

Tarea de Ingeniería	
Historia de Usuario: Registro respues	
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web en la capa Presentación, necesarias para ingresar una respuesta
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Fecha Inicio:18/10/2013	Fecha Fin:22/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación de las paginas IngresoRespuesta.php en la capa Presentación, las mismas que permitirán capturar, procesar, presentar datos, como también consumir el servicio web de la capa Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una respuesta con datos incompletos o erróneos. Despliegue del mensaje “Inserción Exitosa” al ingresar una respuesta con datos correctos.	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Registro de respuesta
Nombre: Despliegue del mensaje “Inserción Exitosa” al ingresar un proveedor con datos correctos.	
Responsable: Elvira Yánez	Fecha: 30/11/2013
Descripción: Verificar el despliegue del mensaje “Inserción Exitosa” del método insertarrespuesta implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar un respuesta si existe una pregunta	
Pasos de ejecución: Ingresar los datos de una respuesta. Clic en el botón enviar. Se visualiza el mensaje “Inserción Exitosa!”.	
Resultado esperado: Emitir el mensaje “Inserción Exitosa!”.	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: Registro de respuesta
Nombre: Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 30/11/2013
Descripción: Verificar el despliegue del mensaje "Inserción fallida” del método insertarrespuesta implementado en la clase ADatos .	
Condiciones de Ejecución: Insertar una respuesta siempre que exista una pregunta y su valor no exceda del valor de la prueba	
Pasos de ejecución: Ingresar los datos de una prueba. Clic en el botón enviar. Se visualiza el mensaje “Inserción fallida” si una respuesta a ya exista o los campos estén incompletos	
Resultado esperado: Emitir el mensaje de "Inserción fallida”	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: AccesoDatos	
Métodos: AccesoDatos() get() set() Conectar()	Colaboradores:

Nombre de la Clase: ADatos	
Responsabilidades: Atributos: id, nombre, valor, estado. Métodos: insertarrespuesta () Set() Get()	Colaboradores: AccesoDatos

Nombre de la Clase:Respuesta	
Responsabilidad	Colaboración
id, nombre, valor, estado. Set() G	

MODIFICAR USURARIO GRUPO LAAR

Historia de Usuario	
Número: 9	Nombre de la historia: Modificar Usurario GRUPO LAAR
Creación de historia de usuario:	
Usuario: Administrador	Iteración Asignada: 3
Prioridad en el Negocio: Bajo	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: El sistema permitirá modificar Usuario GRUPO LAAR.	
Observaciones: Se modificara la siguiente información de cada Usuario GRUPO LAAR de la que se puede alterar nombre, empresa, estado, tipo usuario.	
Pruebas de Aceptación: Despliegue del mensaje "Modificación del Usuario fallida" al intentar modificar un Usuario GRUPO LAAR. con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa"	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Modificar de Usuario GRUPO LAAR.	
Número de Tarea: 1	Nombre de Tarea: Creación del método modificarusuario en clase ADatos de la capa de Acceso_Datos
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Inicio: miércoles 23/10/2013	Fecha Fin: 24/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web modificausu ase ADatos en la capa de Acceso_Datos.	
Pruebas de Aceptación. Despliegue del mensaje "Modificación fallida" al intentar modificar un usuario con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa!" .	

Tarea de Ingeniería	
Historia de Usuario: Modificar de Usuario GRUPO LAAR.	
Número de Tarea: 2	Nombre de Tarea: Creación del método modificarusuario en clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Modificación	Puntos Estimados: 2
Fecha Inicio: Miércoles 25/10/2013	Fecha Fin: 28/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación del método modificausuario en clase LNegocio de la capa de Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar un usuario con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"	

Tarea de Ingeniería	
Historia de Usuario: Modificar Usuario GRUPO LAAR	
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web ModificaUsuario en la capa Presentación,
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Fin: 29/10/2013	Fecha Fin: 30/10/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación de las páginas web ModificaUsu la capa Presentación	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar un usuario con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: : Modificar un Usuario GRUPO LAAR
Nombre: Se despliegue del mensaje "Modificación fallida " al intentar modificar un usuario con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 11/12/2013
Descripción: Despliegue del mensaje "Modificación fallida" al intentar modificar un usuario con datos incompletos o erróneos.	
Condiciones de Ejecución: Modificar un usuario siempre que este exista en la base de daros	
Pasos de ejecución: 1. Ingresar los datos de un usuario. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje "Modificación fallida"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: : Modificar un usuario GRUPO LAAR
Nombre: Despliegue del mensaje "Modificación exitosa!" .	
Responsable: Elvira Yánez	Fecha: 11/12/2013
Descripción Despliegue del mensaje "Modificación exitosa" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Modificar un usuario correctamente	
Pasos de ejecución: 1. Ingresar los datos de un usuario. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: "Modificación exit	
Evaluación de la prueba: satisfactorio	

TARJETA CRC-

Nombre de la Clase: Modificar proveedor	
Responsabilidad	Colaboración
modificarusuario()	AccesoDatos() Usuario()

Nombre de la Clase: Logica_Negocio	
Responsabilidad	Colaboración
modificaproveedor()	Usuario()

MODIFICAR EMPRESA

Historia de Usuario		
Número: 10	Nombre de la historia: Modificar empresa	
Creación de historia de usuario:		
Usuario: Administrador		Iteración Asignada: 3
Prioridad en el Negocio: Bajo		Puntos Estimados:6
Riesgo en el Desarrollo: Medio		Puntos Reales:8
Descripción: El sistema permitirá modificar una empresa determinada		
Observaciones: Se modificara la siguiente información de cada empresa de la que se puede alterar nombre, descripción, estado,		

Pruebas de Aceptación: Despliegue del mensaje "Modificación fallida" al intentar modificar una empresa con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa"

TAREA DE INGENIERÍA.

Tarea de Ingeniería		
Historia de Usuario: Modificar Empresa.		
Número de Tarea: 1	Nombre de Tarea: Creación del método modificarempresa en clase ADatos de la capa de Acceso_Datos	
Tipo de Tarea: Modificación.		Puntos Estimados: 2
Fecha Inicio: miércoles 31/10/2013		Fecha Fin: 01/11/2013
Programador Responsable: Elvira Yánez		
Descripción: Crearemos el método web modificarempresa en la clase ADatos en la capa de Acceso_Datos.		
Pruebas de Aceptación. Despliegue del mensaje "Modificación fallida" al intentar modificar una empresa con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa!" .		

Tarea de Ingeniería		
Historia de Usuario: Modificar Empresa		
Número de Tarea: 2	Nombre de Tarea: Creación del método modificarempresa en clase LNegocio de la capa de Logica_Negocio.	
Tipo de Tarea: Modificación		Puntos Estimados: 2
Fecha Inicio: Miércoles 04/11/2013		Fecha Fin: 05/11/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación del método modificarempresa en clase LNegocio de la capa de Logica_Negocio.		

Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una empresa con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"

Tarea de Ingeniería		
Historia de Usuario: Modificar Empresa		
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web ModificaEmpresa en la capa Presentación,	
Tipo de Tarea: Modificación.		Puntos Estimados: 2
Fecha Fin: 06/11/2013		Fecha Fin: 06/11/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación de las páginas web ModificaEı en la capa Presentación		
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una empresa con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"		

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: : Modificar empresa
Nombre: Se despliegue del mensaje "Modificación fallida " al intentar modificar una empresa con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 11/12/2013
Descripción: Despliegue del mensaje "Modificación fallida" al intentar modificar una empresa con datos incompletos o erróneos.	
Condiciones de Ejecución: Modificar una empresa siempre que este exista en la base de daros	
Pasos de ejecución: 1. Ingresar los datos de una empresa. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje "Modificación fallida"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: : Modificar una empresa
Nombre: Despliegue del mensaje "Modificación exitosa!" .	
Responsable: Elvira Yánez	Fecha: 11/12/2013
Descripción se despliega del mensaje "Modificación exitosa" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Modificar una empresa correctamente	
Pasos de ejecución: 1. Ingresar los datos de una empresa. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: "Modificación exitosa	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Modificar proveedor	
Responsabilidad	Colaboración
modificarempresa()	AccesoDatos() Empresa()

Nombre de la Clase: Logica Negocio	
Responsabilidad	Colaboración
modificarempresa()	Empresa()

MODIFICAR CARGO

Historia de Usuario	
Número: 11	Nombre de la historia: Modificar cargo
Creación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 3
Prioridad en el Negocio: Bajo	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: El sistema permitirá modificar un cargo determinada	
Observaciones: Se modificara la siguiente información de cada empresa de la que se puede alterar nombre, empresa, estado,	
Pruebas de Aceptación: Despliegue del mensaje "Modificación fallida" al intentar modificar un cargo con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa"	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Modificar cargo.	
Número de Tarea: 1	Nombre de Tarea: Creación del método modificarcargo en clase ADatos de la capa de Acceso_Datos
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Inicio: miércoles 13/11/2013	Fecha Fin: 14/11/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web modificarcargo en la clase ADatos en la capa de Acceso_Datos.	
Pruebas de Aceptación. Despliegue del mensaje "Modificación fallida" al intentar modificar un cargo con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa!" .	

Tarea de Ingeniería

Historia de Usuario: Modificar cargo		
Número de Tarea: 2	Nombre de Tarea: Creación del método modificarcargo en clase LNegocio de la capa de Logica_Negocio.	
Tipo de Tarea: Modificación		Puntos Estimados: 2
Fecha Inicio: Miércoles 15/11/2013		Fecha Fin: 18/11/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación del método modificarcargo en clase LNegocio de la capa de Logica_Negocio.		
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar un cargo con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"		

Tarea de Ingeniería		
Historia de Usuario: Modificar cargo		
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web Modificar Cargo en la capa Presentación,	
Tipo de Tarea: Modificación.		Puntos Estimados: 2
Fecha Fin: 19/11/2013		Fecha Fin: 22/11/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación de las páginas web ModificaCa a capa Presentación		
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar un cargo con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"		

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: : Modificar cargo
Nombre: Se despliegue del mensaje "Modificación fallida " al intentar modificar un cargo con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 12/12/2013
Descripción: Despliegue del mensaje "Modificación fallida" al intentar modificar un cargo con datos incompletos o erróneos.	
Condiciones de Ejecución: Modificar un cargo siempre que este exista en la base de daros	
Pasos de ejecución: 1. Ingresar los datos de un cargo. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje "Modificación fallida"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: : Modificar cargo
Nombre: Despliegue del mensaje "Modificación exitosa".	
Responsable: Elvira Yánez	Fecha: 11/12/2013
Descripción se despliega del mensaje "Modificación exitosa" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Modificar un cargo correctamente	
Pasos de ejecución: 1. Ingresar los datos de un cargo. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: "Modificación exito	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Modificar proveedor	
Responsabilidad	Colaboración
modificarcargo()	AccesoDatos() Cargo()

Nombre de la Clase: Logica Negocio	
Responsabilidad	Colaboración
modificarcargo()	Cargo()

MODIFICAR PRUEBA

Historia de Usuario	
Número: 12	Nombre de la historia: Modificar prueba
Creación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 3
Prioridad en el Negocio: Bajo	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: El sistema permitirá modificar una prueba determinada	
Observaciones: Se modificara la siguiente información de cada prueba de la que se puede alterar nombre, valoración, estado,	
Pruebas de Aceptación: Despliegue del mensaje "Modificación fallida" al intentar modificar una prueba con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa"	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Modificar prueba.	
Número de Tarea: 1	Nombre de Tarea: Creación del método modificarprueba en clase ADatos de la capa de Acceso_Datos
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Inicio: miércoles 25/11/2013	Fecha Fin: 26/11/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web modificarprueba en la clase ADatos en la capa de Acceso_Datos.	
Pruebas de Aceptación. Despliegue del mensaje "Modificación f tentar modificar una prueba con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa!" .	
Tarea de Ingeniería	
Historia de Usuario: Modificar prueba	

Número de Tarea: 2	Nombre de Tarea: Creación del método modificarprueba en clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Modificación	Puntos Estimados: 2
Fecha Inicio: Miércoles 27/11/2013	Fecha Fin: 28/11/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación del método modificarprueba en clase LNegocio de la capa de Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"	

Tarea de Ingeniería	
Historia de Usuario: Modificar prueba	
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web ModificarPrueba en la capa Presentación,
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Fin: 29/11/2013	Fecha Fin: 02/12/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación de las páginas web ModificaPrueba.php en la capa Presentación	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una prueba con datos incompletos o erróneos. Despliegue del mensaje "Inserción exito	

PRUEBAS DE ACEPTACIÓN

Prueba de Aceptación:	
Código:1	Historia de Usuario: : Modificar prueba
Nombre: Se despliegue del mensaje "Modificación fallida " al intentar modificar una prueba con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 19/12/2013
Descripción: Despliegue del mensaje "Modificación fallida" al intentar modificar una prueba con datos incompletos o erróneos.	
Condiciones de Ejecución: Modificar una prueba siempre que exista en la base de daros	
Pasos de ejecución: 1. Ingresar los datos de una prueba. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje "Modificación fallida"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: : Modificar prueba
Nombre: Despliegue del mensaje "Modificación exitosa".	
Responsable: Elvira Yánez	Fecha: 19/12/2013
Descripción se despliega del mensaje "Modificación exitosa" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Modificar una prueba correctamente	
Pasos de ejecución: 1. Ingresar los datos de una prueba. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: "Modificación exito	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Modificar proveedor	
Responsabilidad	Colaboración
modificarprueba()	AccesoDatos() Prueba()

Logica Negocio	
Responsabilidad	Colaboración
modificarprueba()	Prueba()

MODIFICAR PREGUNTA

Historia de Usuario	
Número: 13	Nombre de la historia: Modificar pregunta
Creación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 3
Prioridad en el Negocio: Bajo	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: El sistema permitirá modificar una pregunta determinada	
Observaciones: Se modificara la siguiente información de cada pregunta de la que se puede alterar nombre, estado,	
Pruebas de Aceptación: Despliegue del mensaje "Modificación fallida" al intentar modificar una pregunta con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa"	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Modificar pregunta.	
Número de Tarea: 1	Nombre de Tarea: Creación del método modificarpregunta en clase ADatos de la capa de Acceso_Datos
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Inicio: miércoles 03/12/2013	Fecha Fin: 04/12/2013
Programador Responsable: Elvira Yánez	
Descripción: Crearemos el método web modificarpregunta en la clase ADatos en la capa de Acceso_Datos.	
Pruebas de Aceptación. Despliegue del mensaje "Modificación fallida" al intentar modificar una pregunta con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa!" .	

Tarea de Ingeniería	
Historia de Usuario: Modificar pregunta	
Número de Tarea: 2	Nombre de Tarea: Creación del método modificarpregunta en clase LNegocio de la capa de Logica_Negocio.
Tipo de Tarea: Modificación	Puntos Estimados: 2
Fecha Inicio: Miércoles 05/11/2013	Fecha Fin: 06/11/2013
Programador Responsable: Elvira Yánez	
Descripción: Creación del método modificarpregunta en clase LNegocio de la capa de Logica_Negocio.	
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallid r ingresar una pregunta con datos incompletos o erróneos. Despliegue del mensaje "Inserción exito	

Tarea de Ingeniería		
Historia de Usuario: Modificar pregunta		
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web ModificarPregunta en la capa Presentación,	
Tipo de Tarea: Modificación.		Puntos Estimados: 2
Fecha Fin: 09/12/2013		Fecha Fin: 10/12/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación de las páginas web ModificaPregunta.php en la capa Presentación		
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una pregunta con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitos		

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:		
Código:1	Historia de Usuario: : Modificar pregunta	
Nombre: Se despliegue del mensaje "Modificación fallida " al intentar modificar una pregunta con datos incompletos o erróneos.		
Responsable: Elvira Yánez		Fecha: 19/12/2013
Descripción: Despliegue del mensaje "Modificación fallida" al intentar modificar una pregunta con datos incompletos o erróneos.		
Condiciones de Ejecución: Modificar una pregunta siempre que exista en la base de daros		
Pasos de ejecución: 1. Ingresar los datos de una pregunta. 2. Clic en el botón enviar. 3. Se visualiza el mensaje		
Resultado esperado: Emitir el mensaje "Modificación fallida"		
Evaluación de la prueba: satisfactorio		

Prueba de Aceptación:	
Código:2	Historia de Usuario: : Modificar pregunta
Nombre: Despliegue del mensaje "Modificación exitosa".	
Responsable: Elvira Yánez	Fecha: 19/12/2013
Descripción se despliega del mensaje "Modificación exitosa" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Modificar una pregunta correctamente	
Pasos de ejecución: 1. Ingresar los datos de una pregunta. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: "Modificación exitosa"	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Modificar proveedor	
Responsabilidad	Colaboración
modificarpregunta()	AccesoDatos() Pregunta()

Logica Negocio	
Responsabilidad	Colaboración
modificarpregunta()	Pregunta()

MODIFICAR RESPUESTA

Historia de Usuario	
Número: 14	Nombre de la historia: Modificar respuesta
Creación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 3
Prioridad en el Negocio: Bajo	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: El sistema permitirá modificar una respuesta determinada	
Observaciones: Se modificara la siguiente información de cada respuesta de la que se puede alterar nombre, valor, estado,	
Pruebas de Aceptación: Despliegue del mensaje "Modificación fallida" al intentar modificar una respuesta con datos incompletos o erróneos. Despliegue del mensaje "Modificación exitosa"	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Modificar respuesta.	
Número de Tarea: 1	Nombre de Tarea: Creación del método modificarrespuesta en clase ADatos de la capa de Acceso_Datos
Tipo de Tarea: Modificación.	Puntos Estimados: 2
Fecha Inicio: miércoles 11/12/2013	Fecha Fin: 12/12/2013
Programador Responsable: Elvira Yáne	
Descripción: Crearemos el método web modificarres clase ADatos en la capa de Acceso_Datos.	
Pruebas de Aceptación. Despliegue del mensaje "Modificación fallida" al intentar modificar una respuesta con datos incompletos o erróneos.	

Despliegue del mensaje "Modificación exitosa!" .

Tarea de Ingeniería		
Historia de Usuario: Modificar pregunta		
Número de Tarea: 2	Nombre de Tarea: Creación del método modificarrespuesta en clase LNegocio de la capa de Logica_Negocio.	
Tipo de Tarea: Modificación		Puntos Estimados: 2
Fecha Inicio: Miércoles 13/11/2013		Fecha Fin: 16/11/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación del método modificarrespuesta en clase LNegocio de la capa de Logica_Negocio.		
Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una respuesta con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"		

Tarea de Ingeniería		
Historia de Usuario: Modificar respuesta		
Número de Tarea: 3	Nombre de Tarea: Creación de las páginas web ModificarRespuesta en la capa Presentación,	
Tipo de Tarea: Modificación.		Puntos Estimados: 2
Fecha Fin: 17/12/2013		Fecha Fin: 18/12/2013
Programador Responsable: Elvira Yánez		
Descripción: Creación de las páginas web ModificaRespuesta.php en la capa Presentación		

Pruebas de Aceptación. Despliegue del mensaje "Inserción fallida" al intentar ingresar una respuesta con datos incompletos o erróneos. Despliegue del mensaje "Inserción exitosa"

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Modificar respuesta
Nombre: Se despliegue del mensaje "Modificación fallida " al intentar modificar una respuesta con datos incompletos o erróneos.	
Responsable: Elvira Yánez	Fecha: 20/12/2013
Descripción: Despliegue del mensaje "Modificación fallida" al intentar modificar una respuesta con datos incompletos o erróneos.	
Condiciones de Ejecución: Modificar una respuesta siempre que exista en la base de daros	
Pasos de ejecución: 1. Ingresar los datos de una respuesta. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje "Modificación fallida"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: Modificar respuesta
Nombre: Despliegue del mensaje "Modificación exitosa".	
Responsable: Elvira Yánez	Fecha: 19/12/2013
Descripción se despliega del mensaje "Modificación exitosa" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Modificar una respuesta correctamente	
Pasos de ejecución: 1. Ingresar los datos de una respuesta. 2. Clic en el botón enviar. 3. Se visualiza el mensaje	
Resultado esperado: "Modificación exitosa"	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Modificar proveedor	
Responsabilidad	Colaboración
modificarrespuesta()	AccesoDatos() Respuesta()

Logica Negocio	
Responsabilidad	Colaboración
modificarrespuesta()	Respuesta()

EVALUACION ASPIRANTE

Historia de Usuario	
Número: 15	Nombre de la historia: Evaluación aspirante
Creación de historia de usuario:	
Usuario: Aspirante	Iteración Asignada: 4
Prioridad en el Negocio: Bajo	Puntos Estimados:10
Riesgo en el Desarrollo: Medio	Puntos Reales:14
Descripción: El sistema permite al aspirante rendir la evaluación sobre el cargo vacante	
Observaciones: Se debe tener registrado pruebas, preguntas y respuesta de un determinado cargo, las evaluaciones que rinde el aspirante pueden ser más de una, cada una posee un tiempo determinado de 30s	
Pruebas de Aceptación: Despliegue del mensaje "Intento de Copia" al presionar clic derecho, prtsc o al seleccionar el texto. Despliegue del mensaje “Almacenamiento Exitoso" Despliegue del mensaje “Tiempo Agotado"	

Tarea de Ingeniería		
Historia de Usuario: Evaluación aspirante		
Número de Tarea: 1	Nombre de Tarea: Creación de las páginas web Evaluación en la capa Presentación,	
Tipo de Tarea: Desarrollo		Puntos Estimados: 10
Fecha Fin: 26/12/2013		Fecha Fin: 08/01/2014
Programador Responsable: Elvira Yánez		
Descripción: Creación de las páginas web Evaluación.php en la capa Presentación		

Pruebas de Aceptación. Despliegue del mensaje "Intento de Copia" al presionar clic derecho, prtsc o al seleccionar el texto. Despliegue del mensaje “Almacenamiento Exitoso" Despliegue del mensaje “Tiempo Agotado"

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Evaluación al aspirante
Nombre: Se despliegue del mensaje "Intento de Copia" al intentar realizar fraude al momento de rendir la evaluación.	
Responsable: Elvira Yánez	Fecha: 10/01/2014
Descripción: Se despliegue del mensaje "Intento de Copia " al intentar realizar fraude al momento de rendir la evaluación	
Condiciones de Ejecución: Siempre que se intente realizar algún tipo de copia	
Pasos de ejecución: 1. Ingresar a la Evaluación. 2. Clic en el botón iniciar. 3. Presionar clic derecho, prtsc o seleccionar texto 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje "Intento de Copia"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: Evaluación Aspirante
Nombre: Despliegue del mensaje "Modificación exitosa".	
Responsable: Elvira Yánez	Fecha: 10/01/2014
Descripción se despliega del mensaje "Almacenamiento exitoso" este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: Rendir la evaluación correctamente	
Pasos de ejecución: 1. Ingresar a la Evaluación 2. Clic en el botón iniciar 3. Contestar preguntas. 4. Guardar Evaluación 5. Se visualiza el mensaje	
Resultado esperado: " Almacenamiento exitoso"	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:3	Historia de Usuario: Evaluación Aspirante
Nombre: Despliegue del mensaje "Tiempo Agotado".	
Responsable: Elvira Yánez	Fecha: 10/01/2014
Descripción se despliega del mensaje "Tiempo Agotado" este mensaje aparece cuando no el tiempo de la evaluación se haya terminado	
Condiciones de Ejecución: El tiempo para Rendir la evaluación se termine	
Pasos de ejecución: 1. Ingresar a la Evaluación 2. Clic en el botón iniciar 3. Contestar preguntas. 4. Se visualiza el mensaje	
Resultado esperado: " Tiempo Agotado"	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la Clase: Evaluación	
Responsabilidad	Colaboración
Evaluación()	AccesoDatos() Prueba() Pregunta() Respuesta()

RESPUESTA ASPIRANTE

Historia de Usuario	
Número: 16	Nombre de la historia: Respuesta aspirante
Creación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 4
Prioridad en el Negocio: Bajo	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:5
Descripción: El sistema permite al Usuario GRUPO LAAR emitir una respuesta al aspirante sobre el resultado obtenido al rendir la evaluación sobre el cargo vacante	
Observaciones: El aspirante debe haber rendido al menos una evaluación.	
Pruebas de Aceptación: Despliegue del mensaje "Fallo al Almacenar respuesta" si no intenta almacenar información incorrecta o sin llenar información Despliegue del mensaje “Respuesta almacenada"	

TAREA DE INGENIERÍA

Tarea de Ingeniería
Historia de Usuario: Respuesta aspirante

Número de Tarea: 1	Nombre de Tarea: Creación de las páginas web Respuesta en la capa Presentación,
Tipo de Tarea: Desarrollo.	Puntos Estimados: 10
Fecha Fin: 09/01/2014	Fecha Fin: 16/01/2014
Programador Responsable: Elvira Yánez	
Descripción: Creación de las páginas web Respuesta.php en la capa Presentación	
Pruebas de Aceptación. Despliegue del mensaje "Fallo al Almacenar respuesta" si no intenta almacenar información incorrecta o al dejar campos en blanco Despliegue del mensaje “Respuesta almacenada"	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación:	
Código:1	Historia de Usuario: Respuesta al aspirante
Nombre: Se despliegue del mensaje " Fallo al Almacenar respuesta "	
Responsable: Elvira Yánez	Fecha: 12/01/2014
Descripción: Se despliegue del mensaje " Fallo al Almacenar respuesta " al no almacenar la información de manera correcta	
Condiciones de Ejecución: Si no se escribe la respuesta o se deja campos en blanco	
Pasos de ejecución: 1. Ingresar a la aspirante 2. Seleccionar un determinado aspirante 3. Dar respuesta 3. Se visualiza el mensaje	
Resultado esperado: Emitir el mensaje " Fallo al Almacenar respuesta "	
Evaluación de la prueba: satisfactorio	

Prueba de Aceptación:	
Código:2	Historia de Usuario: Respuesta aspirante
Nombre: Despliegue del mensaje " Respuesta almacenada".	
Responsable: Elvira Yánez	Fecha: 12/01/2013
Descripción se despliega del mensaje " Respuesta almacenada " este mensaje aparece cuando no hay ningún conflicto.	
Condiciones de Ejecución: dar respuesta al aspirante correctamente	
Pasos de ejecución: 1. Ingresar aspirante 2. Seleccionar un determinado aspirante. 3. Dar una respuesta 4.Guardar respuesta 5. Se visualiza el mensaje	
Resultado esperado: " Respuesta almacenada "	
Evaluación de la prueba: satisfactorio	

TARJETA CRC

Nombre de la clase: Respuesta	
Responsabilidad	Colaboración
Respuesta()	AccesoDatos() Evaluación() Aspirante()

REPORTE DE ASPIRANTES

Historia de Usuario	
Número: 17	Nombre de la historia: Reporte de Aspirantes.
Modificación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 4
Prioridad en el Negocio: Alta	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: Como usuario quiero visualizar los aspirantes que han rendido determinada evaluación.	
Observaciones: En el menú de la aplicación un botón permitirá mostrar los resultados deseados	
De no existir la información solicitado se emitirá un mensaje de “La información no existe” Mostrar la información solicitada	

TAREA DE INGENIERÍA.

Tarea de Ingeniería	
Historia de Usuario: Reporte de Aspirantes.	
Número Tarea: 1	Nombre de la Tarea: Creación de las páginas web en la capa Presentación, necesarias para realizar el reporte
Tipo de Tarea: Desarrollo	Puntos de Estimados: 6
Fecha de Inicio: 17/01/2014	Fecha de Fin: 24/01/2014
Programador Responsable: Elvira Yánez	
Descripción: Se implementara ReporteAspirate.php en la capa de Presentación.	
Pruebas de Aceptación: ,Mostrar la información solicitada	

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación	
Código: 1	Historia de Usuario: Reporte aspirante.
Nombre: Devolver un mensaje en caso que no exista aspirante	
Responsable: Elvira Yánez	Fecha:06/11/2013
Descripción: Devolver el mensaje “La información no existe”	
Condiciones de Ejecución: En la base de datos no debe existir datos	
Pasos de Ejecución: 1.Seleccionar el Reporte	
Resultado Esperado: - “La información no existe”	
Evaluación de la prueba: Satisfactorio	

Prueba de Aceptación	
Código: 2	Historia de Usuario: Reporte Aspirante.
Nombre: Mostrar la información solicitada.	
Responsable: Elvira Yánez	Fecha:12/11/2013
Descripción: al seleccionar el reporte debe generarse con la información solicitada.	
Condiciones de Ejecución: Debe de estar llena la base de datos para que devuelva datos, además debe existir una conexión con acceso a datos	
Pasos de Ejecución: 1.Seleccionar el Reporte	
Resultado Esperado: - Mostrar la información	
Evaluación de la prueba: satisfactorio	

TARJETA CRC.

Nombre de la Clase: ReporteAspirante	
Responsabilidad	Colaboración
Cedula nombre apellido edad estado, calificación	Evaluación() Aspirante()

REPORTE DE ASPIRANTES EVALUACIÓN

Historia de Usuario	
Número: 18	Nombre de la historia: Reporte de aspirante evaluación
Modificación de historia de usuario:	
Usuario: Usuario GRUPO LAAR	Iteración Asignada: 4
Prioridad en el Negocio: Alta	Puntos Estimados:6
Riesgo en el Desarrollo: Medio	Puntos Reales:8
Descripción: Como usuario quiero visualizar los aspirantes que han rendido determinada evaluación.	
Observaciones: En el menú de la aplicación un botón permitirá mostrar los resultados deseados	
De no existir la información solicitado se emitirá un mensaje de “La información no existe” Mostrar la información solicitada	

TAREA DE INGENIERÍA.

Tarea de Ingeniería		
Historia de Usuario: Reporte de aspirantes. evaluación		
Número Tarea: 1	Nombre de la Tarea: Creación de las páginas web en la capa Presentación, necesarias para realizar el reporte	
Tipo de Tarea: Desarrollo		Puntos de Estimados: 6
Fecha de Inicio: 27/01/2014		Fecha de Fin: 03/02/2014
Programador Responsable: Elvira Yánez		
Descripción: Se implementara ReporteEvaluacion.php en la capa de Presentación.		
Pruebas de Aceptación: ,Mostrar la información solicitada		

PRUEBAS DE ACEPTACIÓN.

Prueba de Aceptación		
Código: 1	Historia de Usuario: Reporte aspirante evaluación	
Nombre: Devolver un mensaje en caso que no exista evaluación rendida por aspirante		
Responsable: Elvira Yánez		Fecha:08/02/2014
Descripción: Devolver el mensaje “La información no existe”		
Condiciones de Ejecución: En la base de datos no debe existir datos		
Pasos de Ejecución: 1.Seleccionar el Reporte		
Resultado Esperado: - “La información no existe”		
Evaluación de la prueba: Satisfactorio		

Prueba de Aceptación	
Código: 2	Historia de Usuario: Reporte aspirante evaluación
Nombre: Mostrar la información solicitada.	
Responsable: Elvira Yánez	Fecha:08/02/2013
Descripción: al seleccionar el reporte debe generarse con la información solicitada.	
Condiciones de Ejecución: Debe de estar llena la base de datos para que devuelva datos, además debe existir una conexión con acceso a datos	
Pasos de Ejecución: 1.Seleccionar el Reporte	
Resultado Esperado: - Mostrar la información	
Evaluación de la prueba: satisfactorio	

TARJETA CRC.

Nombre de la Clase: ReporteEvaluación	
Responsabilidad	Colaboración
cedula, nombre, apellido, edad, estado, prueba calificación	Evaluación() Aspirante()

Anexo 4.

INTERFAZ DE USUARIO

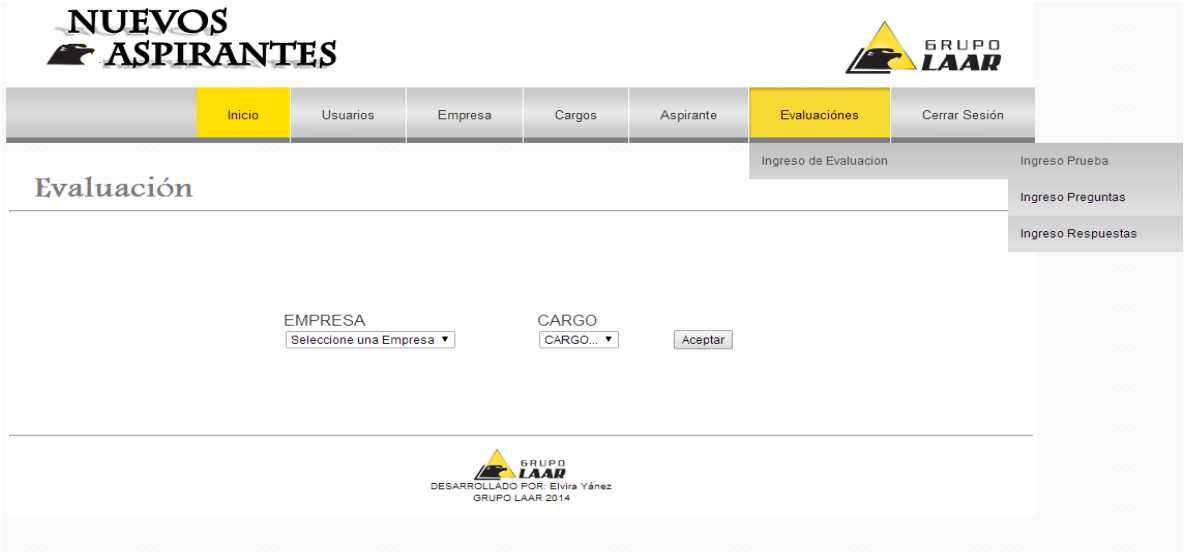
Registro de Aspirante permite al usuario aspirante registrase para poder acceder a la aplicación

Usuario permite ingresar Usuarios que trabajaran directamente con el sistema y de la misma manera modificar la información almacenada previamente , este proceso lo realiza el Administrador

Empresa permite ingresar una nueva empresa y modificar la información de esta una vez haya sido almacenada, este proceso lo realiza el Administrador

Cargos permite ingresar un nuevo cargo de una determinada empresa y de igual manera modificar esta información, este proceso lo realiza el Administrador y el Usuario GRUPO LAAR

Registro de Evaluación permite el ingreso de las pruebas, el ingreso de las preguntas y el ingreso de las respuestas. Y modificar cada una de estas una vez se la haya almacenado, este proceso lo realiza el Administrador y el Usuario GRUPO LAAR



Aspirante permite generar el Reporte de Aspirante y de igual manera dar una respuesta al aspirante una vez que este haya rendido una evaluación, este proceso lo realiza el Administrador y el Usuario GRUPO LAAR

BIBLIOGRAFÍA

[1].**Bandiera, Santiago Gabriel.** Vulnerabilidades Frecuentes en los Sitios Web. [En línea] Marzo de 2011. [Citado el: 13 de Agosto de 2013.] https://subversion.assembla.com/svn/espaciopersonal/tesis/00-Archivos-propios/Tesis%20-%20Vulnerabilidades%20frecuentes%20en%20sitios%20web.odt..

[2]. **UBAFCE.** Conceptos Fundamentales . [En línea] 2012. [Citado el: 15 de 08 de 2013.] http://www.websyllabus.org/files/15848-SI_UBAFCE_Alumnos_2012.pdf.

[3]. **OWASP.** Construir Aplicaciones y Servicios Web. [En línea] 27 de Julio de 2005. [Citado el: 15 de Agosto de 2013.] https://www.OWASP.org/images/b/b2/OWASP_Development_Guide_2.0.1_Spanish.pdf.

[4]. **OWASP** Riesgos Críticos en Aplicaciones Web. [En línea] 2013. [Citado el: 15 de Agosto de 2013.] https://www.OWASP.org/images/5/5f/OWASP_Top_10_-_2013_Final_-_Espa%C3%B1ol.pdf.

[5]. **Burnik, Sebastian.** Herramientas de Penetración. [En línea] 16 de Julio de 2013. [Citado el: 5 de Septiembre de 2013.] http://revista.seguridad.unam.mx/numero-18/pruebas-de-penetraci%C3%B3n-para-principiantes-5-herramientas-para-empezar.

[6]. **Beefproject**. [En línea] [Citado el: 8 de 03 de 2014.] http://beefproject.com/.

[7]. **Zaproxy**. [En línea] [Citado el: 08 de 03 de 2014.] https://code.google.com/p/zaproxy/wiki/Downloads?tm=2.

[8]. **Burp suite**. [En línea] [Citado el: 08 de 03 de 2014.] http://portswigger.net/burp/.

[9]. **OWASP.** Project OWASP ZAP. [En línea] [Citado el: 08 de 03 de 2014.] https://www.OWASP.org/index.php/OWASP_Zed_Attack_Proxy_Project.

[10]. **F.P, Daniel.** Análisis y Modelado de Amenazas. [En línea] 18 de Diciembre de 2006. [Citado el: 1 de Diciembre de 2013.] http://fortinux.com/wp-content/uploads/2010/12/Analisis-y-Modelado-de-Amenazas.pdf.