

Gestión del servicio de autenticación de usuarios a través de un servidor radius en la Universidad de Granma

Management of the user authentication service through a radius server at the University of Granma

Manejo de servicio de autenticación de usuarios con servidores RADIUS.

Martha Lorena Mendoza Navarrete¹

Marcos Tulio Zambrano Zambrano²

Ligia Vanessa Sánchez Parrales³

Manuel José Linares Alvaro⁴

Duliet Hung León⁵

(1) Universidad Laica Eloy Alfaro de Manabí. email: marthalorenamen1@hotmail.com

(2) Universidad Laica Eloy Alfaro de Manabí. email:marcostuliozambrano@hotmail.com

(3) Instituto Tecnológico Superior Portoviejo. Ecuador. email: ligia1980@live.com

(4) Universidad de Granma, Bayamo, Cuba. email: cheche@udg.co.cu

(5) Universidad de Granma, Bayamo, Cuba. email: duliet@udg.co.cu

Contacto: cheche@udg.co.cu

Resumen.

En el centro de datos de la Universidad de Granma, existe un servidor RADIUS para gestionar la autenticación de las conexiones para las redes WiFi y los accesos remotos (PAP) sin embargo, no existe un sistema informático que centralice y estandarice la información del servicio de autenticación de dichas conexiones. Actualmente, el análisis de la información de los registros generados por el servidor antes mencionado se ha vuelto un proceso complejo que consume tiempo y esfuerzo, lo que dificulta el trabajo y la toma de decisiones. Esta investigación se enfoca en el desarrollo de una aplicación web para mejorar el chequeo del servicio de autenticación de usuarios, así como la

obtención de la información precisa de forma segura, consistente y centralizada. Se empleó XP, PHP y CodeIgniter sobre el servidor web Apache; del lado del cliente la aplicación es soportada por el *framework* Bootstrap, basado en HTML5, CSS3 y JavaScript. Como gestor de base de datos se empleó MySQL y se utilizó el entorno de desarrollo integrado NetBeans. Como resultado y conclusión de este trabajo, se puede plantear que actualmente el Departamento de Redes de la Universidad de Granma, cuenta con una herramienta que facilita considerablemente tanto el chequeo del normal funcionamiento del servicio de autenticación de usuarios para los accesos remotos y redes WiFi, como el uso y la carga de trabajo que tiene éste.

Palabras clave: Autenticación de usuarios, radius, Análisis de registros.

Abstract.

In the computer network of the University of Granma, the RADIUS server manages the authentication of connections for Wi-Fi networks and remote access (PAP), however there is no computer system that centralizes and standardizes the information of the authentication service of these connections. Currently, the analysis of the information of the records generated by the aforementioned server has become a complex process that consumes time and effort, making it difficult to work and make decisions. This research focuses on the development of a web application to improve the verification of the user authentication service, as well as obtaining accurate information in a secure, consistent and centralized manner. XP, PHP and CodeIgniter were used on the Apache web server; on the client side, the application is supported by the Bootstrap framework, based on HTML5, CSS3 and JavaScript. MySQL was used as the database manager and the NetBeans integrated development environment was used. As a result and conclusion of this work, it can be stated that currently the Network Department of the University of Granma has a tool that considerably facilitates both the checking of the normal functioning of the user authentication service for remote accesses and WiFi networks, such as the use and workload of the latter.

Keywords: user authentication, Radius server, Log analyzers.

Introducción.

La red de la Universidad de Granma (UdG) presta numerosos servicios a estudiantes, profesores, investigadores y trabajadores de la institución; entre los más necesarios y demandados se pueden citar el acceso a servicios a redes públicas y privadas, navegación en internet, correo electrónico, mensajería instantánea, almacenamiento de información en la nube local, el empleo y acceso a

entornos virtuales de aprendizaje, videoconferencias y la transferencia de ficheros, entre otros. Así mismo, brinda la posibilidad de acceder a los servicios antes mencionados mediante dispositivos móviles empleando conexiones WiFi, lo que facilita la conexión en cualquier parte de las instituciones de sus diferentes sedes.

Por otra parte, el centro de datos de la Universidad de Granma mantiene en explotación un conjunto de herramientas de seguridad para chequear el comportamiento de la red informática, controlar los servicios que presta y estudiar sus registros. La seguridad incluye varios aspectos importantes como el control de quiénes acceden a la red (*Authentication*), a qué servicios (*Authorization*) y por qué tiempo (*Accounting*) (AAA).

Para controlar el acceso a la red a través de conexiones inalámbricas y remotas, la UdG emplea el sistema de seguridad RADIUS, implementado en un servidor *FreeRADIUS*, el cual, se encarga de autenticar a los usuarios que accedan a la red ya sea a través de los servicios de accesos remotos o de los puntos de acceso *Wifi*, generando archivos de texto de registros con información detallada de cada uno de ellos.

En el centro de datos de la Universidad de Granma se gestiona la información del servicio de autenticación mencionada anteriormente, la misma es tomada de los registros generados por el servidor *FreeRadius*, éste genera diariamente archivos de texto con los datos del usuario, que son actualizados periódicamente mientras la sesión está activa. Debido a que los datos se encuentran en dichos archivos, resulta difícil poder obtener la información precisa en su debido momento, por lo que no se logra un aprovechamiento óptimo de los datos que éste brinda.

Estos datos constituyen un registro de hechos y acontecimientos que por sí solos tienen muy poco valor para los usuarios que hacen uso de ellos (supervisores, administradores de sistemas), además su interpretación requiere un alto costo de tiempo, lo que compromete la calidad de los resultados y afecta el factor humano. Sin embargo, procesados debidamente, pueden ser empleados para fines estadísticos, investigaciones de seguridad, actividades informático - forenses y convertirse en una información estratégica para la toma de decisiones tanto en la planeación del crecimiento y desarrollo de la red, como en el diseño de su sistema de seguridad.

A partir del diagnóstico realizado se destacan las siguientes **limitaciones**:

- La consulta de datos se realiza sobre ficheros de texto lo que dificulta su gestión, interpretación y posterior toma de decisiones.

- Resulta extremadamente complejo realizar consultas y búsquedas de información por uno o varios criterios.
- Necesidad de invertir largos períodos de tiempo en la revisión e interpretación de los registros.
- Carencia de una muestra de los datos en forma gráfica que contribuya a optimizar la información registrada.
- Carencia de las facilidades para realizar reportes y/o estudios estadísticos.
- Por lo general para consultar los registros se leen directamente los archivos por ser el método más directo, pero esto constituye una violación de los principios básicos de la informática forense.

Lo antes expuesto ha permitido detectar la existencia de un **problema científico** que consiste en la necesidad de una herramienta, que organice la información de los registros generados por el servidor RADIUS en la Universidad de Granma, de manera que facilite los procesos de interpretación, seguimiento, y la posterior toma de decisiones.

En correspondencia con el problema planteado se formula como **objetivo de la investigación** desarrollar un sistema informático basado en la Web, que favorezca el proceso de gestión del servicio de autenticación de usuarios a través del servidor RADIUS en la Universidad de Granma, a partir de los registros de acceso generados por el mismo.

Esto ha facilitado que se defina como **objeto de estudio de la investigación** el proceso de gestión del servicio de autenticación de usuarios a través del servidor RADIUS.

El cual delimita como **campo de acción** la informatización de la gestión del servicio de autenticación de usuarios a través del servidor RADIUS.

Materiales y métodos.

La metodología empleada para el desarrollo de esta aplicación, se basó en la Programación Extrema (del inglés *Extreme Programming*).

Se realizó un análisis de las características principales del sistema a desarrollar, de acuerdo al problema por el cual fue concebido. Se especificaron las funcionalidades que se desean informatizar en la búsqueda de satisfacer las necesidades del cliente, además de describirse la construcción de la propuesta con el desarrollo de las historias de usuario a través de iteraciones y se definieron tareas y principios de diseño para su implementación.

Se puede mencionar entonces que el desarrollo de la metodología XP, implicó el despliegue de varias de varias etapas, pues ante todo, fue necesaria la definición de las personas relacionadas con el sistema, seguida por la fase de exploración, la cual incluyó las historias de usuarios. Luego fue preciso definir los requerimientos no funcionales y posteriormente la fase de planificación, la cual contuvo la estimación del esfuerzo por historias de usuarios, el plan de duración de las iteraciones y el plan de entrega; a continuación se valoraron las interacciones y las tareas y luego se llegó a la fase de producción con sus correspondientes codificaciones y pruebas de software. Finalmente se procedió a la valoración de los resultados obtenidos.

A continuación, se describen y caracterizan de manera muy resumida, los aspectos más relevantes de la metodología empleada.

Personas relacionadas con el sistema: Se determinó que éste sistema solamente exigía de 2 actores: el administrador, que tendrá acceso a todas las opciones que brinda el sistema, así como a la gestión de las cuentas de usuarios, y el supervisor, quien tendría acceso a todas las opciones que brinda el sistema, excepto a la gestión de las cuentas de usuario.

Fase de exploración, determinación de las historias de usuarios: constituyen la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. Se determinó la existencia de 18 historias de usuarios, que se pueden apreciar en la tabla 2.

Requisitos no funcionales del software:

Interfaces de usuario: Diseño sencillo, moderno y fácil de usar. Se respetó el manual de identidad visual de la organización ya que predominan los colores blanco, azul, gris y rojo. El contenido será mostrado de manera comprensible y fácil de leer.

Usabilidad: incluye el tipo de usuario final y en este caso se puede plantear que la aplicación será manipulada por el cliente que en este caso son los especialistas del Centro de Datos de la Universidad de Granma y el Departamento de Redes.

Tipo de Aplicación Informática: aplicación web. Finalidad: aplicación web del Centro de Datos de la Universidad de Granma, donde dicho departamento, podrá gestionar la información del servicio de autenticación de usuarios generada a través de un servidor RADIUS.

Seguridad: Al desplegarse el sistema en el servidor web, debería garantizarse la seguridad física de los archivos y la base de datos, definiendo los permisos adecuados. El sistema debería proporcionar seguridad y control de acceso para las funciones basadas en permisos de usuario. Los datos de usuario residen en la base de datos con la contraseña encriptada. En la administración, el contenido a mostrarse y las opciones disponibles dependerán de los privilegios del usuario. De este modo, solo usuarios administradores pueden Gestionar Usuarios. Este sitio solo es visible desde cualquiera de las redes de la Universidad de Granma o de la red Universitaria Nacional RedUniv, no es visible desde redes públicas.

Confidencialidad: La gestión de la información estará protegida contra accesos no autorizados utilizando mecanismos de autenticación y autorización. La autenticación consistirá en suministrar un nombre de usuario y una contraseña que debe ser de conocimiento exclusivo de la persona que se autentica.

Disponibilidad: El sistema puede ser utilizado a través de la red, solo por el personal autorizado, hospedado en un servidor web.

Eficiencia: El sistema deberá ser rápido ante las solicitudes de los usuarios, pero dependerá de la velocidad de la conexión a través de la que se accede, de las condiciones tecnológicas del hosting y otros.

Restricciones de diseño: para ello se emplearon varias tecnologías como los framework CodeIgniter y bootstrap para el desarrollo de aplicaciones web basadas en PHP. Esta aplicación se desarrolló para funcionar en un servidor web Apache con un gestor de bases de datos MySQL.

Requerimientos mínimos de software: La tabla 1 muestra los requerimientos mínimos y los recomendados, los cuales fueron desarrollados a partir de los sistemas de control de versiones existentes en el centro de datos de la UdG.

Mínimos	Recomendados
PHP Versión 5	PHP versión 7
MySQL (MariaDB), versión 10	MySQL (MariaDB), versión 10.4
Apache o Nginx con el módulo mod_rewrite activado, versión 2.2	Apache o Nginx con el módulo mod_rewrite activado, versión 2.4

Tabla 1. Requecimientos mínimos y recomendados para la aplicación web desarrollada.

Fase de planificación: Los principales objetivos de esta fase consistieron en fijar la prioridad de cada una de las historias de usuario por parte del cliente, estimar el esfuerzo total necesario para su

implementación a criterio de los desarrolladores y establecer, de conjunto, el contenido y fechas de las entregas del sistema. Una entrega debería obtenerse en no más de 3 meses. Esta fase duró unos pocos días.

Estimación de esfuerzo por historia de usuario: Para el desarrollo de la aplicación propuesta fue necesario realizar una estimación del esfuerzo para cada una de las historias de usuario identificadas, la cual permitió tener una medida real de la velocidad de progreso del proyecto, y ofrece una guía razonable a la cual ajustarse. (Tabla 2). Se aprecia claramente que la historia de usuarios que mayor costo o esfuerzo implica es la de importar datos desde ficheros de texto (5).

N _{ro}	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración
1	Autenticar usuario	Media	Bajo	0.1	1
2	Cerrar sesión	Alta	Bajo	0.1	1
3	Gestionar usuarios del sistema	Media	Bajo	0.3	1
4	Cambiar contraseña	Baja	Bajo	0.1	1
5	Importar datos desde ficheros de texto	Alta	Alto	0.5	1
6	Visualizar usuarios conectados por WiFi	Alta	Alto	0.4	2
7	Visualizar usuarios conectados por accesos remotos	Alta	Alto	0.4	2
8	Visualizar todas las conexiones	Alta	Alto	0.4	2
9	Total de conexiones	Alta	Medio	0.3	2
10	Visualizar usuarios de manera duplicada	Alta	Alto	0.4	2
11	Errores de conexión	Alta	Medio	0.3	3
12	Usuarios con mayor tiempo de conexión	Alta	Medio	0.3	3
13	Ultima conexión	Alta	Medio	0.3	3
14	Gráfico del tiempo de conexión por WiFi	Alta	Alto	0.4	3
15	Grafico del tráfico de datos.	Alta	Alto	0.4	3
16	Exportar reporte de conexiones por WiFi	Alta	Alto	0.3	4
17	Exportar reporte de conexiones por accesos remotos	Alta	Alto	0.3	4
18	Exportar reporte de conexiones.	Alta	Alto	0.3	4

Tabla 2. Estimación de esfuerzo por historia de usuario.

Plan de duración de las iteraciones: tuvo como objetivo definir el número de entregas o releases que se realizarán en el transcurso del proyecto y las iteraciones que se requieren para desarrollar cada una. El cliente se encarga de decidir cuáles historias de usuario comprende la primera entrega según sus prioridades para darle valor a su negocio y que por tanto justifique su ejecución y así sucesivamente para las demás. De acuerdo a lo mencionado anteriormente se decidió realizar dicha planificación en cuatro iteraciones, las cuales se detallan a continuación en la Tabla 3.

Release	Orden de la UH a implementar	Duración total
1	1, 2, 3, 4, 5	2 semanas y un día

2	6, 7, 8, 9, 10	3 semanas y 4 días
3	11, 12, 13, 14, 15	3 semanas y 2 días
4	15, 17, 18	1 semana y 4 días

Tabla 3. Plan de duración de las iteraciones.

El sistema fue diseñado para facilitar la gestión de la información generada por el servidor RADIUS en el Centro de Datos de la Universidad de Granma, de forma dinámica y agradable al usuario. Para el diseño de la aplicación se realizó una maqueta de una estructura que debía presentar una cabecera en la parte superior para mostrar el nombre del sistema y el enlace de cierre de sesión; además, debería existir un panel izquierdo con un menú para el acceso a las diferentes funcionalidades. En el área central se mostrarían los elementos visuales de la historia de usuario en ejecución, ya sean gráficos, tablas e imágenes. (Figura 1)

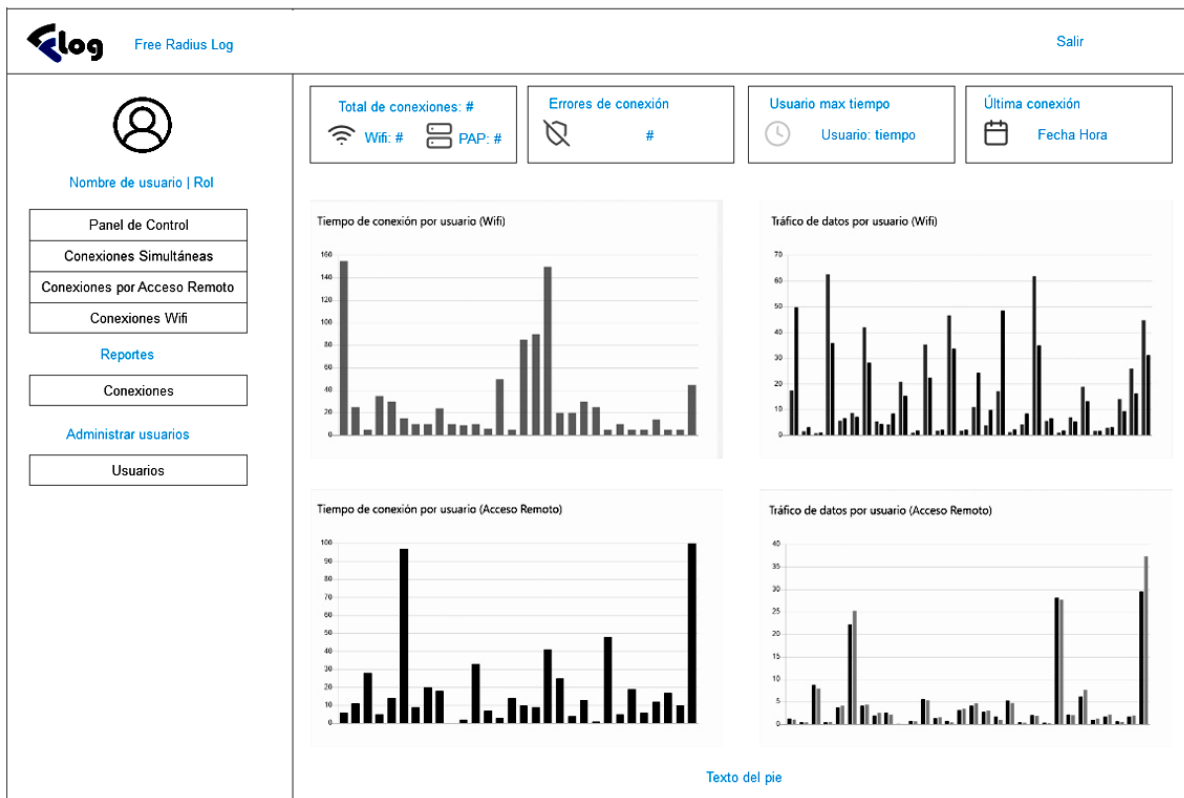


Figura 1. Diseño básico de la interfaz principal del sistema.

La base de datos se diseñó de manera que estuviera compuesta por 9 tablas y normalizada en tercera forma normal, los nombres de tablas y campos se escriben en minúscula, idioma inglés, y se usa guión bajo para separar las palabras en caso necesario. Las claves primarias por lo general se nombrarán id, y los nombres de los campos claves foráneas se forman por el nombre de la tabla de referencia con el sufijo id. En la Figura 2 se muestra el diseño de la base de datos.

Arquitectura del sistema.

La arquitectura del sistema se basa fundamentalmente en el patrón MVC soportado por el framework CodeIgniter. El usuario envía una petición al Controlador, y este le solicita al Modelo los datos; el modelo le devuelve los datos al controlador, el cual selecciona una vista; se devuelve la vista seleccionada al controlador y este devuelve una vista al usuario que carga los datos del modelo seleccionado.

Patrones de diseño y arquitectura.

Como patrón de arquitectura se utilizó MVC que permite una gran separación entre la lógica y la presentación. Un patrón de diseño es una abstracción de una solución en un nivel alto, y solucionan problemas que existen en muchos niveles de abstracción. Uno de los patrones más conocidos son los patrones GRASP (Patrones Generales de Software para Asignar Responsabilidades) que describen los principios fundamentales de la asignación de responsabilidades a objetos. El patrón Experto se utiliza con frecuencia en la asignación de responsabilidades; es un principio de guía básico que se utiliza continuamente en el diseño de objetos. Expresa la intuición común de que los objetos hacen las cosas relacionadas con la información que tienen. El patrón Alta cohesión es un principio evaluativo que aplica un diseñador mientras evalúa todas las decisiones de diseño. Indica la relación que existe entre los elementos de un mismo módulo. Es la medida de la relación funcional de los elementos de un módulo. El objetivo es organizar estos elementos de manera que los que tengan una mayor relación a la hora de realizar una tarea pertenezcan al mismo módulo, y los elementos no relacionados, se encuentren en módulos separados. El patrón Bajo acoplamiento impulsa la asignación de responsabilidades de manera que su localización no incremente el acoplamiento hasta un nivel que nos lleve a los resultados negativos que puede producir un acoplamiento alto. Es el grado de interdependencia entre los módulos. Un buen diseño se caracteriza por un acoplamiento mínimo, es decir, unos módulos tan independientes los unos de los otros como sea posible (Giraldo, Acevedo, y Moreno, 2011).

Codificación.

En la implementación de las Historias de Usuario se utiliza el lenguaje de programación web PHP5 el cual es un lenguaje del lado del servidor, con el framework de desarrollo CodeIgniter y el gestor de base de datos MySQL. Se hizo necesario un estándar de codificación adaptable al cambio, legible y entendible, es por ello que para realizarlo se tuvo en cuenta el estándar de codificación de CodeIgniter.

Estándares de codificación.

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, este debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez. Al comenzar un proyecto de software, se debe establecer un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada (Lizard, 2012). El framework CodeIgniter sigue una serie de estándares que serán respetados, sin embargo los que aquí se detallan son los adoptados por el programador al escribir nuevo código sobre la estructura base.

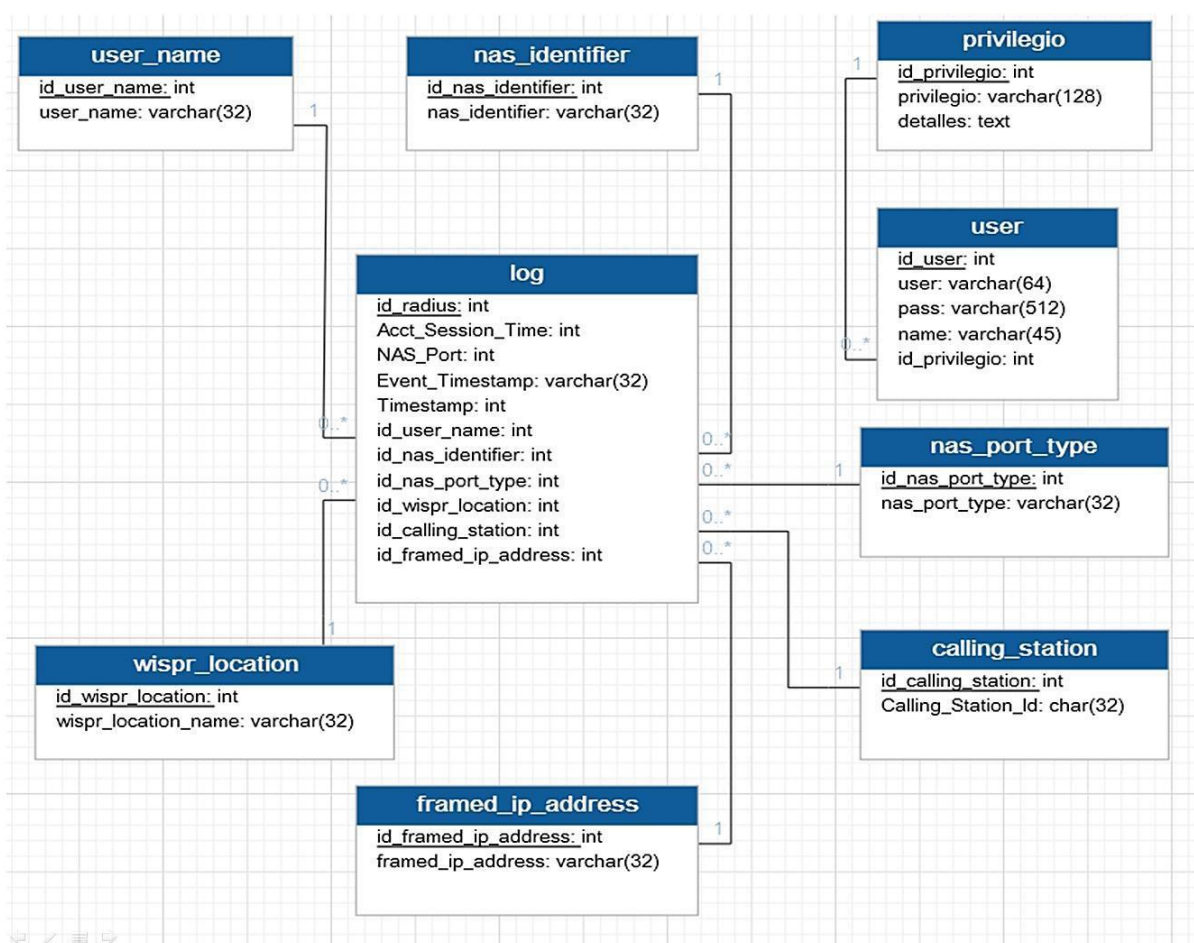


Figura 2 Diseño de la base de datos.

Nomenclatura de Clases y Métodos: los nombres de clases comienzan con una letra mayúscula. Varias palabras se separan con un guión de subrayado y no usar Camel Case. Todos los otros métodos de clase se escribirán completamente en minúsculas y su nombre debería indicar claramente su función. Se evita los nombres demasiado largos y detallados. Las funcionalidades relacionadas se

agruparán en clases, siempre respetando el patrón MVC, de modo que se escriba lo relacionado al acceso a base de datos en los modelos, el HTML en las vistas y la lógica de negocio en los controladores.

Nombres de variables: la directriz para el nombramiento de variables es muy similar al usado para métodos de clase. Concretamente, las variables deben contener solamente letras minúsculas, usar guiones de subrayado como separadores y tener un nombre que razonablemente indique su propósito y contenido. Se escribirá una sentencia por línea.

Pruebas de software.

Se realizaron pruebas de dos tipos: de aceptación y unitarias. Las pruebas de aceptación fueron creadas conjuntamente con el cliente y deben reflejar las necesidades funcionales de este. Las unitarias son realizadas por el programador para verificar que todos los métodos de las clases funcionan correctamente. A una historia de usuario se le aplican tantas pruebas como sean necesarias para garantizar su correcta implementación; hasta que no pase sus pruebas, la historia de usuario no es aceptada. La Tabla 3 muestra el caso de prueba de aceptación realizada a la historia de usuario “Importar datos desde ficheros de texto.” En el Anexo 3 se pueden consultar el resto de los casos de pruebas de aceptación.

Caso de prueba de aceptación	
Número de caso de prueba: 13	Número de historia de usuario: 5
Nombre de la prueba: verificar que los datos se importaron correctamente.	
Descripción de la Prueba: El usuario accede a cualquier reporte del sistema, y comprueba que los datos existen en la base de datos.	
Condiciones de Ejecución: El usuario debe estar autenticado.	
Condiciones de Ejecución: El usuario debe estar autenticado. Entrada / Pasos de ejecución: El usuario selecciona la opción del menú “Reporte de conexiones por usuario”. El sistema muestra el listado con los datos del reporte. El usuario comprueba que el reporte contiene los datos que han sido importado	
Resultado Esperado: El reporte contiene los datos importados de los ficheros de texto.	
Evaluación de la Prueba: Satisfactorias	

Tabla 3 Prueba de aceptación “Verificar que los datos se importaron correctamente”

Los casos de pruebas de aceptación definidos, fueron ejecutados de conjunto con el cliente por cada iteración, lo que permitió probar el sistema desde el punto de vista del usuario. El siguiente gráfico

(4), muestra los resultados, diferenciándolos entre los casos exitosos y los que fallaron, donde se observa una clara tendencia hacia el aumento de los casos de éxito.

Resultados y discusión.

La implementación del producto informático generado a partir de esta investigación beneficia a la entidad, pues se mejora el proceso de gestión del servicio de autenticación de usuarios a través de un servidor RADIUS de las redes WIFI y PAP. Con el uso de la aplicación se facilita el monitoreo de dicho proceso, invirtiendo un tiempo considerablemente menor, lo que significa mayor agilidad en el trabajo y la reducción de las posibilidades de cometer errores durante la manipulación de los datos.

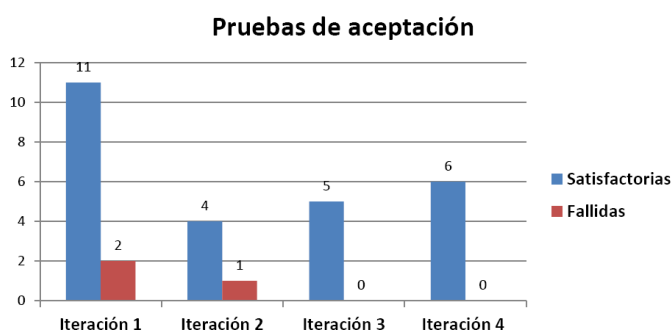


Figura 4 Resultados de las pruebas de aceptación al final de cada iteración.

Detallando lo antes mencionado, se puede plantear que el nuevo software conduce a una mayor estabilidad del servicio de autenticación, pues a partir del momento de la puesta en marcha del sistema, se tiene un control permanente del normal funcionamiento del RADIUS, lo cual conduce de manera indudable a que mejore la estabilidad del mismo, al facilitar la reducción a valores mínimos de tiempos de respuesta, ante interrupciones no planificadas. Se tiene un mejor control sobre el empleo que dan los usuarios a los accesos a la red a través de terminales remotas o dispositivos móviles, pues se pueden conocer parámetros, como tiempo de conexión, tráfico generado, intentos de conexiones fallidas, entre otros; que anteriormente no era posible determinar, a partir del momento en que se implementó esta herramienta, es posible generar con extrema facilidad informes con formatos compatibles con Excel o documentos portables (pdf) a partir de la información organizada, y con toda la información que genera la nueva plataforma, se condiciona una adecuada planificación sobre las necesidades de crecimiento tanto mediatas inmediatas de los servicios de accesos remotos y acceso por redes inalámbrica desde dispositivos portátiles.

Conclusiones.

- El Departamento de Redes de la Universidad de Granma cuenta actualmente con un sistema informático (FreeRadiusLog) que favorece los procesos de chequeo y seguimiento del servicio de autenticación de los usuarios que acceden tanto por las redes inalámbricas como a través del servicio de accesos remotos mediante líneas conmutadas.
- Esta herramienta permitirá por un lado, un mejor control del empleo que se le da a los servicios de accesos remotos y conexiones inalámbricas por parte de los usuarios, pues facilitará conocer con certeza detalles estadísticos importantes relacionados con el uso real que dan los éstos a los servicios antes mencionados, y por otro facilitará la emisión de informes relacionados con los usuarios o el servicio.
- Este sistema garantizará una mejor calidad del servicio, pues al tenerse una aplicación capaz de monitorear el funcionamiento del mismo, resultará inevitable que mejoren tanto la estabilidad como la eficiencia del servicio prestado.

Bibliografías.

- Abrahamsson, P., Salo, O., Ronkainen, J., y Warsta, J. (2017). Agile software development methods: Review and analysis. arXiv preprint arXiv:1709.08439.
- Adail. (2017). Vulnerabilidad de salto de autenticación en FreeRADIUS, from <http://ciberseguridad.cubava.cu/2017/06/vulnerabilidad-de-salto-de-autenticacion-en-freeradius/>
- Agudelo, B. P. (2018). Analisis, diseño e implementacion de un sistema de gestion de competiciones deportivas y recreativas en comfamiliar risaralda, from <http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/8944/T006.7019%20P438.pdf?sequence=1&isAllowed=y>.
- Aguilar, F. S. (2018). Metodologías Ágiles: Extreme Programming o XP. from <https://www.franciscosanchezaguil.com/metodologias-agiles-extreme-programming-xp/>
- Alba;, T. (2014). 10 Razones para desarrollar una web con Bootstrap, from <https://diseño creativo.com/developar-web-bootstrap/>
- Aries, B. (2010). Ventajas de Apache Web Server, from https://techlandia.com/ventajas-apache-web-server-lista_109947/
- Auben. (2014). Monitoreo y Gestión de Redes. from <http://www.auben.net/index.php/tecnologias/monitoreo-y-gestion-de-red>

- Barzaga, A. L. (2016). Sistema de gestión de información para la entrega de medios a casos críticos en la Dirección de Trabajo y Seguridad Social de Guisa. (Tesis).
- Benítez, F. C. (2017). Capítulo III. Trabajos relacionados. from <https://docplayer.es/15253574-Capitulo-iii-trabajos-relacionados.html>
- Calabria, L., y Píriz, P. (2003). Metodología xp. Informe. Montevideo: Universidad ORT Uruguay, Facultad de Ingeniería.
- Carrero, R. G. (2012). 10 razones para usar CSS, from <http://www.maestrosdelweb.com/usarcss/>
- Corporation, L. (2015). Entornos de Desarrollos Integrados. Retrieved 23 de enero, 2015, from <http://es.slideshare.net/GhaBiithahh/entornos-de-desarrollo-integrados>
- Chavarría-Báez, L. (2016). Sobre el uso de herramientas CASE para la enseñanza de bases de datos, from [http://www.iiisci.org/journal/CV\\$/risici/pdfs/CB321MR16.pdf](http://www.iiisci.org/journal/CV$/risici/pdfs/CB321MR16.pdf)
- Chavarría-Báez, L., y Rojas, N. O. (2016). Sobre el uso de herramientas CASE para la enseñanza de bases de datos. *Sistemas, Cibernetica e Informática*, 13(2), 51-56.
- Dailín, M. C., y Hernán, H. M. (2016). IMPLEMENTACIÓN DE UN SISTEMA DE GESTIÓN EFICAZ PARA LA RED NACIONAL UNIVERSITARIA. CUJAE.
- De la Torre, A. (2006). Lenguajes del lado servidor o cliente. from http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html.
- Delgado, J. K. C., Dulce, E. R., y Toledo, R. A. J. (2016). La importancia del monitoreo en redes de datos. *Boletín Informativo CEI*, 2(3).
- Días, C. (2012). Ventajas/Desventajas, from <http://herramientascase.blogspot.com/2012/04/ventajasdesventajas.html>.
- Dias, K., Jose, A., y Fernandes, L. (2019). REMOTE SENSOR NETWORKS FOR PREDICTION OF CHILLI CROP DISEASES USING INFRARED IMAGE PROCESSING TECHNIQUES.
- Díaz, C. M. B., Aliaga, Z. L., Galindo, L. V. C., Rivero, L. A. S., Céspedes, A. P., y Alicio, A. M. (2018). Propuesta de sistema informático para la gestión del plan de trabajo individual de los profesores. *Revista de Investigación en Tecnologías de la Información*, 6(11).
- Díaz, L. M., y Collazo, G. (2017). La programación extrema. from https://www.researchgate.net/publication/318211906_La_programacion_extrema
- Duarte, E. (2012). ¿Qué Es JavaScript ? Ventajas Y Desventajas, from <http://blog.capacityacademy.com/2012/10/19/que-es-javascript-ventajas-y-desventajas/>
- Escalona, S. B. (2012). Protocolos de control de acceso RADIUS. *Revista Telemática*, 10(1).
- Escobar Pérez, R. X. (2012). Implementación del protocolo radius en una red virtual para mejorar las seguridades informáticas. Quito: Universidad de las Américas, 2012

- Fuentes, C. A. M., Deras, K. A. V., y Hernández, R. E. H. (2012). Implementación de un prototipo de red inalámbrica que permita elevar los niveles de seguridad a través de la autenticación de un servidor Radius para los usuarios que accedan a internet en el edificio Francisco Morazán de la UTEC. from <https://studylib.es/doc/8677865/facultad-de-inform%C3%A1tica-y-ciencias>. Universidad Tecnológica de El Salvador
- Gardey, J. P. P. y. A. (2012). Definición de html. from <https://www.definicion.de/html/>
- Giraldo, G. L., Acevedo, J. F., y Moreno, D. A. (2011). Una ontología para la representación de conceptos de diseño de software. *Revista Avances en Sistemas e Informática*, 8(3), 103-110.
- González, J. F. (2013). Introducción a las metodologías ágiles. Otras formas de analizar y desarrollar.
- González, Y. D., y Romero, Y. F. (2012). Patrón Modelo-Vista-Controlador. *Revista Telemática*, 11(1), 47-57.
- Gouchat, J. D. (2018). El gran libro de HTML5, CCS y JavaScript.
- Hidalgo, L. M. T. (2013). SERVIDOR REMOTO DE AUTENTICACIÓN DE USUARIOS (RADIUS) PARA LA INTEGRACIÓN CON EL SISTEMA ADMINISTRATIVO CONTABLE EN EL PROVEEDOR DE SERVICIO DE INTERNET SPEEDYCOM CIA. LTDA. DE LA CIUDAD DE AMBATO. From http://repositorio.uta.edu.ec/bitstream/123456789/6275/1/Tesis_t858si.pdf
- Hinostroza. (2010). ¿Cómo funciona PHP?. from <http://phpptatic.blogspot.com/2010/04/como-funciona-php.html>
- Ibáñez, D. H. (2012). Implantación d directrices ITIL en un Departamento de Soporte y Operaciones de una empresa. (Tesis de grado), Universidad Carlos III de Madrid, Leganés.
- Iglesias;, P. D., y Ávila, C. P. (2007). Seguridad en Sistemas de Información. from <http://sabia.tic.udc.es/docencia/ssi/old/2006-2007/docs/trabajos/11%20-%20Radius.pdf>
- Iruela, J. (2016). Los gestores de bases de datos más usados- *Revista Digital INESEM*. from <https://www.revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados> . *Revista Digital INESEM*
- Jordisan. (2006). ¿Qué es un „framework“? from. <https://jordisan.net/blog/2006/que-es-un-framework>
- Jorge. (2009). Qué nos ofrece CodeIgniter frente a otros frameworks., from <https://blogs.ua.es/jorgeparent/2009/11/24/que-nos-ofrece-codeigniter-frente-a-otros-frameworks>
- Joskowicz, J. (2008). Reglas y prácticas en eXtreme Programming. Universidad de Vigo, 22.
- Larson, Q. (2016). Ventajas Y Desventajas De JavaScript, from <https://github.com/ashish9342/FreeCodeCamp/wiki/Ventajas-Y-Desventajas-De-JavaScript>

- Letelier, P., y Penadés, M. C. (2012). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP).
- Liz. (2014). Chart.js Buenas graficas desde JS , from <http://soyprogramador.liz.mx/chart-js-buenas-graficas-desde-js/>
- Lizard, M. E. A. (2012). Qué es un Estándar de Codificación, from <https://arevalomaria.wordpress.com/2012/11/02/propuesta-de-estandar-de-desarrollo-o-codificacion-primera-entrega-programacion>
- Luis, G. R. (2016). Compartición de recursos, usando NAS, from <https://es.scribd.com/document/315006200/Articulo-radius>
- Madayeli Pérez, H. (2017). Ventajas y desventajas de MySQL, oracle, visual foxpro y access from <https://mape309site.wordpress.com/2017/11/15/ventajas-y-desventajas-de-mysql-oracle-visual-foxpro-y-access/>
- Maldonado Tapia, Á. V. (2012). Implementación de un portal cautivo que permita el control de acceso al servicio de internet a los estudiantes del Colegio San Luís Gonzaga a través de una autenticación de los usuarios mediante un servicio AAA implementando en un servidor que trabaje con protocolo Radius.
- Mendoza González, G. (2008). Herramienta de Desarrollo Netbeans. Herramienta de Desarrollo Netbeans.
- Morales, E. (2019). TIC en la Educación, from <https://ticenlaeducacion111.blogspot.com/2019/01/algunos-terminos-utilizados-en-la.html>
- Moreno, J. C., y Marciszack, M. M. (2013). La Usabilidad Desde La Perspectiva De La Validación de Requerimientos No Funcionales Para Aplicaciones Web. Córdoba: Universidad Tecnológica Nacional.
- Pastuña, C., Elizabeth, P., y Gallardo Arias, Í. G. (2016). Implementación y configuración de un servidor cautivo utilizando herramientas de software libre (linux) para mejorar el acceso a la red inalámbrica en el laboratorio de redes de la carrera de ingeniería en informática y sistemas computacionales de la Universidad Técnica de Cotopaxi durante el periodo 2015. LATACUNGA/UTC/2016.
- Pecos, D. (2012). MySQL. from http://www.danielpecos.com/docs/mysql_postgres/x57.html
- Peña. (2017). Ventajas y desventajas de Java Netbeans, from <http://kdsjafk.blogspot.com/>
- Perez, J. C. G. (2012). TODO SOBRE BASES DE DATOS (GBD), from <http://lasbasesdedatoss.blogspot.com/2012/11/sistemas-gestores-de-bases-de-datos.html>
- Pineda, J. M. V. (2016). Qué es CodeIgniter y cuáles son algunas de sus ventajas. from <https://www.coriaweb.hosting/codeigniter-cuales-algunas-ventajas/>

- Prado, C. E. P. (2014). ¿Qué es y por qué aprender SQL? from <https://www.devcode.la/blog/que-es-sql/>
- Presto, M. E. (2017). Pruebas en las metodologías ágiles, from <https://mariaelisapresto.wordpress.com/author/mepresto/>
- Rahman, S. F. (2018). 24 bibliotecas de JavaScript para crear bellas gráficas, from <https://www.sitepoint.com/15-best-javascript-charting-libraries/>
- Reyes, J. J. P. (2018). ¿Qué es HTML? From <https://www.devcode.la/blog/que-es-html/>
- Rodríguez, O. C. (2012). Sistema Automatizado para el Control de la Aplicación de los Resultados de la Ciencia. from <http://dspace.uclv.edu.cu/bitstream/handle/123456789/7745/Proyecto%20%20Diploma%20ultimo%20%28Autoguardado%29.pdf?sequence=1&isAllowed=y> . Universidad Central “Marta Abreu” de Las Villas.
- Santos, L. A. P. (2013). Estudio comparativo de los sistemas de gestión y monitoreo basados en los requerimientos generales de la red de un campus universitario. (Tesis de maestría), Pontificia universidad católica del Ecuador, Quito.
- Souza, A. A. (2017). Aplicación Web para la Gestión y Control de Horarios en las Carreras de la Facultad de Administración, Finanzas e Informática. Babahoyo: UTB.
- Tamayo Hidalgo, L. M. (2013). Servidor Remoto de Autenticación de Usuarios (RADIUS) para la integración con el Sistema Administrativo Contable en el Proveedor de Servicio de Internet SPEEDYCOM CIA. LTDA. de la Ciudad de Ambato. Universidad Técnica de Ambato. Facultad de Ingeniería en Sistemas.
- TAPIA, A. V. M. (2012). Implementación de un portal cautivo que permita el control de acceso al servicio de internet a los estudiantes del colegio san luis gonzaga a través de una autenticación de los usuarios mediante un servicio aaa implementado en un servidor que trabaje con protocolo radius.
- Toledo, H. F. R. (2017). Servicios. from <http://www.tecnotopia.com.mx/redes/redservicios.htm>
- Torres, M. (2014). Lenguajes de programación del lado servidor. - michelletorres.mx. from <https://www.michelletorres.mx/lenguajes-de-programacion-del-lado-servidor/>
- Vanamala, M., Yuan, X., y Morgan, M. (2019). A Course Module on HTML5 New Features and Security Concerns. *The Journal of Computing Sciences in Colleges*, 23.
- Vázquez, J. M. A. (2018). Diseño e implantación de un servidor RADIUS basándose en servidores Kerberos, from <http://bibing.us.es/proyectos/abreproy/12446/fichero/PFC-2446-ACERO.pdf>
- Velandia, L. N. M., y López, W. M. L. (2015). Escoger una metodología para desarrollar software, difícil decisión. *Revista Educación en Ingeniería*, 10(20), 98-109.

Vickey, D. K., Ali, M., y Maheshwari, K. (2018). Smart Display Notice Board Using Raspberry Pi.