

Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación

Trabajo Especial de Grado  
**Sistema de Administración y Monitoreo  
para los Punto de acceso Inalámbricos de  
la Facultad de Ciencias de la Universidad  
Central de Venezuela.**



Trabajo Especial de Grado presentado  
Por el bachiller:

**Kendall Rober Navarro Urbina**  
C.I.: 16.412.971

Para optar al Título de  
**Licenciado en Computación**

**Tutora:**  
Prof. Ana Verónica Morales.

Caracas, Octubre de 2011

## ACTA

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación de la Facultad de Ciencias de la Universidad Central de Venezuela, para examinar el Trabajo Especial de Grado titulado: **Sistema de Administración y Monitoreo para los Punto de acceso Inalámbricos de la Facultad de Ciencias de la Universidad Central de Venezuela**, presentado por el bachiller Kendall R. Navarro U., C.I.: 16.412.971, a los fines de optar por el título de Licenciado en Computación, dejan constancia de lo siguiente:

Dicho trabajo, leído por cada uno de los miembros del jurado, se fijó el día \_\_\_\_\_, a las \_\_\_\_\_, para que sus autores lo defendieran en forma pública en la Escuela de Computación, mediante una presentación oral de su contenido, luego de lo cual respondieron a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobarlo con la nota de \_\_\_\_\_ puntos.

En fe de lo cual se levanta la presente Acta, en Caracas a los veinte (20) días del mes de Octubre del año dos mil once (2011), dejando constancia de que actuó como coordinador del jurado la Profesora Ana Verónica Morales Bezeira.

---

Prof. Ana Morales  
(Tutor)

---

Prof. María E. Villapol  
(Jurado)

---

Prof. Jaime Parada  
(Jurado)

---

Prof. Robinson Ribas  
(Jurado)

## **Dedicatoria.**

A nuestros padres, por brindarnos su apoyo y estar allí en los momentos en que los necesitábamos, de manera incondicional.

A nuestros hijos, que nos dan la fuerza necesaria para continuar y ser más cada día, que con su inocencia nos inspira a continuar luchando por lo que queremos.

A nuestras compañeras,  
complementos de nuestro ser.

## **Agradecimientos.**

A Dios en primer lugar, por bendecir nuestras vidas de tantas maneras.

A nuestras madres por todo el amor y los sacrificios que han realizados para lograr que alcanzáramos este preciado logro. Sabemos que para ustedes esta meta alcanzada también es un logro para ustedes.

A nuestros padres, porque gracias a sus lecciones y enseñanzas estamos alcanzando ser mejores personas cada día.

A nuestros hermanos, porque de ustedes aprendemos cada día el valor de poder contar siempre con alguien.

A mi morenita Elianny López por apoyarme en todo este camino y estar conmigo en los momentos más difíciles de mi vida. Para ti va dedicado este logro que se que compartiremos juntos toda la vida.

A nuestros hijos que gracias a su sonrisa, amor y compañía nos llenan de energía para continuar este largo camino de éxitos.

A mi compadre Andrés “Porfi” Blanco que siempre está conmigo en las buenas y en las malas.

A mí cuñado Williams Quevedo por siempre estar cuando lo necesito.

A nuestros suegros que siempre nos brindaron su apoyo cuando los necesitamos.

A la profesora Ana Morales por brindarnos su apoyo en los momentos más difíciles de nuestra carrera, sin su apoyo este logro hubiera sido imposible. Muchísimas gracias profesora.

Muchas personas tocaron nuestras vidas para bien durante nuestro paso por la Facultad de Ciencias, a todos aquellos con quienes compartimos momentos de estudio, como también de “sano ocio”. Gracias por su tiempo y disposición.

A la ilustre Universidad Central de Venezuela, por brindarnos una educación de excelencia, tanto en lo académico como en lo humano. Y por formar parte de nuestras vidas.

## **Resumen.**

El presente Trabajo Especial de Grado propone un sistema Web que permita la administración y el monitoreo de los Puntos de Acceso que proveen acceso a Internet de manera Inalámbrica dentro de la Facultad de Ciencias de la Universidad Central de Venezuela. De esta manera, todo el proceso de administración y monitoreo pueda ser realizado por el Administrador de una forma sencilla, rápida y de manera centralizada e independiente del fabricante de los equipos.

La arquitectura usada para el desarrollo de la misma es la arquitectura Cliente/Servidor, la cual consiste en una serie de servicios almacenados en un servidor y clientes que acceden a dichos servicios.

Para el desarrollo de la aplicación, se implemento la Metodología de Desarrollo Rápido de Aplicaciones, en el cual se realizaron procesos Iterativos para depurar y mejorar la aplicación, basados en la pruebas y en los comentarios de los usuarios finales de la aplicación.

Con el uso de este conjunto de tecnologías y metodologías de desarrollo se tuvo como resultado una aplicación sencilla, con una interfaz amigable y usable, en el cual el administrador del sistema puede llevar a cabo las tareas de gestión de los dispositivos que conforman la red inalámbrica de la Facultad de Ciencias de forma centralizada.

## **Palabras Clave:**

Aplicación Web, punto de acceso, Cliente/Servidor, Desarrollo Rápido de Aplicaciones.

# Índice de Contenidos.

|  |           |
|--|-----------|
| Índice de Tablas.....  | 8         |
| Índice de Figuras.....   | 9         |
| Introducción .....   | 11        |
| <b>Capítulo I. Problema de Investigación.</b> .....  | <b>13</b> |
| 1.1 Planteamiento del Problema.....  | 13        |
| 1.2 Objetivos.....   | 14        |
| 1.2.1 Objetivo General. ....   | 14        |
| 1.2.2 Objetivos Específicos. ....  | 15        |
| 1.3 Justificación e Importancia.....   | 15        |
| 1.4 Alcance. ....  | 16        |
| 1.5 Metodología.....   | 16        |
| <b>Capítulo II. Marco Conceptual.</b> .....  | <b>18</b> |
| 2.1 Aplicaciones Web Cliente/Servidor. ....  | 18        |
| 2.1.1 Aplicaciones Web. ....   | 18        |
| 2.1.2 Arquitectura Cliente/Servidor .....  | 19        |
| 2.1.3 El Patrón de Diseño MVC (Modelo-Vista-Controlador). ....                                       | 20        |
| 2.2 Herramientas tecnológicas para el desarrollo de aplicaciones Web.....                            | 22        |
| 2.2.1 Tecnologías del lado del Cliente.....  | 23        |
| 2.2.1.1 Lenguaje de marcado de hipertexto (HTML o <i>Hiper Text Markup Language</i> ). ....          | 23        |
| 2.2.1.2 Cascading Style Spreadsheet (Hojas de Estilo en Cascada). ....                               | 24        |
| 2.2.1.3 JavaScript. ....   | 25        |
| 2.2.1.4 Ajax. ....   | 26        |
| 2.2.1.5 JQuery .....   | 28        |
| 2.2.2 Tecnologías del Lado del Servidor .....  | 28        |
| 2.2.2.1 Apache.....  | 28        |
| 2.2.2.2 PHP.....   | 29        |
| 2.2.3 Tecnologías del lado del Servidor de Bases de Datos .....                                      | 30        |
| 2.2.3.1 MySQL.....   | 30        |
| 2.3 Herramientas para el Monitoreo y Administración de Redes .....                                   | 32        |
| 2.3.1 NMAP.....  | 32        |
| 2.3.2 SNMP ( <i>Simple Network Management Protocol</i> o Protocolo Simple de Gestión de Redes). .... | 33        |
| <b>Capítulo III. Marco Metodológico</b> .....  | <b>36</b> |
| 3.1. Metodología de Desarrollo de Software .....   | 36        |
| 3.1.1 Desarrollo Rápido de Aplicaciones (RAD) .....  | 36        |
| 3.1.1.1 Características de RAD. ....   | 38        |
| 3.1.1.2 Ventajas de RAD. ....  | 39        |
| 3.1.1.2 Desventajas de RAD. ....   | 39        |
| 4.1 Proyecto. ....   | 41        |
| 4.2 Fases del Proyecto. ....   | 42        |
| 4.2.1 Fase de Planeación de Requerimientos. ....   | 42        |
| II. Proceso Autenticación (Login). ....  | 44        |
| III. Proceso Recuperar Contraseña.....   | 45        |

|           |   |           |
|-----------|---|-----------|
| IV.       | Proceso Cambiar Contraseña. ....                              | 46        |
| V.        | Proceso Administrar Usuario .....                             | 47        |
| VI.       | Proceso Agregar Usuario .....                                 | 49        |
| VII.      | Proceso ver información de Usuario.....                       | 50        |
| VIII.     | Proceso Editar Usuario.....                                   | 51        |
| IX.       | Proceso Eliminar Usuario.....                                 | 52        |
| X.        | Proceso Administrar Punto de acceso.....                      | 53        |
| XI.       | Proceso Agregar Punto de acceso. ....                         | 54        |
| XII.      | Proceso Editar información del Punto de acceso.....           | 57        |
| XIII.     | Proceso Ver información del Punto de acceso. ....             | 58        |
| XIV.      | Proceso Eliminar Punto de acceso. ....                        | 60        |
| 4.2.2     | Taller de Diseño. ....  | 60        |
|           | Requerimiento 1: Autenticación .....                          | 61        |
|           | Requerimiento 2: Creación de Usuarios.....                    | 62        |
|           | Requerimiento 3: Visualización y Eliminación de Usuarios..... | 64        |
|           | Requerimiento 4: Editar información del Usuario. ....         | 67        |
|           | Requerimiento 5: Agregar AP .....                             | 68        |
|           | Requerimiento 6: Visualizar y Eliminar Punto de acceso .....  | 70        |
|           | Requerimiento 7: Editar Punto de acceso .....                 | 71        |
|           | Requerimiento 8: Escanear Red .....                           | 72        |
| 4.3.3.    | Implementación. ....  | 74        |
| <b>V.</b> | <b>Escenarios de Prueba.....</b>                              | <b>83</b> |
|           | Escenario 1. ....   | 83        |
|           | Escenario 2. ....   | 85        |
|           | <b>Conclusiones .....</b>                                     | <b>90</b> |
|           | <b>Referencias Bibliográficas.....</b>                        | <b>92</b> |

## Índice de Tablas.

|  |    |
|--|----|
| Tabla 1. Descripción del Caso de Uso Autenticar Usuario. ....                      | 44 |
| Tabla 2. Descripción de Caso de Uso Recuperar Contraseña. ....                     | 46 |
| Tabla 3. Descripción del Caso de Uso Cambiar Contraseña.....                       | 47 |
| Tabla 4. Descripción Caso de Uso Administrar Usuario.....                          | 48 |
| Tabla 5. Descripción Caso de Uso Agregar Usuario. ....                             | 49 |
| Tabla 6. Descripción del Caso de Uso Ver información de Usuario. ....              | 50 |
| Tabla 7. Descripción del Caso de Uso Editar Usuario. ....                          | 51 |
| Tabla 8. Descripción del Caso de Uso Eliminar Usuario. ....                        | 53 |
| Tabla 9. Descripción del Caso de Uso Administrar Punto de acceso .....             | 54 |
| Tabla 10. Descripción del Caso de Uso Agregar Punto de acceso.....                 | 55 |
| Tabla 11. Descripción del Caso de Uso Editar información del Punto de acceso. .... | 57 |
| Tabla 12. Descripción Caso de Uso Ver información del Punto de acceso.....         | 59 |
| Tabla 13. Descripción del Caso de Uso Eliminar Punto de acceso. ....               | 60 |

## Índice de Figuras.

|  |    |
|--|----|
| Figura 1. Arquitectura Cliente-Servidor. ....  | 20 |
| Figura 2. Arquitectura MVC.....  | 21 |
| Figura 3. Red Administrada SNMP .....  | 34 |
| Figura 4. Fases del RAD. ....  | 37 |
| Figura 5. Modelo Entidad-Relación del modelo de datos de la aplicación.....                  | 43 |
| Figura 6. Caso de Uso Autenticar Usuario (Login).....  | 44 |
| Figura 7. Casos de Uso Recuperar Contraseña.....   | 45 |
| Figura 8. Caso de Uso Cambiar Contraseña. ....   | 47 |
| Figura 9. Caso de uso Administrar Usuario .....  | 48 |
| Figura 10. Caso de Uso Agregar Usuario .....   | 49 |
| Figura 11. Caso de Uso Ver información de Usuario. ....                                      | 50 |
| Figura 12. Caso de Uso Editar Usuario. ....  | 51 |
| Figura 13. Caso de Uso Eliminar Usuario. ....  | 52 |
| Figura 14. Caso de Uso Administrar Punto de acceso.....                                      | 53 |
| Figura 15. Caso de Uso Agregar Punto de acceso .....   | 55 |
| Figura 16. Caso de Uso Editar Información de Punto de acceso. ....                           | 57 |
| Figura 17. Caso de Uso Ver información del Punto de acceso.....                              | 58 |
| Figura 18. Caso de Uso Eliminar Punto de acceso.....   | 60 |
| Figura 19. Pantalla de Autenticación del Sistema. ....                                       | 61 |
| Figura 20. Datos almacenados en adm_tusuarios para validar el proceso de autenticación. .... | 61 |
| Figura 21. Metodo validaLogin() .....  | 62 |
| Figura 22. Pantalla que permite realizar la creación de un usuario .....                     | 63 |
| Figura 23. Método addUser() .....  | 64 |
| Figura 24. Interfaz de Visualización de Usuario .....  | 65 |
| Figura 25. Botón de eliminación de Usuario en el Sistema. ....                               | 65 |
| Figura 26. Metodo getAllUser(). ....   | 66 |
| Figura 27. Metodo deleteUser().....  | 66 |
| Figura 28. Interfaz para la Edición del Usuario.....   | 67 |
| Figura 29. Metodo editUser().....  | 68 |
| Figura 30. Ventana para agregar nuevo Punto de acceso.....                                   | 68 |
| Figura 31. Metodo addRouter(). ....  | 69 |
| Figura 32. Interfaz de Visualización de Punto de acceso. ....                                | 70 |
| Figura 33. Botón para eliminar un Punto de acceso. ....                                      | 70 |
| Figura 34. Método deleteRouter(). ....   | 71 |
| Figura 35. Metodo getAllRouter().....  | 71 |
| Figura 36. Botón “Editar Router” .....   | 72 |
| Figura 37. Método editRouter(). ....   | 72 |
| Figura 38. Interfaz de Escanear Red. ....  | 73 |
| Figura 39. Método networkScan(). ....  | 74 |
| Figura 40. Prueba de Agregar Usuario. ....   | 75 |
| Figura 41. Interfaz de visualización de Usuarios. ....                                       | 75 |
| Figura 42. Prueba de Editar Usuario. ....  | 76 |
| Figura 43. Datos Modificados para el Usuario josmaguirre. ....                               | 76 |

|   |    |
|---|----|
| Figura 44. Prueba de Eliminar Usuario.....                        | 77 |
| Figura 45. Tabla de Visualización de Usuarios.....                | 77 |
| Figura 46. Tabla adm_tusuarios con los usuarios registrados. .... | 78 |
| Figura 47. Prueba de Agregar Punto de acceso manual.....          | 78 |
| Figura 48. Interfaz de Visualización de Punto de acceso. ....     | 79 |
| Figura 49. Interfaz de Modificación de Punto de acceso.....       | 79 |
| Figura 50. Ventana de Eliminación de Punto de acceso.....         | 80 |
| Figura 51. Tabla router con los Punto de acceso registrados. .... | 80 |
| Figura 52. Rastreo de Punto de acceso. ....                       | 81 |
| Figura 53. Resultado del Rastreo del Punto de acceso.....         | 81 |
| Figura 54. Punto de acceso registrado luego del rastreo. ....     | 82 |
| Figura 55. Inicio de la prueba, Escenario 1. ....                 | 84 |
| Figura 56. Resultados del rastreo en el Escenario 1. ....         | 84 |
| Figura 57. Resultado mostrado al usuario en el Escenario 1. ....  | 85 |
| Figura 58. Inicio de la prueba en el Escenario 2. ....            | 86 |
| Figura 59. Resultados de las pruebas en el Escenario 2.....       | 86 |
| Figura 60. Resultado mostrado al usuario en el Escenario 2. ....  | 87 |
| Figura 61. Inicio de la prueba en el Escenario 3. ....            | 88 |
| Figura 62. Resultados de las pruebas en el Escenario 3.....       | 88 |
| Figura 63. Resultado mostrado al usuario en el Escenario 3. ....  | 89 |

## Introducción

Desde la creación de la red ARPANET a finales de los sesenta, el desarrollo y evolución de las redes que permiten la comunicación entre dispositivos ha tenido un enorme crecimiento. Actualmente, existen una gran cantidad de estándares, topologías y protocolos que permiten la comunicación efectiva entre dispositivos interconectados. La necesidad de realizar esta comunicación de manera sencilla llevó al origen de las redes inalámbricas, como extensión para prestar servicio en donde la instalación e implementación de una red cableada no es factible ya sea por cuestiones de costo o especificaciones de infraestructura. Este hecho se remonta a 1979 [1] cuando ingenieros de IBM en Suiza realizaron un experimento que consistía en utilizar enlaces infrarrojos para crear una red local en una fábrica. Este hecho puede considerarse el punto de partida para la evolución de la tecnología inalámbrica.

En la actualidad, la Facultad de Ciencias de la Universidad Central de Venezuela cuenta con un conjunto de aplicaciones Web que permiten realizar distintas tareas de manera sencilla, desde procesos de inscripción hasta solicitudes estudiantiles y docentes.

Uno de los problemas que se presentan en esta Facultad es la dificultad de realizar, de una manera rápida y sencilla, los procesos de monitoreo y administración de los dispositivos que conforman la red inalámbrica de la misma.

El presente Trabajo Especial de Grado consiste en utilizar las tecnologías de desarrollo de aplicaciones Web para ayudar a los administradores de la Red de la Facultad de Ciencias de la Universidad Central de Venezuela a gestionar y monitorear de una manera sencilla y centralizada los dispositivos que se encuentran en la Red actual.

El presente documento se encuentra estructurado de la siguiente manera:

En el **Capítulo I** se describe el problema de la investigación donde se presenta el contexto del problema relacionado con la gestión y monitoreo de los Punto de

acceso de la Facultad de Ciencias, así como el objetivo general y los objetivos específicos.

En el **Capítulo II** se presenta el Marco Conceptual, el cual presenta las bases en las que se fundamenta el desarrollo de este trabajo. Se presentan las aplicaciones Web Cliente/Servidor, así como también la arquitectura Cliente/Servidor. Se presentan las tecnologías que van de la mano Tanto del Cliente como del servidor, así como las herramientas usadas para el monitoreo de redes.

En el **Capítulo III** se presenta el Marco Metodológico donde se define el método de Desarrollo Rápido de Aplicaciones, su característica, las ventajas y desventajas de la misma.

En el **Capítulo IV** se presenta el Marco Aplicativo el cual especifica el conjunto de actividades realizadas así como los Casos de Uso de cada una de las funcionalidades del sistema. También se definen las Iteraciones del método usado y las pruebas realizadas para verificar el funcionamiento de la aplicación.

## **Capítulo I. Problema de Investigación.**

En el siguiente capítulo se expondrá la situación actual en la que se encuentra la Facultad de Ciencias de la Universidad de Venezuela con respecto a la red inalámbrica, se mostrara el contexto del problema, que consiste en cómo se realiza el monitoreo y administración de los punto de acceso. De igual manera se expondrán los objetivos, general y específico, la importancia y justificación de implementar un sistema centralizado de administración y monitoreo, así como también el alcance y los límites que se presentan al momento de implementar la solución al problema planteado.

### **1.1 Planteamiento del Problema.**

En la actualidad, dentro de la Facultad de Ciencias de la Universidad Central de Venezuela, se encuentran desplegados una serie de puntos de accesos inalámbricos para cubrir espacios críticos donde el acceso a la red por otros medios no es factible. El objetivo de dichos punto de acceso es apoyar las actividades académicas, docentes, de investigación y extensión.

Con el tiempo, el despliegue de estos punto de acceso ha aumentado notoriamente, por lo que se han adquirido dispositivos de diferentes fabricantes y modelos, debido a factores como el presupuesto disponible por la Universidad Central de Venezuela para tal fin, la disponibilidad del mercado venezolano o porque la adquisición de dispositivos de cierto fabricante puede ser de más provecho a la hora de cubrir ciertas necesidades.

Debido a este crecimiento y diversidad, la tarea de conocer y monitorear el estado de cada punto de acceso, así como realizar las labores de administración y monitoreo de los mismos es una tarea bastante compleja. Para realizar las tareas de monitoreo, es necesario acceder a cada punto de acceso de manera separada, lo cual requiere que el personal de soporte técnico del Centro de Computación que son los encargados de realizar dichas tareas, deban conocer toda la información de cada uno de los dispositivos, así como su ubicación física dentro de la Facultad de Ciencias, dirección IP para el acceso a la interfaz de

configuración de cada dispositivo, nombres de usuario y contraseña asignados a cada uno de estos punto de acceso. Esta situación, en la que el administrador debe estar dentro del rango de servicio de cada punto de acceso, así como conocer minuciosamente la información de cada dispositivo a monitorear, da como resultado que esta tarea se realice con lentitud y que la misma no sea agradable al administrador.

Otro problema presente es que, al momento de realizar las tareas de mantenimiento, a fin de conocer si un Punto de Acceso está en correcto funcionamiento y prestando servicio de Internet, el personal de soporte técnico del Centro de Computación debe entrar en el rango de servicio de cada punto de acceso y ver si el mismo está prestando servicio de Internet.

Otra problemática es lo heterogéneo de la red actual, ya que se observan dispositivos de distintos fabricantes y modelos, por lo que es necesario por parte del administrador, conocer de qué manera se pueden configurar y gestionar cada uno de estos dispositivos.

Debido a estos problemas, se planteó la idea de desarrollar e implementar una aplicación Web, la cual permita el monitoreo y administración de cada uno de los Punto de Acceso activos en la red de una manera centralizada, rápida y efectiva, permitiendo también la incorporación de nuevos dispositivos necesarios para ampliar la cobertura dentro de la misma, facilitando la labor general del mantenimiento de la red.

## **1.2 Objetivos.**

### **1.2.1 Objetivo General.**

Desarrollar una aplicación Web que permita administrar y monitorear de manera centralizada los Puntos de Acceso que conforman la red inalámbrica de la Facultad de Ciencias de la Universidad Central de Venezuela.

### 1.2.2 Objetivos Específicos.

- Recolectar información acerca de los dispositivos que conforman la red inalámbrica actual.
- Analizar la información recolectada acerca de los Puntos de Acceso que se encuentran actualmente en funcionamiento.
- Realizar el diseño de la aplicación Web.
- Implementar las funcionalidades contempladas dentro de la aplicación Web.
- Definir escenarios de prueba para validar el comportamiento de la aplicación.

### 1.3 Justificación e Importancia.

En la actualidad la Facultad de Ciencias cuenta con un conjunto de punto de acceso para dar cobertura a lugares donde no existe o no está disponible la red cableada, permitiendo acceso de manera inalámbrica a la red de datos de la Universidad Central de Venezuela e Internet en general a sus estudiantes, docentes, empleados y obreros.

El número de estos dispositivos se espera que crezca no solo en la Facultad de Ciencias sino en todo el campus Caracas de la Universidad Central de Venezuela, gracias a la puesta en marcha del macro proyecto de red inalámbrica para todo el campus universitario.

Debido a esto, el no contar con una interfaz común que permita administrar y monitorear el estado de estos dispositivos hace que la labor de administración de la red sea tediosa, lenta y complicada, debido que para poder conocer el estado de toda la red, se debe acceder a cada punto de acceso por separado, por tal motivo surge la necesidad de implementar una aplicación Web que permita tener información centralizada de todos los punto de acceso de la red para lograr así una administración eficiente.

Con esta aplicación la tarea de administrar y monitorear los Puntos de Acceso se podrá realizar de una manera más rápida y eficiente, incrementando la calidad del servicio que se brinda a todos los usuarios.

#### **1.4 Alcance.**

La aplicación Web propuesta debe cubrir las siguientes necesidades:

- Permitir al Administrador de la Red poder visualizar la información de los Puntos de Accesos que están en la red.
- Permitir al Administrador de Red agregar, eliminar o modificar información acerca de un punto de acceso.
- Permitir el monitoreo de los punto de acceso registrados para conocer su estado.
- Permitir la administración de cada uno de los Puntos de Acceso registrados, mediante su interfaz de configuración.
- Permitir al Administrador agregar, eliminar o modificar la información de los usuarios que maneja la aplicación, así como también los roles de los mismos.
- Realizar un rastreo a través de la red para así para determinar si existe un dispositivo nuevo en la red.
- Generar alertas en caso de que algún dispositivo administrado cambie de estado.
- Almacenar estadísticas acerca del estado de los Punto de acceso.
- Visualizar de manera grafica, los Tiempos de Respuesta de cada uno de los Punto de acceso.

#### **1.5 Metodología.**

La metodología a usar permite un desarrollo y construcción rápido de la solución planteada. Dicha metodología es llamada Desarrollo Rápido de Aplicaciones o RAD, la cual permite realizar el desarrollo del software de manera

incremental en un periodo de tiempo relativamente corto. El RAD se desarrolla en 3 fases:

- **Planeación de Requerimientos:** En esta fase se realiza el análisis de los objetivos planteados y los requerimientos necesarios para cubrir dichos objetivos.
- **Taller de Diseño:** Esta fase involucra el **diseño y construcción** de los requerimientos. Esta fase es iterativa, lo que permite que el uso de este modelo de desarrollo sea incremental.
- **Implementación:** En esta fase se realizaron todas las pruebas referentes al desarrollo de cada uno de los requerimientos en las iteraciones del taller de diseño.

Las tecnologías usadas para realizar el desarrollo de la aplicación Web son las siguientes.

- **PHP:** Hipertext Pre-Procesor o Pre-procesador de Hipertexto, es un lenguaje de programación interpretado para la creación de páginas Web.
- **Ajax:** Asynchronous JavaScript and XML o JavaScript asíncrono y XML es un lenguaje de desarrollo de aplicaciones Web enriquecidas (Rich Internet Applications).
- **JavaScript:** lenguaje de programación Web usado para el desarrollo de paginas dinámicas.
- **Apache:** es un servidor Web multiplataforma el cual se utiliza para la publicación de páginas Web HTML.

Este conjunto de tecnologías permitirán implementar el conjunto de funcionalidades definidas. El servidor Apache servirá de contenedor para la aplicación, mientras que el modulo PHP permitirá interpretar el desarrollo de las funcionalidades.

Los lenguajes Ajax y JavaScript, permitirán dar contenido dinámico a la aplicación.

## **Capítulo II. Marco Conceptual.**

En este capítulo se definirá de manera detallada las tecnologías utilizadas para el desarrollo e implementación de la solución al problema planteado. Se expondrá la arquitectura Cliente-Servidor, así como el patrón MVC (Modelo-Vista-Controlador).

Se explicaran las tecnologías que se ejecutan del lado del cliente y del lado del Servidor, así como también tecnologías que permiten el desarrollo de aplicaciones Web.

Por último se explicaran las herramientas para el monitoreo de redes y para el descubrimiento de dispositivos dentro de la misma.

### **2.1 Aplicaciones Web Cliente/Servidor.**

#### **2.1.1 Aplicaciones Web.**

Se denomina Aplicación Web a 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. [1]

Una de las ventajas de las aplicaciones Web cargadas desde Internet o Intranet es la facilidad de mantener y actualizar dichas aplicaciones sin la necesidad de distribuir e instalar un software en, potencialmente, miles de clientes. También la posibilidad de ser ejecutadas en múltiples plataformas hace a las Aplicaciones Web una opción muy atractiva a la hora del desarrollo de aplicaciones. [1]

Las Aplicaciones Web usan comúnmente una combinación de scripts del lado del servidor (ASP, PHP, etc.) y scripts del lado del cliente (HTML, JavaScript, etc.). Los scripts del lado del cliente se encargan de la presentación de la información mientras que los scripts del lado del servidor se encargan de toda la lógica utilizada para servir la información a los clientes. [1]

### **2.1.2 Arquitectura Cliente/Servidor**

El modelo arquitectónico cliente-servidor es un modelo de sistema en el que dicho sistema se organiza como un conjunto de servicios y servidores asociados, más unos clientes que acceden y usan los servicios. Los principales componentes de este modelo son:

- Un conjunto de servidores que ofrecen servicios a otros subsistemas. Ejemplos de servidores son: servidores de impresoras que ofrecen servicios de impresión, servidores de archivos que ofrecen servicios de gestión y servidores de compilación, que ofrecen servicios de compilación de lenguajes de programación.
- Un conjunto de clientes que llaman a los servicios ofrecidos por los servidores. Éstos son normalmente subsistemas en sí mismos. Puede haber varias instancias de un programa cliente ejecutándose concurrentemente.
- Una red que permite a los clientes acceder a estos servicios. Esto no es estrictamente necesario ya que los clientes y los servidores podrían ejecutarse sobre una única máquina. En la práctica, sin embargo, la mayoría de los sistemas cliente-servidor se implementan como sistemas distribuidos. [2]

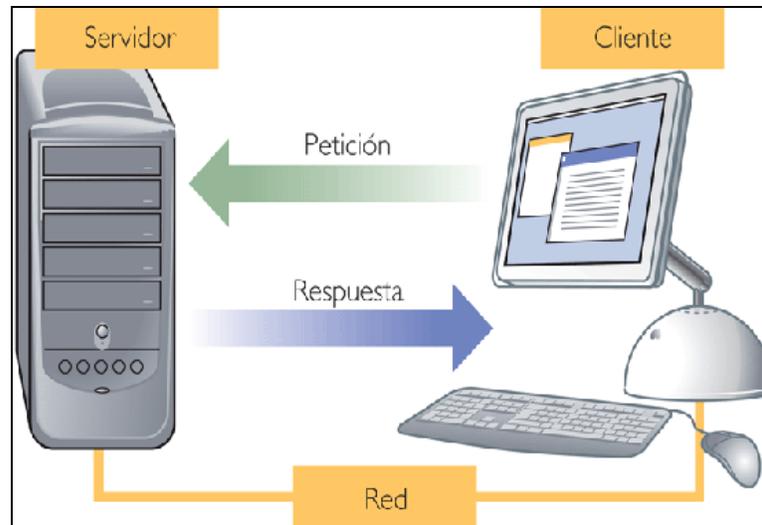


Figura 1. Arquitectura Cliente-Servidor.

En la figura 1 podemos observar un ejemplo de arquitectura Cliente/Servidor. Los clientes pueden conocer los nombres de los servidores disponibles y los servicios que éstos proporcionan. Sin embargo, los servidores no necesitan conocer la identidad de los clientes o cuántos clientes tienen. Los clientes acceden a los servicios proporcionados por un servidor a través de llamadas a procedimientos remotos usando un protocolo de petición-respuesta tal como el protocolo HTTP usado en la WWW. Básicamente, un cliente realiza una petición a un servidor y espera hasta que recibe una respuesta. [2]

### 2.1.3 El Patrón de Diseño MVC (Modelo-Vista-Controlador).

Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón de llamada y retorno MVC se ve frecuentemente en Aplicaciones Web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista. [3]

Cada vista tiene un objeto controlador asociado que maneja las entradas del usuario y la interacción de los dispositivos. Por lo tanto, un modelo que representa datos numéricos puede tener una vista que represente los datos como

un histograma y una vista que presente los datos como una tabla. El modelo se puede editar cambiando los valores en la tabla o alargando o acortando las barras en el histograma. En la figura 2 se puede observar un ejemplo de la arquitectura MVC. [3]



Figura 2. Arquitectura MVC.

El **modelo** es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Define las reglas de negocio (la funcionalidad del sistema). Un ejemplo de regla puede ser: "Si la mercancía pedida no está en el almacén, consultar el tiempo de entrega estándar del proveedor".
- Lleva un registro de las vistas y controladores del sistema.
- Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo (por ejemplo, un

fichero batch que actualiza los datos, un temporizador que desencadena una inserción, etc.).

El **controlador** es responsable de:

- Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
- Contener reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas. Una de estas peticiones a las vistas puede ser una llamada al método "Actualizar()". Una petición al modelo puede ser "Obtener\_tiempo\_de\_entrega( nueva\_orden\_de\_venta )".

Las **vistas** son responsables de:

- Recibir datos del modelo y mostrarlos al usuario.
- Mantener un registro de su controlador asociado (normalmente porque además lo instancia).
- Pueden dar el servicio de "Actualización()", para que sea invocado por el controlador o por el modelo (cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes).

## 2.2 Herramientas tecnológicas para el desarrollo de aplicaciones Web.

En un principio la Web era sencillamente una colección de páginas estáticas, documentos, etc., para su consulta o descarga. El paso inmediatamente posterior en su evolución fue la inclusión de un método para elaborar páginas dinámicas que permitieran que lo mostrado tuviese carácter dinámico (es decir, generado a partir de los datos de la petición). Este método fue conocido como CGI (*Common Gateway Interface*) y definía un mecanismo mediante el que se podía pasar información entre el servidor y ciertos programas externos. Los CGIs siguen utilizándose ampliamente; la mayoría de los servidores Web permiten su uso

debido a su sencillez. Además, dan total libertad para elegir el lenguaje de programación que se desea emplear. [4]

Con el paso del tiempo, el mundo Web ha venido evolucionando, con el desarrollo de nuevas tecnologías y lenguajes (tanto del lado del Servidor como del lado del Cliente) los cuales permiten que la interacción entre el usuario y la información sea un poco más dinámica, intuitiva y fácil de manejar. [4]

Una de las potencialidades desarrolladas fue la de Sun Microsystems con Java, al permitir incrustar código Java dentro de páginas HTML que el servidor convierte en programas ejecutables JSP (*Java Server Pages* o “Paginas de Servidor Java”), y un método de programación que se ejecuta del lado del servidor denominado Java Servlet. [4]

Una de las Tecnologías más utilizadas actualmente para el desarrollo de aplicaciones Web es el lenguaje PHP. PHP (*Hypertext Preprocesor* o Pre-procesador de Hipertexto) es un lenguaje interpretado que permite la inclusión de lenguaje HTML dentro de los mismos, con una sintaxis derivada de C y Perl. La ventaja de ser sencillo y muy potente ha contribuido a convertir al lenguaje PHP en una herramienta muy útil para el desarrollo de aplicaciones Web. [4]

## **2.2.1 Tecnologías del lado del Cliente.**

### **2.2.1.1 Lenguaje de marcado de hipertexto (HTML o *Hiper Text Markup Language*).**

HTML o Lenguaje de Marcado de hipertexto, es un estándar reconocido en todo el mundo y cuyas normas define el *World Wide Web Consortium*, más conocido como W3C. Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo. [5]

El origen de HTML se remonta a 1980, cuando el físico Tim Berners-Lee, trabajador del CERN (*Organización Europea para la Investigación Nuclear*) propuso un nuevo sistema de "hipertexto" para compartir documentos. [5]

Los sistemas de "hipertexto" habían sido desarrollados años antes. En el ámbito de la informática, el "hipertexto" permitía que los usuarios accedieran a la información relacionada con los documentos electrónicos que estaban visualizando. De cierta manera, los primitivos sistemas de "hipertexto" podrían compararse a los enlaces de las páginas Web actuales. [5]

El propio W3C<sup>1</sup> define el lenguaje HTML como "un lenguaje reconocido universalmente y que permite publicar información de forma global". Desde su creación, el lenguaje HTML ha pasado de ser un lenguaje utilizado exclusivamente para crear documentos electrónicos a ser un lenguaje que se utiliza en muchas aplicaciones electrónicas como buscadores, tiendas online y banca electrónica.

El lenguaje HTML permite a los desarrolladores:

- Publicar en línea documentos con cabeceras, textos, tablas, listas, fotos, etc.
- Obtener información vía enlaces de Hipertexto (*links*) con solo un clic.
- Construir formularios que pueden desencadenar transacciones en servicios remotos, para la búsqueda de información, hacer reservaciones, ordenar productos, etc.
- Se pueden incluir hojas de estilo, clips de video, sonido, y otras aplicaciones directamente en los documentos. [5]

### 2.2.1.2 Cascading Style Spreadsheet (Hojas de Estilo en Cascada).

CSS o en español *Hoja de Estilo en Cascada* es un lenguaje de hojas de estilo usado para describir los esquemas de presentación de un documento escrito en un lenguaje de Marcado. Es muy utilizado para las aplicaciones desarrolladas en HTML o XHTML para definir estilos, aunque también puede ser aplicado para cualquier tipo de documento XML. [6]

---

<sup>1</sup> Conocido como *World Wide Web Consortium* o Consorcio de la Red de Area Amplia, es un consorcio internacional que produce recomendaciones para la Red de Area Amplia o *World Wide Web*.

Las CSS son diseñadas para permitir la separación de lo que es el contenido del documento con la presentación del mismo, como son los paneles, colores y fuentes. Esta separación puede mejorar el acceso al contenido, proveer mayor flexibilidad y control en la especificación de las características de la presentación, permitiendo que múltiples documentos compartan el mismo formato, reduciendo la complejidad y repetición del contenido estructural. CSS puede también permitir que un mismo documento pueda ser presentado en diferentes estilos para diferentes métodos de presentación, sean en pantalla, en impresión, etc. [6]

CSS especifica un esquema de prioridad para determinar cual regla de estilo aplica cuando múltiples definiciones de estilos aplican a un mismo elemento. Por esto es llamado **cascada**, ya que las prioridades son calculadas y asignadas en reglas para que el resultado sea predecible. [6]

Las ventajas a la hora de usar CSS son muy variadas, algunas de ellas son:

- Control centralizado de la presentación de un documento Web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Los navegadores permiten a los usuarios especificar su propia hoja de estilo local, que será aplicada a un documento Web, con lo que aumenta considerablemente la accesibilidad.
- Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o, incluso, a elección del usuario.
- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño.

### **2.2.1.3 JavaScript.**

JavaScript es un lenguaje interpretado orientado a las páginas web, con una sintaxis semejante a la del lenguaje Java. [7]

El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que fabricó los primeros navegadores de Internet

comerciales. Se utiliza en páginas Web HTML, para realizar tareas y operaciones en el marco de la aplicación cliente (*client-side*). [7]

JavaScript se considera un lenguaje simple y altamente dinámico. Decir que un lenguaje de programación es dinámico no quiere decir nada desde un punto de vista formal, pero se acepta en algunos círculos que esto se refiere a que es un lenguaje de muy alto nivel, donde muchas operaciones que comúnmente suceden en tiempo de compilación, se realizan en tiempo de ejecución. Operaciones tales como el sistema de tipos, la generación y ejecución de código "al vuelo", así como extender los objetos y sus definiciones mientras se ejecuta la aplicación. [7]

Algunas de las ventajas importantes del uso de JavaScript son:

- El lenguaje de scripting es seguro y fiable porque está en claro que hay que interpretarlo, por lo que puede ser filtrado; para el mismo JavaScript, la seguridad es casi total.
- Los scripts tienen capacidad limitada, por razones de seguridad, por lo cual no es posible hacer todo con JavaScript, sino que es necesario usarlo conjuntamente con otros lenguajes evolucionados, posiblemente más seguros.
- El código JavaScript se ejecuta en el cliente por lo que el servidor no es solicitado más de lo debido; un script ejecutado en el servidor, sin embargo, sometería al mismo a dura prueba y los servidores de capacidades más limitadas podrían resentir de una continua solicitud por un mayor número de usuarios.

#### 2.2.1.4 Ajax.

AJAX (acrónimo para *Asynchronous JavaScript And XML. JavaScript y XML Asíncronos*) es un enfoque de desarrollo basado en un conjunto de tecnologías ya existentes, agrupadas para presentar información e interactuar dinámicamente, de manera asíncrona, con un servidor Web. [8]

Entre las tecnologías que agrupa AJAX se destacan las siguientes como las principales:

**HTML y CSS:** para la presentación, estructuración y formato del contenido.

**DOM (Document Object Model):** Con el modelo de objetos del documento se logra obtener la estructura del documento HTML. Utilizando esta estructura se pueden agregar, eliminar y modificar, de manera dinámica, elementos de la página mediante el uso de la tecnología JavaScript.

**XML:** Para el intercambio de datos entre el cliente (navegador Web) y el servidor.

**JavaScript:** Mediante esta tecnología del lado del cliente se realizan las peticiones de manera asíncrona y, junto con el manejo del DOM, se logra la interacción dinámica con el usuario.

Con AJAX se busca entonces proveer de una mayor riqueza de interacción entre el usuario y la aplicación en comparación con la que proveen las aplicaciones Web tradicionales. Este enfoque cambia el esquema tradicional de interacción con las aplicaciones Web, procurando una mayor velocidad de respuesta hacia el usuario, permitiéndole mantener la interacción con la aplicación inclusive durante tiempos de procesamiento, con elementos de interfaz más dinámicos e interactivos. [8]

El enfoque AJAX permite el envío de peticiones al servidor en segundo plano, es decir, sin interrumpir la interacción entre el usuario y la aplicación. Esto disminuye el típico tiempo de espera entre peticiones y permite mantener la continuidad en la interacción entre el usuario y la aplicación, lo que le permite al usuario realizar acciones como la modificación de campos de un formulario, despliegue de menús, visualización de datos, entre otros. [8]

Luego de enviar una petición y obtener la información requerida (respuesta) del servidor, las aplicaciones AJAX tienen la capacidad de modificar la vista (estructura de la página) dinámicamente sin necesidad de solicitar una página distinta. Esta información obtenida del servidor está constituida típicamente de

datos en formato XML o texto plano, no es necesario recibir en cada petición un documento HTML entero, ya que la aplicación AJAX utiliza estos datos para modificar la página sin necesidad de cambiarla por una nueva. [8]

### **2.2.1.5 JQuery**

JQuery es un biblioteca o framework de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. [9]

JQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio. [9]

Las principales características de JQuery son:

- Se pueden seleccionar elementos del árbol DOM mediante funciones.
- Modificaciones transversales del árbol DOM.
- Manipulación de Hojas de Estilo.
- Se pueden dar efectos y animaciones a los elementos que conforman el documento.
- Es extensible.

## **2.2.2 Tecnologías del Lado del Servidor**

### **2.2.2.1 Apache**

Apache es un servidor Web HTTP de código abierto que garantiza la disponibilidad de la mayoría de los sitios web activos actualmente. El servidor está diseñado para operar sobre los más populares plataformas y sistemas operativos como Linux, Unix, Windows; Solaris, Novell, FreeBSD, Mac OS, OS/2, etc. [10]

Existen dos versiones populares acerca del origen del nombre del proyecto. La primera y más aceptada, que el nombre del servidor fue escogido por el nombre de la tribu Nativa Americana APACHE, muy conocidos por su resistencia y habilidades militares. La segunda versión da el nombre por el grupo de parches agregados al núcleo del servidor HTTP NCSA 1.3, lo cual hace al mismo un “servidor parcheado” (en inglés “*a patchy server*”). [10]

El servidor fue desarrollado por la comunidad de fuente abierta Apache Software Foundation, en donde sus miembros están constantemente añadiendo funcionalidades, con el único propósito de proveer una plataforma segura y extensible que mejore el servicio HTTP de la mano de los estándares HTTP actuales. [10]

Las ventajas más conocidas del Servidor Apache son:

- Modular
- De código abierto
- Multi-plataforma
- Extensible

#### 2.2.2.2 PHP

*PHP* es un lenguaje de programación interpretado de código abierto y alto nivel usado generalmente para la creación de contenido dinámico para sitios Web, el cual puede ser embebido dentro de páginas HTML, es interpretado del lado del servidor. *PHP* es el acrónimo recursivo de “*PHP: Hypertext Preprocessor*”, (inicialmente se llamó *PHP Tools* o *Personal Home Page Tools*) últimamente también es usado para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica. [11]

La facilidad de uso que caracteriza a PHP en conjunción con la similitud que tiene con lenguajes más comunes de programación estructurada, como C, Java y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje bastante suave. También

les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas. [11]

PHP tiene la capacidad de ser ejecutado en los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes de Unix (incluyendo HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS, entre otros. PHP soporta la mayoría de servidores Web de hoy en día, incluyendo Apache, Microsoft Internet Information Server (IIS), Personal Web Server, Netscape e iPlanet, O'Reilly Website Pro Server, Caudium, Xitami, OmniHTTPd y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede ser usado como un procesador CGI. [11]

Con *PHP* se puede hacer cualquier cosa que se pueda hacer con un *script CGI*, como procesar la información proveniente de formularios, generar páginas con contenidos dinámicos, o enviar y recibir *cookies*, entre otras cosas. [11]

### 2.2.3 Tecnologías del lado del Servidor de Bases de Datos

#### 2.2.3.1 MySQL

MySQL es un servidor de Base de Datos SQL (*Structured Query Language o Lenguaje Estructurado de Consultas*) rápido, multi-hilo, multi-usuario y robusto. El servidor MySQL está diseñado para entornos de producción críticos, con alta carga de trabajo así como para integrarse en software para ser distribuido. MySQL es una marca registrada de MySQL AB. [12]

El software MySQL tiene una doble licencia. Los usuarios pueden elegir entre usar el software MySQL como un producto Open Source bajo los términos de la licencia GNU *General Public License* o pueden adquirir una licencia comercial estándar de MySQL AB. [12]

La siguiente lista describe algunas de las características más importantes del software de base de datos MySQL:

- Portabilidad.

- Desarrollado en C y en C++.
- Probado con un amplio rango de compiladores diferentes.
- Funciona en diferentes plataformas (AIX, FreeBSD, HP-UX , GNU/Linux, Mac OS X, Novell Netware , OpenBSD, OS/2, Solaris, Windows).
- Usa GNU Automake, Autoconf, y Libtool para portabilidad.
- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- Uso completo de multi-hilos mediante hilos del kernel. Pueden usarse fácilmente múltiple CPUs si están disponibles.
- Proporciona sistemas de almacenamientos transaccionales y no transaccionales.
- Usa tablas en disco B-tree (MyISAM<sup>2</sup>) muy rápidas con compresión de índice.
- Relativamente sencillo de añadir otro sistema de almacenamiento. Esto es útil si desea añadir una interfaz SQL para una base de datos propia.
- Un sistema de reserva de memoria muy rápido basado en hilos.
- Joins muy rápidos usando un multi-join de un paso optimizado.
- Tablas hash en memoria, que son usadas como tablas temporales.
- Las funciones SQL están implementadas usando una librería altamente optimizada y deben ser tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización para consultas.
- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

---

<sup>2</sup> MyISAM es una tecnología usada para el almacenamiento de datos, la cual permite un procesamiento rápido de consultas

## 2.3 Herramientas para el Monitoreo y Administración de Redes

### 2.3.1 NMAP

NMAP (*Network Mapper* o *Mapeador de Redes*) es una herramienta de código abierto para exploración de red y auditoría de seguridad. Se diseñó para analizar rápidamente grandes redes, aunque funciona muy bien contra equipos individuales. NMAP utiliza paquetes IP "crudos" en formatos originales para determinar qué equipos se encuentran disponibles en una red, qué servicios (nombre y versión de la aplicación) ofrecen, qué sistemas operativos (y sus versiones) ejecutan, qué tipo de filtros de paquetes o cortafuegos se están utilizando así como docenas de otras características. Aunque generalmente se utiliza NMAP en auditorías de seguridad, muchos administradores de redes y sistemas lo encuentran útil para realizar tareas rutinarias, como puede ser el inventariado de la red, la planificación de actualización de servicios y la monitorización del tiempo que los equipos o servicios se mantiene activos. [13]

La salida de NMAP es un listado de objetivos analizados, con información adicional para cada uno dependiente de las opciones utilizadas. La información primordial es la "tabla de puertos interesantes". Dicha tabla lista el número de puerto y protocolo, el nombre más común del servicio, y su estado. El estado puede ser **open** (*abierto*), **filtered** (*filtrado*), **closed** (*cerrado*), o **unfiltered** (*no filtrado*). [13]

**Abierto** significa que la aplicación en la máquina destino se encuentra esperando conexiones o paquetes en ese puerto. [13]

**Filtrado** indica que un cortafuego, filtro, u otro obstáculo en la red está bloqueando el acceso a ese puerto, por lo que NMAP no puede saber si se encuentra abierto o cerrado. [13]

Los puertos **cerrados** no tienen ninguna aplicación escuchando en los mismos, aunque podrían abrirse en cualquier momento. Los clasificados como **no filtrados** son aquellos que responden a los sondeos de NMAP, pero para los que NMAP no puede determinar si se encuentran abiertos o cerrados. [13]

NMAP informa de las combinaciones de estado **open|filtered** y **closed|filtered** cuando no puede determinar en cuál de los dos estados está un puerto. La tabla de puertos también puede incluir detalles de la versión de la aplicación cuando se ha solicitado detección de versiones. NMAP ofrece información de los protocolos IP soportados, en vez de puertos abiertos. [13]

Además de la tabla de puertos interesantes, NMAP puede dar información adicional sobre los objetivos, incluyendo el nombre de DNS según la resolución inversa de la IP, un listado de sistemas operativos posibles, los tipos de dispositivo, y direcciones MAC. [13]

### **2.3.2 SNMP (*Simple Network Management Protocol* o *Protocolo Simple de Gestión de Redes*).**

El protocolo de administración SNMP es un protocolo a nivel de capa de aplicación que facilita el intercambio y la administración entre dispositivos de red. Es parte de la suite del Protocolo de Control de Transmisión/Protocolo de Internet (TCP/IP). SNMP permite a los administradores de red gestionar el funcionamiento de la red, encontrar y resolver problemas de red, así como planificar la expansión de la misma. [14]

Existen tres versiones del protocolo: SNMP version1 (SNMPv1), SNMP versión 2 (SNMPv2) y SNMP versión 3 (SNMPv3). Estas versiones poseen un conjunto de características en común, pero SNMPv2 ofrece mejoras, como operaciones adicionales del protocolo y SNMPv3 introduce el modelo de seguridad basado en el usuario (User - Based Security Model) para mensajes de seguridad y el modelo de control de acceso basado en la vista (View - Based Access Control Model) para el control de acceso. [14]

Una red administrada por SNMP consiste en tres componentes claves: dispositivos administrados, agentes y sistemas de administración de red (Network Management System) como se puede observar en la figura 3. [14]

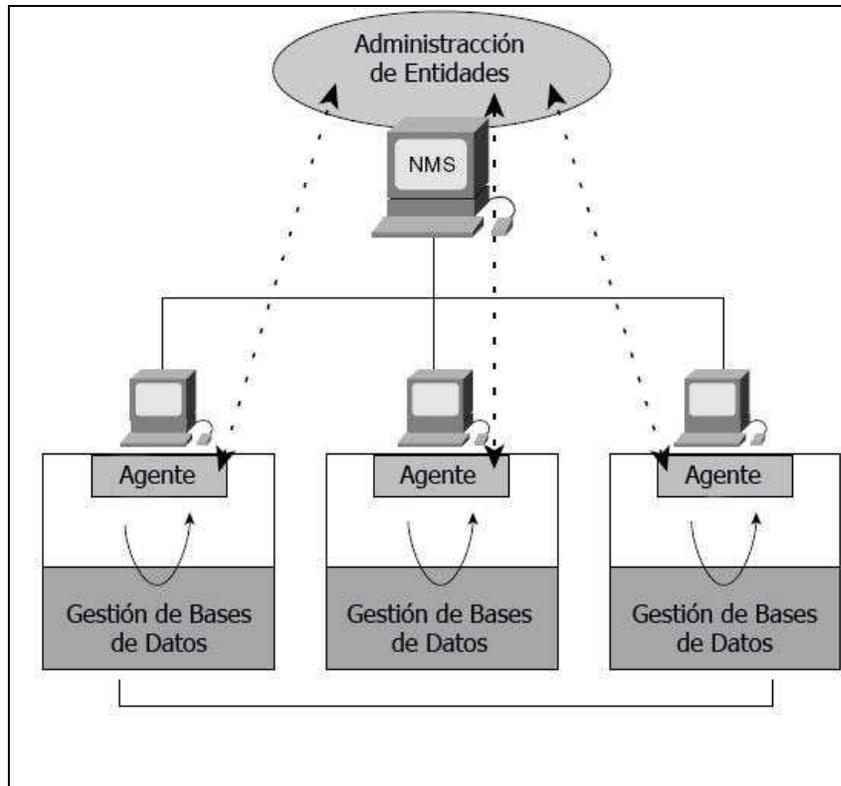


Figura 3. Red Administrada SNMP

- Un dispositivo administrado es un nodo de red que contiene un agente SNMP y este reside en una red administrada. Los dispositivos administrados almacenan información de administración, la cual es utilizada por los sistemas de administración de la red usando SNMP. Los dispositivos administrados, algunas veces hacen llamadas a elementos de la red (routers, servidores de acceso, switches, puentes, hubs, hosts o impresoras).
- Un agente SNMP es un software de administración de red que reside en un dispositivo administrado. Un agente SNMP posee un conocimiento local de la información de administración y transforma esa información en una forma compatible con el protocolo SNMP.
- Un sistema de administración de red ejecuta aplicaciones que controlan y monitorean a los dispositivos administrados. Los sistemas de administración de red proveen el conjunto de recursos de procesamiento y

memoria requeridos para la administración de la red. Uno o más de estos sistemas deben existir sobre cualquier red administrada.

## Capítulo III. Marco Metodológico

En el siguiente capítulo se explicara la Metodología usada para el Desarrollo de la Aplicación.

Se definirá el Modelo de Desarrollo Rápido de Aplicaciones, el cual es el proceso usado para el desarrollo e implementación de la solución al problema planteado. Se expondrán las fases que forman parte del proceso así como también las principales ventajas y desventajas del uso de este modelo de desarrollo.

### 3.1. Metodología de Desarrollo de Software

#### 3.1.1 Desarrollo Rápido de Aplicaciones (RAD)

El desarrollo rápido de aplicaciones o RAD (*Rapid Application Development*) es un proceso de desarrollo de software, desarrollado inicialmente por James Martin en 1980. El método comprende el desarrollo iterativo, la construcción de prototipos y el uso de utilidades CASE (Computer-Aided Systems Engineering). Tradicionalmente, el desarrollo rápido de aplicaciones tiende a englobar también la usabilidad, utilidad y la rapidez de ejecución. El Desarrollo Rápido de Aplicaciones (*Rapid Application Development* o *RAD*) es un modelo de proceso del desarrollo del software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. DRA es una adaptación a "Alta velocidad" en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes. Si se comprenden bien los requisitos y se limita el ámbito del proyecto, el proceso DRA permite al equipo de desarrollo crear un "sistema completamente funcional" dentro de periodos cortos de tiempo. Cuando se utiliza principalmente para aplicaciones de sistemas de información, el enfoque DRA comprende de las siguientes etapas, como se muestra en la figura 4. [2]



Figura 4. Fases del RAD.

- **Etapa de planificación de los requerimientos:** Esta etapa requiere que usuarios con un vasto conocimiento de los procesos que intervienen, para determinar cuáles serán las funciones del sistema. Por lo general esta etapa se completa rápidamente cuando se crean equipos que envuelven usuarios con un conocimiento amplio sobre las necesidades. La planificación de los requisitos se da en modalidad de taller conocido como Junta de Planificación de Requisitos (JRP por sus siglas en inglés). [15]
- **Etapa de diseño:** Esta consiste de un análisis detallado de las actividades de la compañía en relación al sistema propuesto. Los usuarios participan activamente en talleres bajo la tutela de profesionales de la informática. En ellos descomponen funciones y definen entidades asociadas con el sistema. Una vez completado el análisis se crean los diagramas que definen las alteraciones entre los procesos y la data. Al finalizar el análisis se traza el diseño del sistema. Se desarrollan los procedimientos y los esquemas de pantallas. Los prototipos de procedimientos críticos se construyen y se repasan y el plan para implementar el sistema se prepara. [15]
- **Etapa de Construcción:** En la etapa de construcción el equipo de desarrolladores trabajando de cerca con los usuarios finalizan el diseño y la construcción del sistema. La construcción de la aplicación consiste de una serie de pasos donde los usuarios tienen la oportunidad de afirmar los

requisitos y repasar los resultados. Las pruebas al sistema se llevan a cabo durante esta etapa. También se crea la documentación y las instrucciones necesarias para manejar la nueva aplicación, rutinas y procedimientos para operar el sistema. [15]

- **Etapa de Implementación:** Esta etapa envuelve la implementación del nuevo producto y el manejo del cambio del viejo al nuevo sistema. Se hacen pruebas comprensivas y se adiestran los usuarios. Los cambios organizacionales y la operación del nuevo sistema se hacen en paralelo con el viejo sistema hasta que el nuevo se establezca completamente. [15]

#### 3.1.1.1 Características de RAD.

**Bajos costos:** RAD, por lo general, resulta en costos más bajos. Esto se debe a que se forman pequeños equipos de profesionales quienes utilizan herramientas de alta capacidad para generar los sistemas. Estas herramientas conocidas como CASE (*Computer-Aided Systems Engineering* o Ingeniería de Sistemas Asistidos por Computador) permiten que se aligere el proceso, lo cual ayuda a que los costos aún sean más bajos. El método RAD utiliza estas herramientas computadorizadas y talento humano para cumplir con las metas requeridas rápida y efectivamente. [15]

Las herramientas integradas "CASE" proveen herramientas para que la planificación, análisis e itinerarios se creen gráficamente. Los analistas de sistemas interactúan con estas herramientas por medio de diagramas.

El propósito de las herramientas "CASE" es aligerar el proceso de diseño y a su vez disminuir los costos de desarrollo sin sacrificar la calidad del producto.

**Calidad:** La calidad de un sistema se mide en términos de hasta qué punto ese sistema cumple con los requisitos de la compañía y sus usuarios al momento que se implementa. El uso de herramientas "CASE" tiene el propósito de integrar diagramas para representar la información y crear modelos del sistema. Se crean diseños y estructuras bien detalladas. Cuando es apropiado, los diagramas ayudan

a visualizar los conceptos. Estas herramientas computarizadas refuerzan la exactitud de los diagramas. [15]

Las herramientas "CASE" junto con generadores de códigos y otros instrumentos para crear prototipos proveen un medio para asegurar la calidad del producto cuando se emplean utilizando la metodología adecuada. Un término apropiado para definir la calidad de una aplicación desarrollada con el modelo RAD es satisfacer los requisitos de los usuarios lo más eficazmente posible al momento que el sistema se implementa. Mientras menos tiempo transcurre en el desarrollo del sistema menos habrán cambiado las necesidades de los usuarios.

En compañías donde se ha utilizado el método tradicional de diseño de aplicaciones, al momento de instalar el sistema ha pasado tanto tiempo que las funciones definidas por los usuarios al comienzo del desarrollo han cambiado. Este significa volver a emplear tiempo y recursos humanos en modificar esos cambios lo que resulta en una pobre calidad del producto.

#### **3.1.1.2 Ventajas de RAD.**

- Los entregables pueden ser fácilmente trasladados a otra plataforma.
- El desarrollo se realiza a un nivel de abstracción mayor.
- Visibilidad temprana.
- Mayor flexibilidad.
- Menor codificación manual.
- Involucra más a los usuarios.
- Posiblemente menos fallas.
- Posiblemente menor costo.
- Ciclos de desarrollo más pequeños.
- Interfaz gráfica estándar.

#### **3.1.1.2 Desventajas de RAD.**

- Solo se puede aplicar si el sistema se puede modularizar de forma que permita completarse cada una de las funciones principales en menos de tres meses.
- Para proyectos grandes puede requerir muchos equipos de trabajo distintos.
- Requiere clientes y desarrolladores comprometidos en las rápidas actividades necesarias.
- No resulta adecuado cuando los riesgos técnicos son elevados.
- Se pueden tener problemas con la aceptación del prototipo.

## **Capítulo IV. Marco Aplicativo.**

En este capítulo se explica de manera detallada como se realizó el desarrollo del proyecto.

Las fases de la metodología de Desarrollo Rápido de Aplicaciones son definidas y explicadas detalladamente, la Fase de Planeación de Requerimientos, cada una de las Iteraciones del Taller de Diseño así como las pruebas de Implementación del sistema.

### **4.1 Proyecto.**

El proyecto consiste en el desarrollo e implementación de una aplicación Web centralizada para la administración y monitoreo de los Punto de acceso de la Facultad de Ciencias de la universidad Central de Venezuela. Dicha aplicación está desarrollada bajo el lenguaje PHP sobre el framework de desarrollo MVC. Además, se utilizó MySQL como Sistema Manejador de Bases de Datos gracias a la rapidez y sencillez de su funcionamiento para el almacenamiento de los datos y metadatos de dicha aplicación.

Para la realización de este proyecto se realizó una adaptación del método de desarrollo RAD, por lo que se dividió en tres fases: Planeación de requerimientos, Taller de Diseño (que constituye el diseño y la construcción de la aplicación) y fase de implementación.

En la primera fase se desarrollaron los Diagramas de Casos de Uso de los requerimientos planteados por los usuarios pertenecientes al Centro de Computación de la Facultad de Ciencias de la Universidad Central de Venezuela, que son los encargados de realizar los procesos de monitoreo y gestión de dichos Punto de acceso.

En la segunda fase, se realizó la construcción de cada uno de los procesos a nivel de código, realizando iteraciones dependiendo de cada requerimiento en específico, creando así un desarrollo rápido de la aplicación.

En la tercera fase, se realizó un grupo de pruebas para determinar la eficiencia y el feedback con el usuario final de la aplicación, a fin de conocer si la misma cumple con el funcionamiento esperado.

## **4.2 Fases del Proyecto.**

### **4.2.1 Fase de Planeación de Requerimientos.**

En esta fase se muestran el modelo de datos definido en base a los procesos que intervienen, una breve descripción de los mismos, algunos problemas que se encontraron y el análisis de la solución hallada mediante el uso de Diagramas de Casos de Uso.

#### **I. Modelo de datos**

El modelo de datos está definido por medio del modelo entidad-relación presente en la figura 5. Allí se pueden observar las estructuras definidas para el manejo de los datos que maneja la aplicación Web.

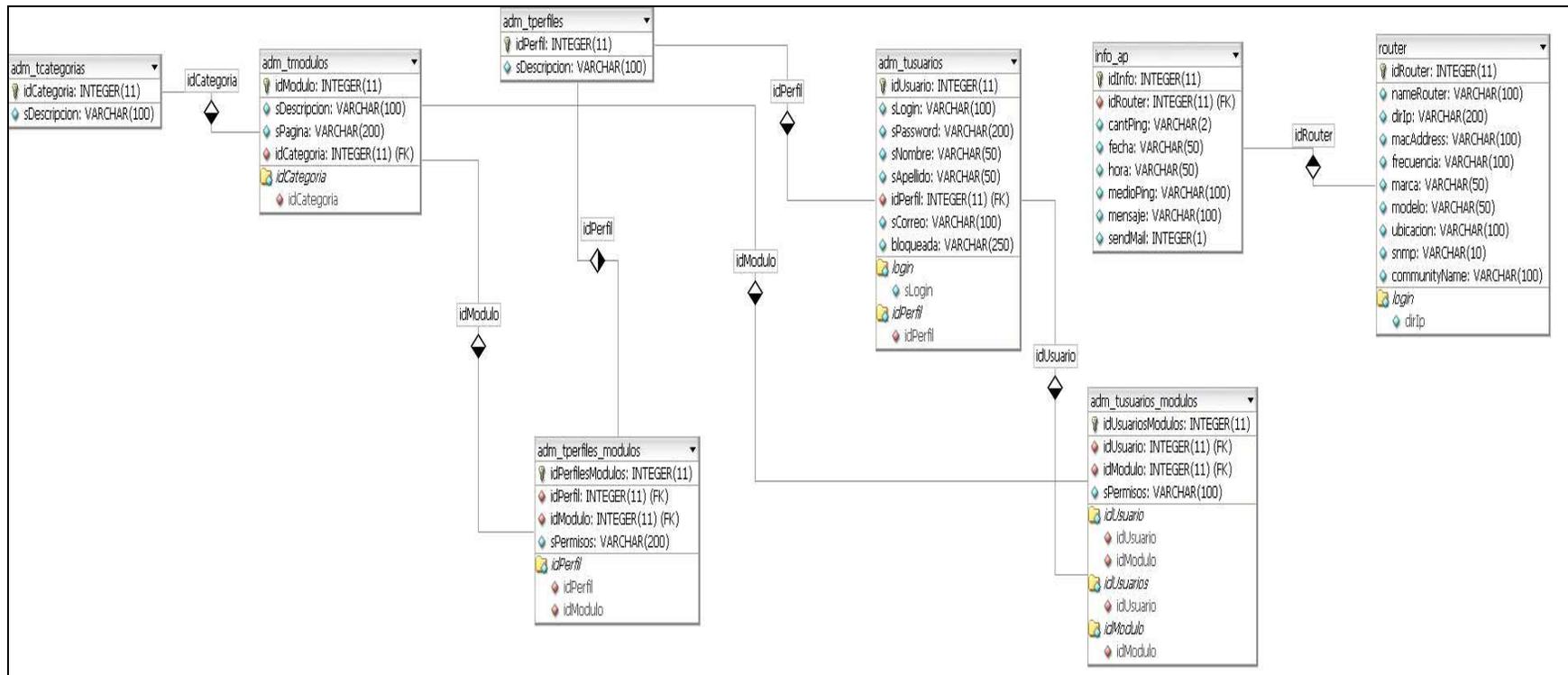


Figura 5. Modelo Entidad-Relación del modelo de datos de la aplicación.

## II. Proceso Autenticación (Login).

**Usuario involucrado:** Administrador del Sistema.

Este proceso es el punto de partida para el uso del sistema. Todo usuario que necesite realizar un monitoreo o en su defecto la administración de algún dispositivo mediante la aplicación debe estar registrado como un usuario en la misma. Este usuario será el usado para poder ingresar a la misma y poder tomar parte en todos los procesos desarrollados en la aplicación.

Cabe destacar que cada usuario en la aplicación posee una información personal asociada, así como también un rol, el cual le permitirá realizar ciertas funciones dentro de la aplicación.

Para llevar a cabo dicho requerimiento se planteo una solución que está representada en la Figura 6.

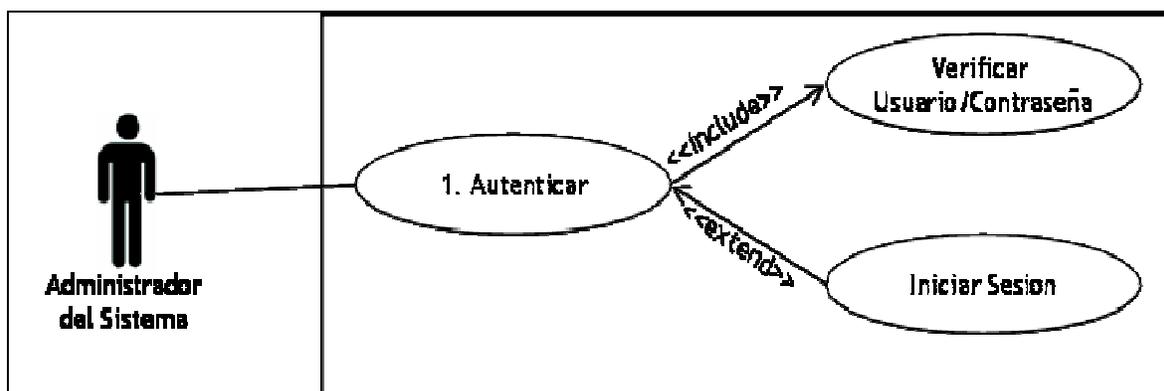


Figura 6. Caso de Uso Autenticar Usuario (Login)

Tabla 1. Descripción del Caso de Uso Autenticar Usuario.

|   |  |
|---|--|
| <b>Nombre: Autenticar Usuario</b>           |  |
| Actor                                       | Administrador del Sistema  |
| Descripción                                 | Permite realizar la autenticación del usuario. El usuario introduce el Nombre de usuario y la contraseña para acceder al sistema |
| Pre-condición                               | No aplica.   |
| Post-condición                              | En el momento de que se realiza la autenticación se verifica si el usuario es válido.  |
| <b>Nombre: Verificar Usuario/Contraseña</b> |  |
| Actor                                       | Administrador del Sistema.   |
| Descripción                                 | Verifica si la información suministrada por el usuario es válida.  |

|                |  |
|----------------|--|
| Pre-condición  | No aplica.   |
| Post-condición | Si el usuario es válido se inicia la sesión, en caso contrario se informa al usuario que la información suministrada no es válida. |

Es importante aclarar que para poder realizar una autenticación exitosa en la aplicación, el usuario debe estar previamente registrado en la misma. Este registro lo realizan los administradores del sistema, los cuales tienen privilegios para la creación de nuevos usuarios.

### III. Proceso Recuperar Contraseña.

**Usuario involucrado:** Administrador del Sistema.

Este proceso que se describe en la figura 7 se realiza cuando el usuario desea recuperar la contraseña con la que accede al sistema. Para recuperar la contraseña, se verifica si el correo suministrado por el usuario que hace la petición coincide con los que se encuentran registrados en la base de datos del sistema, para así entonces crear una nueva clave temporal y enviar la misma al correo del usuario.

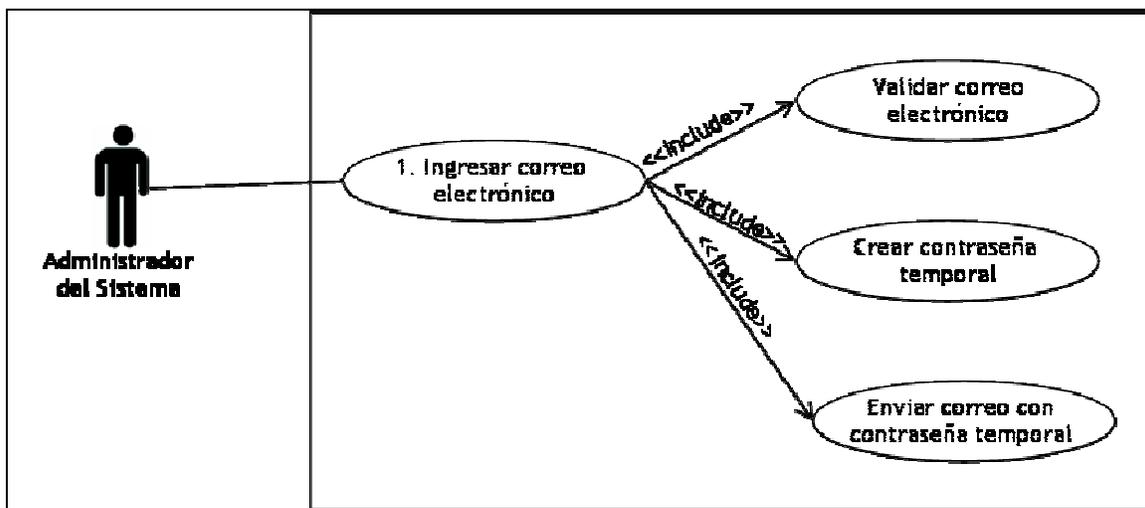


Figura 7. Casos de Uso Recuperar Contraseña.

En la tabla 2 se puede observar la descripción del Caso de Uso Recuperar Contraseña:

Tabla 2. Descripción de Caso de Uso Recuperar Contraseña.

|  |   |
|--|---|
| <b>Nombre: Ingresar correo electrónico</b>                   |   |
| Actor  | Administrador del Sistema   |
| Descripción  | Permite que un usuario ingrese su cuenta de correo electrónico asociada a su usuario para realizar la recuperación de la contraseña.  |
| Pre-condición  | No aplica.  |
| Post-condición   | No aplica.  |
| <b>Nombre: Validar correo electrónico</b>                    |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Verifica que el correo electrónico Suministrado por el usuario se encuentre en el sistema.  |
| Pre-condición  | No aplica.  |
| Post-condición   | Si el correo electrónico es Valido se realiza el proceso de creación de clave temporal, en caso contrario, se informa al usuario que el correo electrónico suministrado no es válido. |
| <b>Nombre: Crear contraseña temporal</b>                     |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Se crea una contraseña temporal para el usuario asociado al correo electrónico suministrado.  |
| Pre-condición  | La cuenta de correo suministrada debe estar asociada a un usuario.  |
| Post-condición   | No aplica.  |
| <b>Nombre: Enviar Correo electrónico con clave temporal.</b> |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Se envía al usuario un correo electrónico con la nueva clave temporal.  |
| Pre-condición  | No aplica.  |
| Post-condición   | No aplica.  |

#### IV. Proceso Cambiar Contraseña.

**Usuarios involucrados:** Administrador de Sistema.

Este proceso, el cual describe la figura 8, permite a un usuario registrado en el sistema poder cambiar su contraseña. Para poder realizar el cambio de contraseña se requiere colocar la contraseña anterior como requisito para realizar un cambio efectivo. Si esto se cumple se realiza el cambio satisfactorio de la contraseña de usuario y dicha información es registrada.

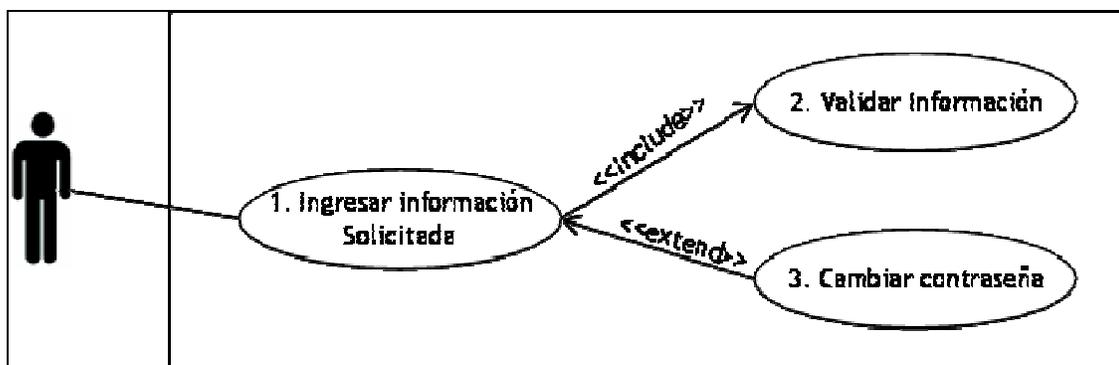


Figura 8. Caso de Uso Cambiar Contraseña.

La descripción de los pasos que se realizan en este proceso se puede observar en la tabla 3:

Tabla 3. Descripción del Caso de Uso Cambiar Contraseña.

|  |   |
|--|---|
| <b>Nombre: Ingresar información solicitada</b> |   |
| Actor  | Administrador del Sistema   |
| Descripción                                    | Permite ingresar la contraseña actual y la nueva contraseña para realizar el cambio |
| Pre-condición                                  | No aplica.  |
| Post-condición                                 | No aplica.  |
| <b>Nombre: Validar información</b>             |   |
| Actor  | Administrador del Sistema.  |
| Descripción                                    | Verifica que la información suministrada por el usuario es válida.                  |
| Pre-condición                                  | No aplica.  |
| Post-condición                                 | Si la información es válida se procede a el cambio de contraseña                    |
| <b>Nombre: Cambiar contraseña</b>              |   |
| Actor  | Administrador del Sistema.  |
| Descripción                                    | Se procede a modificar la contraseña asociada al usuario.                           |
| Pre-condición                                  | No aplica.  |
| Post-condición                                 | No aplica.  |

## V. Proceso Administrar Usuario

### Usuarios Involucrados: Administrador del Sistema

Por medio de este proceso el administrador del Sistema puede controlar todas las funciones básicas de los usuarios dentro de la aplicación. Por medio de estos procesos se podrá crear, modificar, visualizar y eliminar cualquier usuario en el sistema. La siguiente figura (figura 9) y posteriormente la tabla (tabla 4)

muestran el modelado y la descripción de los pasos que intervienen en este proceso:

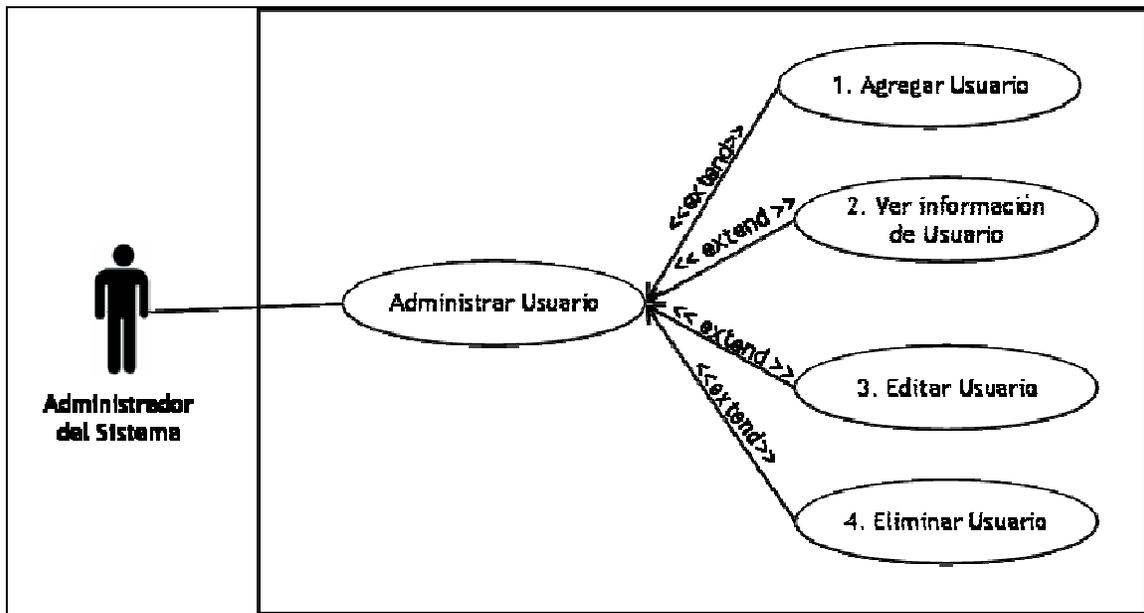


Figura 9. Caso de uso Administrar Usuario

Tabla 4. Descripción Caso de Uso Administrar Usuario

|   |  |
|---|--|
| <b>Nombre: Agregar Usuario</b>            |  |
| Actor                                     | Administrador del Sistema  |
| Descripción                               | Permite al administrador agregar un usuario en el sistema                                      |
| Pre-condición                             | Debe estar autenticado en la aplicación.   |
| Post-condición                            | No aplica.   |
| <b>Nombre: Ver información de Usuario</b> |  |
| Actor                                     | Administrador del Sistema.   |
| Descripción                               | Permite al administrador ver toda la información del usuario.                                  |
| Pre-condición                             | Debe estar autenticado en el sistema.  |
| Post-condición                            | No aplica.   |
| <b>Nombre: Editar Usuario</b>             |  |
| Actor                                     | Administrador del Sistema.   |
| Descripción                               | Permite al administrador poder modificar la información de un usuario existente en el sistema. |
| Pre-condición                             | Debe estar autenticado en el sistema.  |
| Post-condición                            | No aplica.   |
| <b>Nombre: Eliminar Usuario</b>           |  |
| Actor                                     | Administrador del Sistema.   |
| Descripción                               | Permite al administrador eliminar toda la información asociada a un usuario                    |
| Pre-condición                             | Debe estar autenticado en el sistema.  |

|                |            |
|----------------|------------|
| Post-condición | No aplica. |
|----------------|------------|

## VI. Proceso Agregar Usuario

**Usuarios involucrados:** Administrador del Sistema

Este proceso permite la creación de un usuario en el sistema. El administrador deberá suministrar cierta información de la persona a la cual se le asignara dicho usuario y se validará si el usuario que se desea registrar es válido (no está registrado ya en la aplicación por ejemplo). Si dicho usuario no está registrado entonces se procederá a la creación de la cuenta asociada al mismo. La siguiente figura (figura 10) se puede observar el Modelo de Caso de Uso de este proceso:

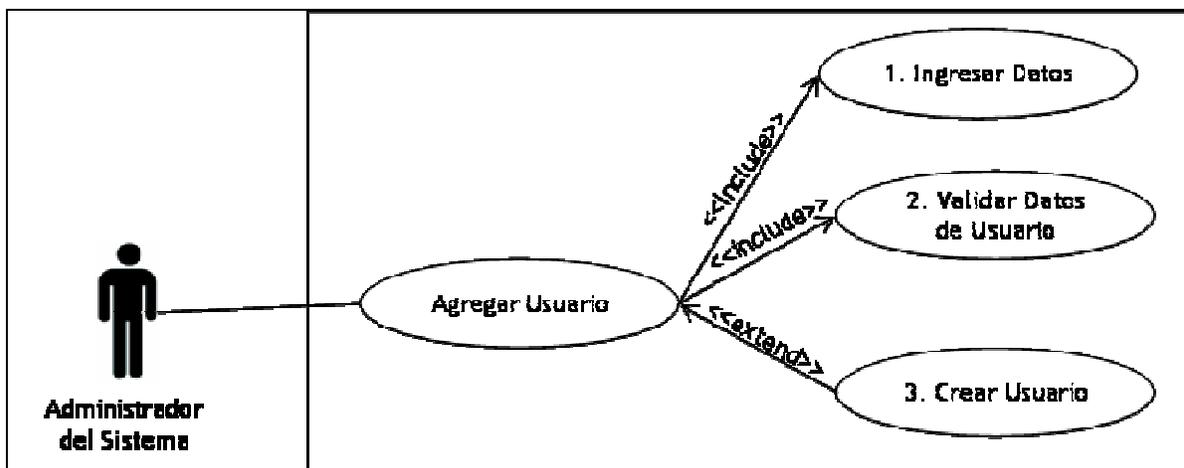


Figura 10. Caso de Uso Agregar Usuario

La descripción de dicho Caso de Uso se puede observar en la siguiente tabla (tabla 5):

Tabla 5. Descripción Caso de Uso Agregar Usuario.

|   |  |
|---|--|
| <b>Nombre: Ingresar Datos</b>           |  |
| Actor                                   | Administrador del Sistema  |
| Descripción                             | Se presenta un formulario donde el administrador del sistema podrá introducir la información asociada a la cuenta de usuario a crear |
| Pre-condición                           | Debe estar autenticado en la aplicación.   |
| Post-condición                          | Se valida la información suministrada.   |
| <b>Nombre: Validar Datos de Usuario</b> |  |
| Actor                                   | Administrador del Sistema.   |
| Descripción                             | El sistema valida la información suministrada por el administrador   |
| Pre-condición                           | Debe estar autenticado en el sistema.  |

|                              |   |
|------------------------------|---|
| Post-condición               | Si la información es válida se procede a crear la cuenta, de lo contrario se muestra al administrador que campos suministrados son inválidos. |
| <b>Nombre: Crear Usuario</b> |   |
| Actor                        | Administrador del Sistema.  |
| Descripción                  | Toda la información suministrada por el administrador es registrada y se crea una nueva cuenta de usuario.                                    |
| Pre-condición                | La información suministrada por el administrador debe ser valida  |
| Post-condición               | No aplica.  |

## VII. Proceso ver información de Usuario.

**Usuarios involucrados:** Administrador del Sistema.

Este proceso permite al administrador visualizar la información de los usuarios registrados en la aplicación. Para ello, se tendrá una lista de usuarios registrados en la cual el administrador podrá seleccionar cualquier usuario de la lista para ver en detalle la información asociada al mismo. El caso de uso asociado a este proceso se puede observar en la figura 11.

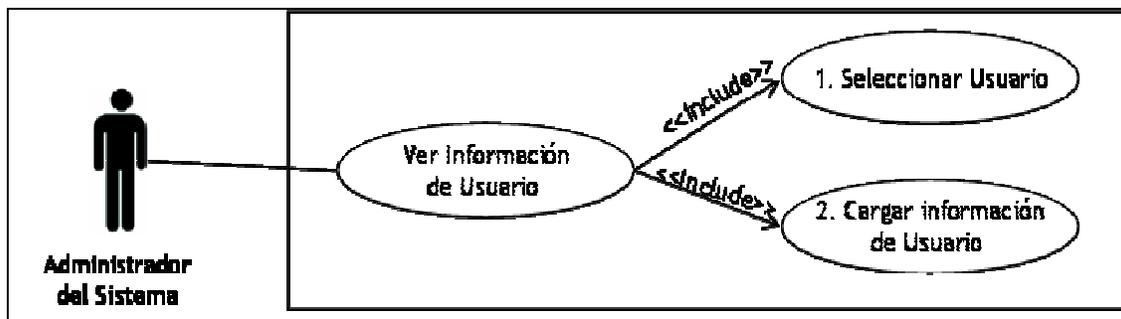


Figura 11. Caso de Uso Ver información de Usuario.

La descripción de dicho Caso de Uso se observa en la siguiente tabla (tabla 6):

Tabla 6. Descripción del Caso de Uso Ver información de Usuario.

|                                    |   |
|------------------------------------|---|
| <b>Nombre: Seleccionar Usuario</b> |   |
| Actor                              | Administrador del Sistema   |
| Descripción                        | Proceso donde el administrador del sistema selecciona un usuario de la lista para visualizar la información asociada al mismo |
| Pre-condición                      | Debe estar autenticado en la aplicación.  |
| Post-condición                     | No aplica.  |

|   |   |
|---|---|
| <b>Nombre: Validar Datos de Usuario</b> |   |
| Actor                                   | Administrador del Sistema.  |
| Descripción                             | Al seleccionar el usuario, el sistema recopila la información asociada al usuario seleccionado y es mostrada al administrador |
| Pre-condición                           | Debe estar autenticado en el sistema.   |
| Post-condición                          | No aplica.  |

### VIII. Proceso Editar Usuario

**Usuarios involucrados:** Administrador del Sistema

Este proceso, modelado en la figura 12, permite modificar la información asociada a un usuario si este desea actualizar la misma. El administrador podrá modificar tanto la información propia como la información asociada a otros usuarios. La descripción de dicho proceso se observa en la tabla 7.

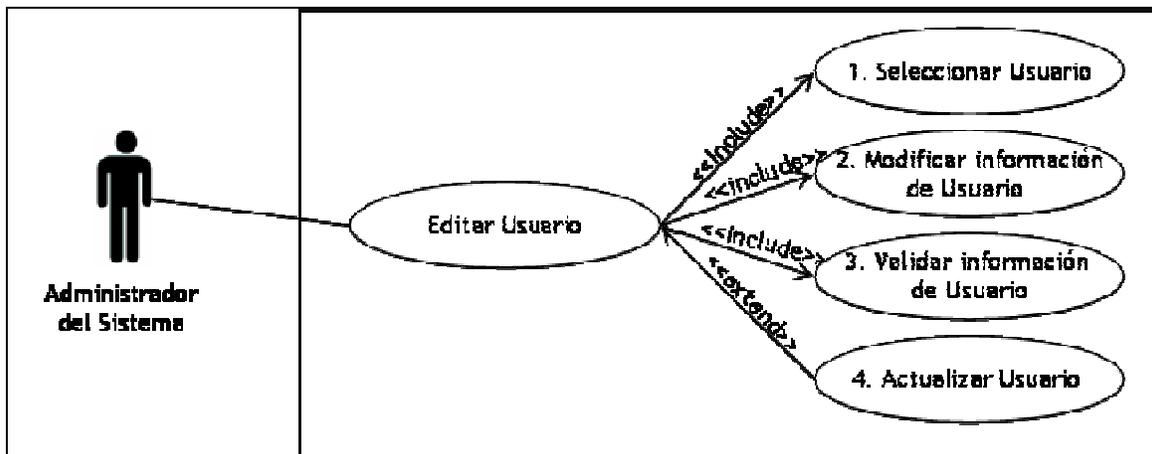


Figura 12. Caso de Uso Editar Usuario.

Tabla 7. Descripción del Caso de Uso Editar Usuario.

|   |  |
|---|--|
| <b>Nombre: Seleccionar Usuario</b>              |  |
| Actor   | Administrador del Sistema  |
| Descripción                                     | Permite seleccionar un usuario para modificar su información                     |
| Pre-condición                                   | Debe estar autenticado en la aplicación.   |
| Post-condición                                  | No aplica.   |
| <b>Nombre: Modificar información de Usuario</b> |  |
| Actor   | Administrador del Sistema.   |
| Descripción                                     | Permite al administrador poder modificar la información del usuario seleccionado |

|   |  |
|---|--|
| Pre-condición                                 | Se debe seleccionar un usuario en específico           |
| Post-condición                                | Se valida que la información suministrada es correcta. |
| <b>Nombre: Validar información de Usuario</b> |  |
| Actor   | Administrador del Sistema.                             |
| Descripción                                   | Se valida que la información suministrada sea correcta |
| Pre-condición                                 | Debe estar autenticado en el sistema.                  |
| Post-condición                                | No aplica.   |
| <b>Nombre: Actualizar Usuario</b>             |  |
| Actor   | Administrador del Sistema.                             |
| Descripción                                   | Se registra la nueva información en la BD.             |
| Pre-condición                                 | Debe estar autenticado en el sistema.                  |
| Post-condición                                | No aplica.   |

En la fase de construcción y diseño, que se abarcara en la sección de Taller de Diseño, se observa que la creación de un usuario así como la modificación de un usuario ya existente, son módulos íntimamente ligados, ya que tendrán la misma vista aunque para cada proceso se realizaran transacciones diferentes.

### IX. Proceso Eliminar Usuario.

**Usuarios involucrados:** Administrador del Sistema

Este proceso permite al administrador del sistema eliminar un usuario registrado en la aplicación. Para ello el usuario debe tener privilegios suficientes para poder eliminar usuarios dentro del sistema. La siguiente figura (figura 13) define el Modelo de Caso de Uso para este proceso:

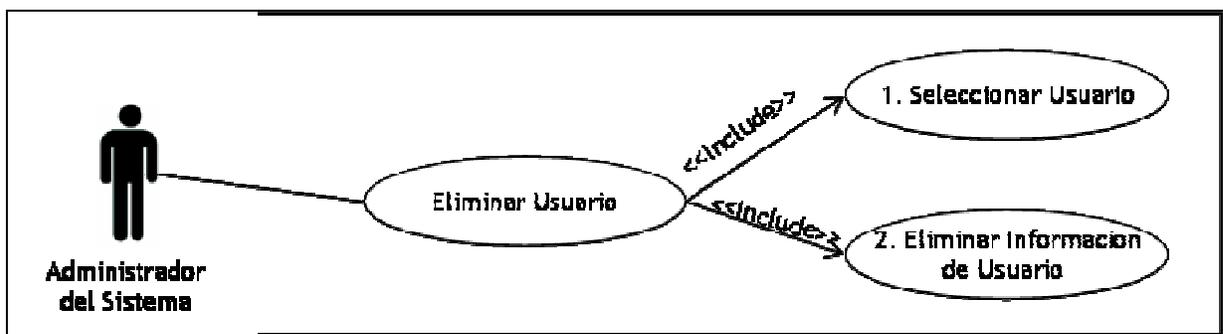


Figura 13. Caso de Uso Eliminar Usuario.

Cada proceso que interviene en este Caso de Uso se describe en la siguiente tabla (tabla 8):

Tabla 8. Descripción del Caso de Uso Eliminar Usuario.

|   |  |
|---|--|
| <b>Nombre: Seleccionar Usuario</b>              |  |
| Actor   | Administrador del Sistema  |
| Descripción                                     | Permite seleccionar un usuario para su eliminación.                                    |
| Pre-condición                                   | Debe estar autenticado en la aplicación.   |
| Post-condición                                  | No aplica.   |
| <b>Nombre: Eliminar información del Usuario</b> |  |
| Actor   | Administrador del Sistema.   |
| Descripción                                     | Permite al administrador eliminar toda la información asociada al usuario seleccionado |
| Pre-condición                                   | Debe tener privilegios para eliminar usuarios  |
| Post-condición                                  | No aplica.   |

**X. Proceso Administrar Punto de acceso.**

**Usuarios involucrados:** Administrador del Sistema.

Por medio de esta serie de funcionalidades el administrador del sistema puede controlar todo lo referente a los punto de acceso que se gestionan mediante la aplicación. Por medio de los procesos asociados el administrador puede agregar dispositivos (de manera manual o por medio de una búsqueda en la red), Visualizar información acerca del dispositivo (estado, información de gestión, interfaz de configuración), editar la información específica de cada dispositivo y eliminar dispositivos administrados. La figura 14 muestra el modelo de Casos de Uso de este proceso.

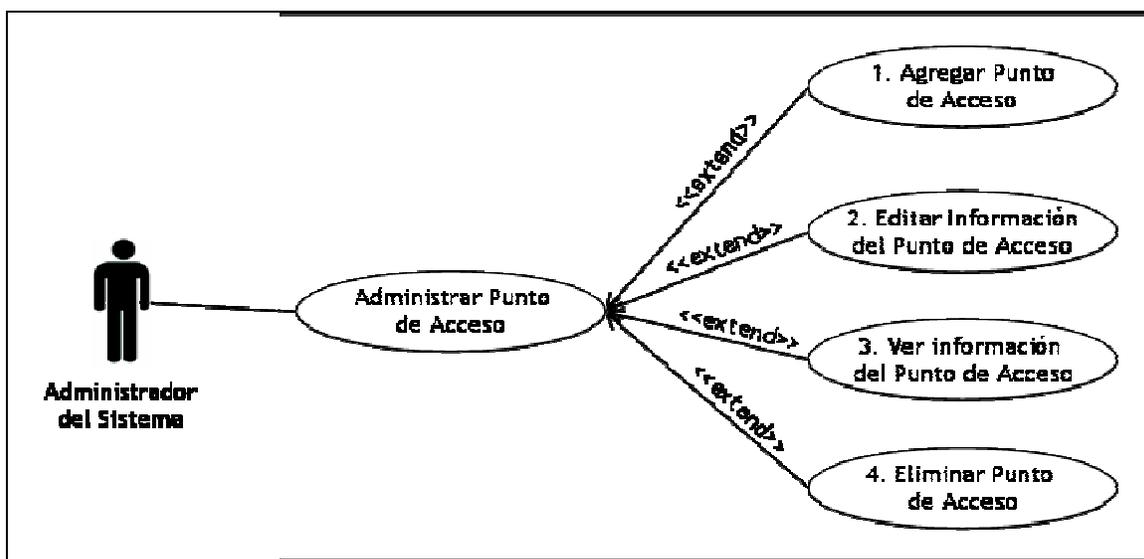


Figura 14. Caso de Uso Administrar Punto de acceso.

Dicho modelo se describe a continuación:

Tabla 9. Descripción del Caso de Uso Administrar Punto de acceso

|   |   |
|---|---|
| <b>Nombre: Agregar Punto de acceso</b>                |   |
| Actor   | Administrador del Sistema   |
| Descripción   | Permite agregar un Punto de acceso para su gestión  |
| Pre-condición   | Debe estar autenticado en la aplicación.  |
| Post-condición  | No aplica.  |
| <b>Nombre: Editar información del Punto de acceso</b> |   |
| Actor   | Administrador del Sistema.  |
| Descripción   | Permite al administrador modificar la información de un Punto de acceso administrado  |
| Pre-condición   | Debe estar autenticado en la aplicación.  |
| Post-condición  | No aplica.  |
| <b>Nombre: Ver información del Punto de acceso</b>    |   |
| Actor   | Administrador del Sistema.  |
| Descripción   | Permite al administrador visualizar la información de gestión y los datos asociados a las estadísticas de un Punto de acceso. |
| Pre-condición   | Debe estar autenticado en la aplicación.  |
| Post-condición  | No aplica.  |
| <b>Nombre: Eliminar Punto de acceso</b>               |   |
| Actor   | Administrador del Sistema.  |
| Descripción   | Permite al administrador eliminar un Punto de acceso administrado   |
| Pre-condición   | Debe estar autenticado en la aplicación.  |
| Post-condición  | No aplica.  |

## XI. Proceso Agregar Punto de acceso.

Este proceso (figura 15) permite al administrador del sistema poder agregar un dispositivo para su posterior administración. Este proceso tiene dos maneras de realizarse: el administrador puede agregar manualmente la información asociada al Punto de acceso en caso de que este en conocimiento de ello como también puede hacer un rastreo en la red, suministrando una IP perteneciente a la misma (esta IP puede ser una Ipv4, una Ipv6 o una Ipv4 con notación CIDR), y así poder seleccionar del rastreo el dispositivo de red a administrar.

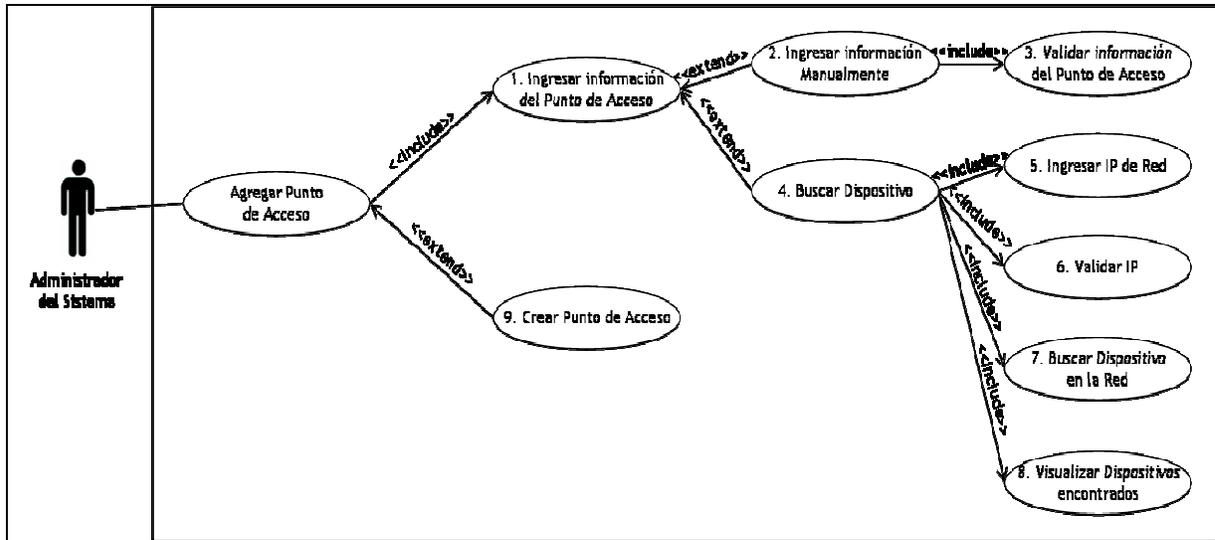


Figura 15. Caso de Uso Agregar Punto de acceso

La descripción de los procesos que intervienen en el Caso de Uso son los siguientes:

Tabla 10. Descripción del Caso de Uso Agregar Punto de acceso

|  |   |
|--|---|
| <b>Nombre: Ingresar información Manualmente</b>        |   |
| Actor  | Administrador del Sistema   |
| Descripción  | Permite al administrador incluir un nuevo Punto de acceso para su administración de manera manual, ingresando la información necesaria  |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | Se valida la información suministrada por el administrador del sistema  |
| <b>Nombre: Validar información del Punto de acceso</b> |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Proceso en el cual se valida que la información suministrada por el administrador del sistema sea correcta y que esta ya no esté registrada con anterioridad                          |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | En caso de que la información no sea correcta (un campo invalido o la información suministrada ya está registrada) se envía un mensaje al administrador indicando el error presentado |
| <b>Nombre: Buscar Dispositivo</b>                      |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Permite al administrador del sistema poder realizar un rastreo en la red para la búsqueda de dispositivos.  |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | No aplica.  |

|  |   |
|--|---|
| <b>Nombre: Ingresar IP de Red</b>                  |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | El administrador suministra a la aplicación un IP para su rastreo. Dicha IP puede ser incluida en forma de cuatro dígitos entre 0 y 255 separados por el símbolo “.” o en notación CIDR.                  |
| Pre-condición                                      | Debe estar autenticado en la aplicación.  |
| Post-condición                                     | Se valida la dirección IP   |
| <b>Nombre: Validar IP</b>                          |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Valida que la IP suministrada por el administrador sea válida   |
| Pre-condición                                      | Debe estar autenticado en la aplicación.  |
| Post-condición                                     | En caso de que la IP suministrada sea incorrecta se envía un mensaje indicándole al usuario que la IP no es válida.   |
| <b>Nombre: Buscar Dispositivo en la Red</b>        |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | El sistema realizar el rastreo de la red definida por la IP suministrada por el administrador en búsqueda de dispositivos de Red  |
| Pre-condición                                      | La IP suministrada debe ser válida.   |
| Post-condición                                     | En caso de no encontrar dispositivos en la red se envía un mensaje indicando que no se encontraron dispositivos de red. En caso contrario se muestra una lista con los dispositivos encontrados.          |
| <b>Nombre: Visualizar Dispositivos encontrados</b> |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Se muestra en el sistema la lista de dispositivos administrados en el rastreo   |
| Pre-condición                                      | Debe estar autenticado en la aplicación.  |
| Post-condición                                     | Si el usuario selecciona un dispositivo, la información asociada a este es incluida en el formulario para agregar. En caso contrario el usuario puede repetir el proceso de agregar un nuevo dispositivo. |
| <b>Nombre: Crear Punto de acceso</b>               |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | El administrador procede a agregar un nuevo Punto de acceso con la información suministrada ya sea por el proceso manual o por el rastreo de red.   |
| Pre-condición                                      | Debe estar autenticado en la aplicación.  |
| Post-condición                                     | No aplica.  |

Es necesario acotar que cuando el administrador del sistema introduce una IP para el rastreo, dicho rastreo se realiza solo en esa dirección IP. Para realizar un rastreo de toda la red es necesario que el administrador suministre la IP en

notación CIDR, la cual permite al sistema conocer la subred en la cual está dicha dirección y así realizar un rastreo completo de toda la subred asociada.

## XII. Proceso Editar información del Punto de acceso

Este proceso permite al administrador tener la facilidad de modificar la información registrada del Punto de acceso en el caso de que ocurra algún cambio (ya sea que dicho punto cambie de subred por ejemplo). El modelo de Casos de Uso de este proceso se puede observar en la siguiente figura (figura 16):

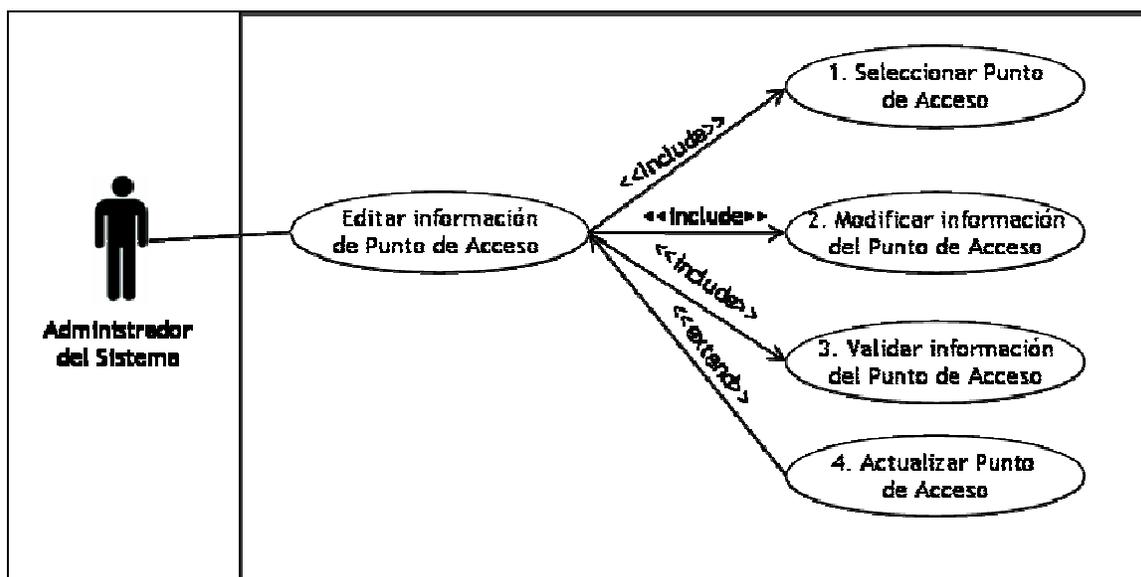


Figura 16. Caso de Uso Editar Información de Punto de acceso.

La definición del anterior Caso de Uso es la siguiente:

Tabla 11. Descripción del Caso de Uso Editar información del Punto de acceso.

|  |  |
|--|--|
| <b>Nombre: Seleccionar Punto de acceso</b>               |  |
| Actor  | Administrador del Sistema  |
| Descripción  | El administrador del sistema selecciona el Punto de acceso que desea modificar               |
| Pre-condición  | Debe estar autenticado en la aplicación.   |
| Post-condición   | No aplica.   |
| <b>Nombre: Modificar información del Punto de acceso</b> |  |
| Actor  | Administrador del Sistema.   |
| Descripción  | El administrador realiza los cambios necesario en la información asociada al Punto de acceso |
| Pre-condición  | Debe estar autenticado en la aplicación.   |
| Post-condición   | No aplica.   |
| <b>Nombre: Validar información del Punto de acceso</b>   |  |

|   |  |
|---|--|
| Actor                                     | Administrador del Sistema.   |
| Descripción                               | Se valida que la información asociada al Punto de acceso sea valida                                      |
| Pre-condición                             | Debe estar autenticado en la aplicación.   |
| Post-condición                            | En caso de no ser válida se envía un mensaje al usuario indicando que campo es invalido en el formulario |
| <b>Nombre: Actualizar Punto de acceso</b> |  |
| Actor                                     | Administrador del Sistema.   |
| Descripción                               | El sistema registra y actualiza en la base de datos la información del Punto de acceso                   |
| Pre-condición                             | La información suministrada debe ser válida.   |
| Post-condición                            | No aplica.   |

**XIII. Proceso Ver información del Punto de acceso.**

Este proceso, definido en la figura 17 permite al administrador poder visualizar toda la información asociada al Punto de acceso. Dicha información es, además de la registrada por el administrador, información para el monitoreo y la gestión del punto registrado. Por medio de diversos procesos asociados a la visualización de la información del Punto de acceso como lo son: ver la interfaz de configuración del Punto de acceso, ver las estadísticas del Punto de acceso y ver la gráfica generada en base a las estadísticas, el administrador puede monitorear y administrar cada punto individualmente, de una manera rápida.

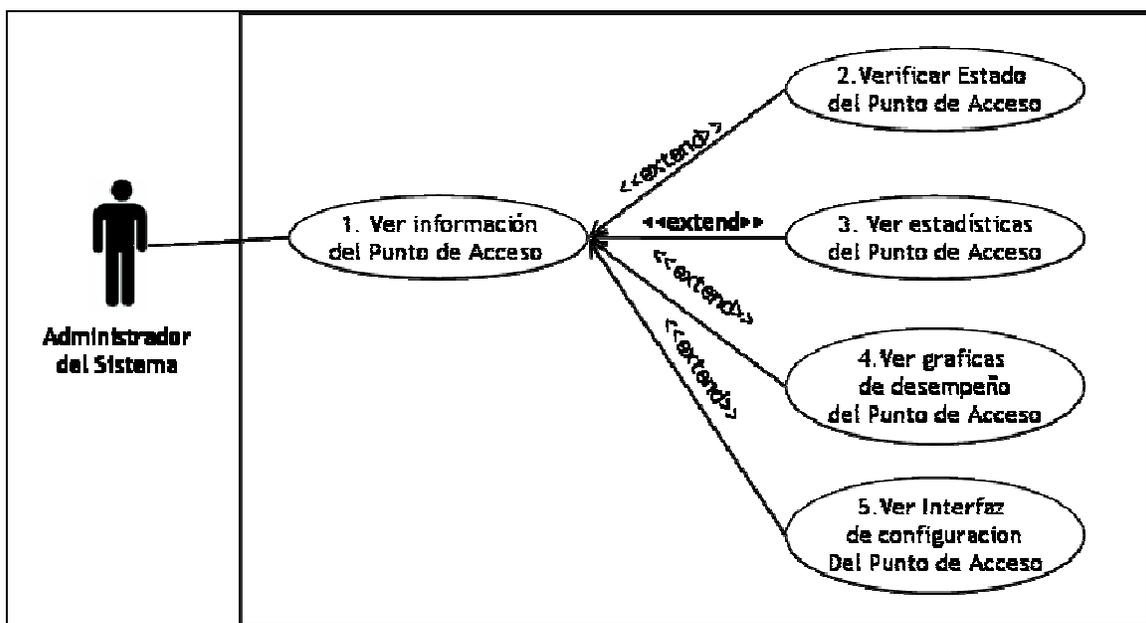


Figura 17. Caso de Uso Ver información del Punto de acceso.

La descripción del Caso de Uso anterior es el siguiente:

Tabla 12. Descripción Caso de Uso Ver información del Punto de acceso.

|  |   |
|--|---|
| <b>Nombre: Ver información del Punto de acceso</b>               |   |
| Actor  | Administrador del Sistema   |
| Descripción  | El administrador puede mediante este proceso visualizar toda la información asociada al Punto de acceso.  |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | No aplica.  |
| <b>Nombre: Verificar Estado del Punto de acceso</b>              |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Este proceso realiza un envío de paquetes ICMP (ping) al Punto de acceso seleccionado para verificar si esta activo. Los resultados son mostrados al administrador. |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | No aplica.  |
| <b>Nombre: Ver estadísticas del Punto de acceso</b>              |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Por medio de este proceso el administrador puede visualizar las veces que se hizo el monitoreo del Punto de acceso para verificar su estado.                        |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | No aplica.  |
| <b>Nombre: Ver graficas de desempeño del Punto de acceso</b>     |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Por medio de este proceso el administrador puede visualizar una grafica de la media de tiempos de respuesta del Punto de acceso.                                    |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | No aplica.  |
| <b>Nombre: Ver interfaz de configuración del Punto de acceso</b> |   |
| Actor  | Administrador del Sistema.  |
| Descripción  | Este proceso permite al administrador poder acceder a la página de configuración del Punto de acceso, la cual permite la gestión del mismo.                         |
| Pre-condición  | Debe estar autenticado en la aplicación.  |
| Post-condición   | No aplica.  |

En el momento en el que el administrador entra a la interfaz de configuración del Punto de acceso, el mismo está en un sistema distinto a la aplicación, ya que cada Punto de acceso posee una interfaz distinta. Por esa razón se incluyen las interfaces de configuración como un subsistema de la aplicación a desarrollar.

#### XIV. Proceso Eliminar Punto de acceso.

Este proceso (figura 18) permite al administrador eliminar la información asociada al punto de acceso, incluyendo estadísticas recopiladas.

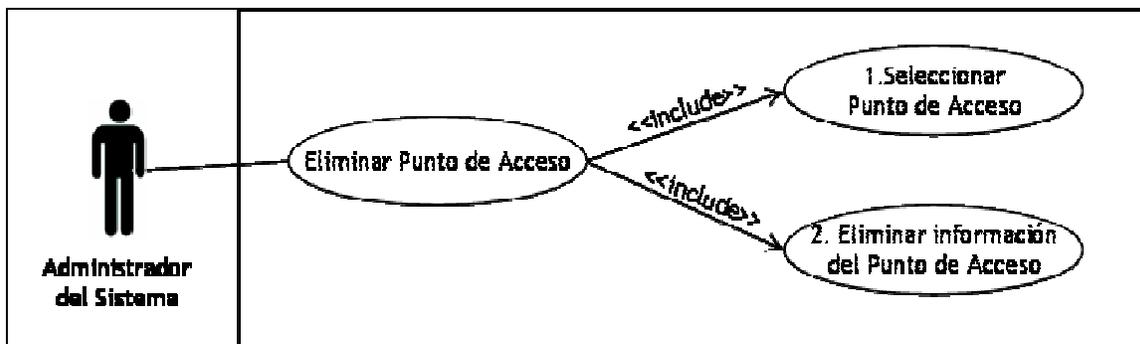


Figura 18. Caso de Uso Eliminar Punto de acceso.

La descripción asociada a este caso de uso es el siguiente:

Tabla 13. Descripción del Caso de Uso Eliminar Punto de acceso.

|   |  |
|---|--|
| <b>Nombre: Seleccionar Punto de acceso</b>              |  |
| Actor   | Administrador del Sistema  |
| Descripción   | El administrador selecciona el Punto de acceso que desea eliminar                |
| Pre-condición   | Debe estar autenticado en la aplicación.   |
| Post-condición  | No aplica.   |
| <b>Nombre: Eliminar información del Punto de acceso</b> |  |
| Actor   | Administrador del Sistema.   |
| Descripción   | El administrador elimina la información asociada al Punto de acceso seleccionado |
| Pre-condición   | Se debe seleccionar un Punto de acceso.  |
| Post-condición  | No aplica.   |

#### 4.2.2 Taller de Diseño.

A continuación se muestra por cada problema planteado, el diseño y la construcción de cada requerimiento por cada iteración realizada en el proceso de desarrollo. Para dar solución al conjunto de requerimientos se desarrollo e implemento una interfaz propia para la aplicación y se desarrollaron una serie de métodos para adaptar el desarrollo a las necesidades del usuario.

## Requerimiento 1: Autenticación

### Iteración 1:

#### Diseño

Para este requerimiento se desarrollo una interfaz que permitiera al usuario poder ingresar el nombre de usuario y contraseña en un formulario, y que este diera acceso en caso de que el usuario este registrado.

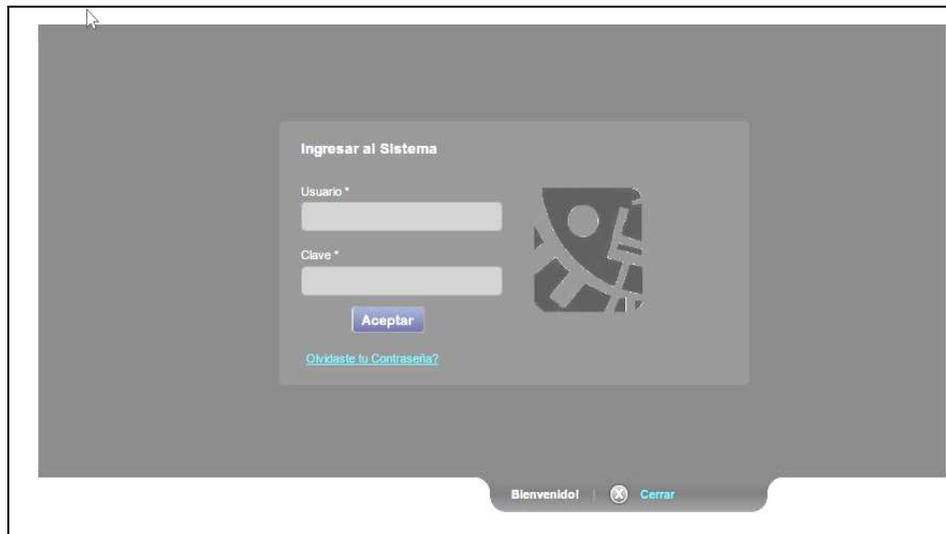


Figura 19. Pantalla de Autenticación del Sistema.

#### Construcción

El proceso de autenticación debe validar si la información suministrada por el usuario es de un usuario registrado en la aplicación.

Para ello se creó una tabla llamada adm\_tusuarios a nivel de BD donde se almacena la información referente a cada usuario registrado. En la figura 20 se puede observar la información almacenada en dicha tabla.

|                          | idUsuario | sLogin      | sPassword                        | sNombre        | sApellido | idPerfil | sCorreo               | bloqueada |
|--------------------------|-----------|-------------|----------------------------------|----------------|-----------|----------|-----------------------|-----------|
| <input type="checkbox"/> | 1         | kendinho    | 67fb78198e920c789200556418f09110 | Kendall        | Navarro   | 1        | kendinho22@gmail.com  | 0         |
| <input type="checkbox"/> | 8         | josmaguirre | cec1349bb23dfb00b7400db9a0b47652 | José<br>Manuel | Aguirre   | 1        | josmaguirre@gmail.com | NULL      |

[Marcar todos/as](#) / [Desmarcar todos](#) Para los elementos que están marcados:

Figura 20. Datos almacenados en adm\_tusuarios para validar el proceso de autenticación.

Para validar que el usuario estuviera registrado se desarrollo el siguiente método:

```
public function validaLogin($conn){
    $this->logger->info("Entro al método: validaLogin");
    $this->tiempo_inicio = microtime(true);
    //$password = $_POST['password'];
    $password = md5($_POST['password']);
    $q = 'SELECT idUsuario,sNombre,sApellido,bloqueada FROM adm_tusuarios
WHERE sLogin = \''.$_POST['login'].'\' AND sPassword = \''.$pa$
    //die($q);
    $r = $conn->Execute($q);
    if(!$r->EOF){
        $this->datosLogin($r);
        $this->return = 'exito';
        $this->logger->info("la consulta al query: ".$q." Se ejecuto
correctamente, retornando: ".$this->return);
    }else {
        $this->return = 'fallo';
        $this->logger->error("Ocurrio un Error al ejecutar el query: ".$q);
    }
    $this->tiempo_fin = microtime(true);
    $this->logger->info("Metodo validaLogin Ejecutado en: ".$this->
>tiempo_fin - $this->tiempo_inicio)." ms");
    return $this->return;
}
```

**Figura 21. Metodo validaLogin()**

Este método realiza la encriptación del password suministrado por el usuario y luego de ello realiza una consulta SQL para determinar si la información suministrada por el usuario se encuentra en la tabla adm\_tusuarios. Si dicha información se encuentra entonces el usuario esta registrado y tendrá acceso a la aplicación.

## Requerimiento 2: Creación de Usuarios.

### Iteración 2:

#### Diseño.

Para la creación del usuario, se desarrollo un formulario en el cual el administrador pudiera colocar la información asociada al usuario a crear, así como también la permisología que pudiera tener este sobre las funcionalidades del sistema. En la siguiente figura se muestra la interfaz mediante la cual el administrador puede realizar la creación de usuarios:

The screenshot shows a web application interface for user management. The main window is titled 'Administrar Usuarios' and contains a modal form titled 'Agregar Usuario'. The form has the following elements:

- Form Fields:** 'Nombre:' (text input), 'Apellido:' (text input), 'Login:' (text input), 'Password:' (password input), 'Perfil:' (dropdown menu with 'Seleccione' selected), and 'Email:' (text input). Red asterisks indicate required fields.
- Buttons:** 'Cambiar Password' (next to the password field) and 'Guardar' (at the bottom center).
- Modulos:** A list box containing 'Administracion - Administracion de Usuarios' and 'Administracion - Administracion de AP'.
- Permisos:** A section with checkboxes for 'Modificar', 'Borrar', 'Evaluar', 'Consultar', and 'Agregar'. A mouse cursor is pointing at the 'Evaluar' checkbox.
- Footer:** A note '(\*) Campos Obligatorios' and a 'CLOSE X' button in the bottom right corner.

Figura 22. Pantalla que permite realizar la creación de un usuario

### Construcción

Para realizar la inserción se tuvo en cuenta diversos aspectos. El primero a considerar es el hecho de que el usuario puede tener ciertos privilegios, los cuales le permiten tener acceso a ciertas funcionalidades en la aplicación.

Se definieron de esta manera 2 perfiles:

**Administrador de Usuarios:** Este usuario tiene la capacidad de visualizar y gestionar toda la información referente a los usuarios.

**Administrador de AP:** Este usuario tiene la capacidad de Visualizar y gestionar toda la información referente a los Punto de acceso registrados en la aplicación.

Es preciso indicar que estos perfiles no son excluyentes, es decir, un usuario puede poseer un perfil de Administrador de Usuarios y Administrador de AP. En dicho caso, este usuario tendrá la capacidad de manejar y gestionar toda la información manejada por el sistema, así como todas las funcionalidades.

Para la creación del Usuario se construyó el método que se observa en la figura 23.

```
public function addUser($conn,$nombre,$apellido,$login,$perfil,$email) {
    $nueva_clave.=rand(0,9);
    $password = md5($nueva_clave);
    //$password = md5('Ciencias123');
    $correo = $email;
    $q = 'INSERT INTO adm_tusuarios (sNombre,sApellido,sLogin,
idPerfil, sPassword, sCorreo) ';
    $q.= 'VALUES
(\\'. $nombre. '\\,\\'. $apellido. '\\,\\'. $login. '\\,\\'. $perfil. '\\,\\'. $pa
ssword. '\\,\\'. $correo. '\\) ';
    //die($q);
    $r = $conn->Execute($q);
    $array_permisos = json_decode($_POST['json_array_permisos']);
    $q = 'INSERT INTO adm_tusuarios_modulos
(idUsuario, idModulo, sPermisos) ';
    $q.= 'VALUES ('. $idUsuario. ', '. $id_modulo. ', \\'. $permiso. '\\) ';
    //die($q);
    $r = $conn->Execute($q);
    return $this->return;
}
```

Figura 23. Método addUser()

Este método toma la información del formulario y registra un nuevo usuario en el sistema. Luego del registro, se envía un correo a la dirección suministrada por el usuario con la información de la clave aleatoria generada, luego de ello el usuario puede acceder al sistema.

### Requerimiento 3: Visualización y Eliminación de Usuarios

#### Iteración 3:

#### Diseño.

Para el requerimiento de Visualización del Usuario, se diseñó una interfaz en la cual el administrador pudiera observar los usuarios en una lista. Dicha lista mostrara la información relevante referente al usuario. Al momento en el que el usuario seleccione un elemento de dicha lista, este podrá ver toda la información asociada al mismo, como se puede observar en la siguiente figura 24.

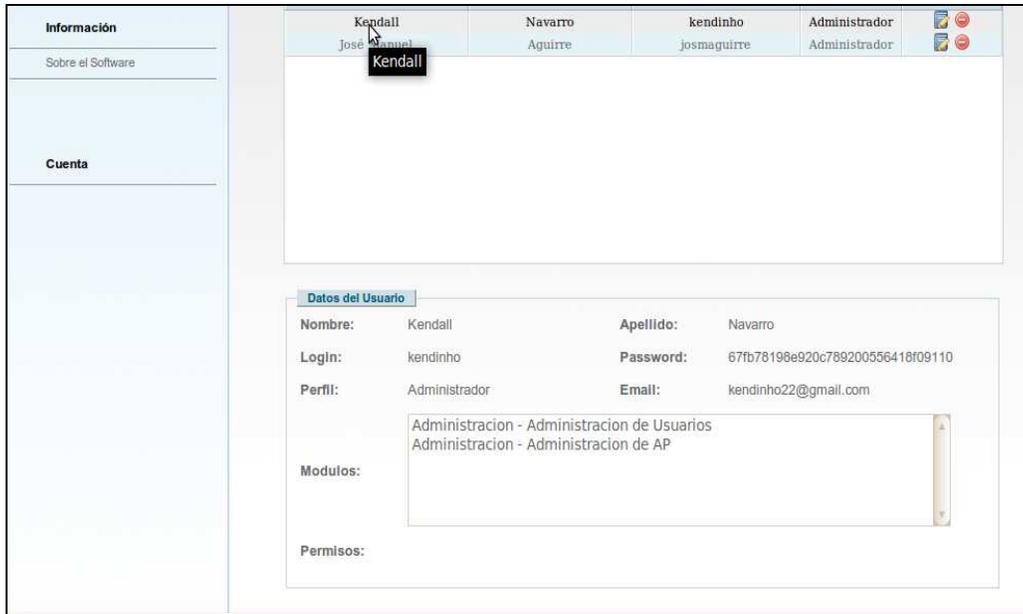


Figura 24. Interfaz de Visualización de Usuario

Para la eliminación del Usuario se tiene un icono de eliminación en cada elemento de la lista que permite realizar la eliminación del usuario. En la figura a continuación se muestra el icono que permite realizar la eliminación de un usuario en específico.

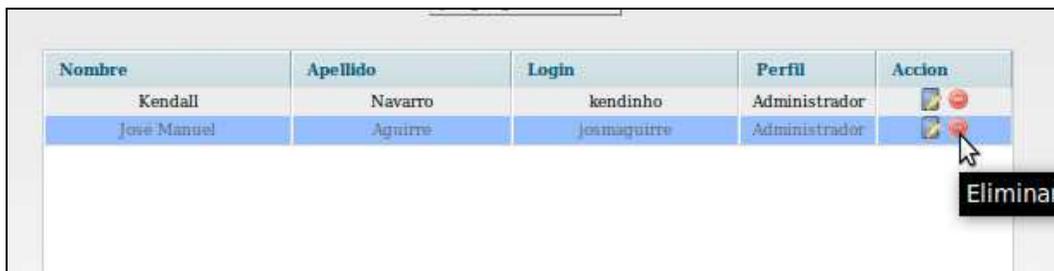


Figura 25. Botón de eliminación de Usuario en el Sistema.

### Construcción.

Para la visualización de Usuarios se desarrollo un método llamado getAllUser() (figura 26) que permitiera obtener toda la información de todos los usuarios del sistema.

```

public function getAllUser($conn,$orden='idUserario',$json=false){
    $this->logger->info("Entro al método: getAllUser");
    $this->tiempo_inicio = microtime(true);
    $q = 'SELECT idUsuario FROM adm_tusuarios ';
    $q.= 'ORDER BY '.$orden.' ';
    //die($q);
    $r = ($max!=0) ? $conn->SelectLimit($q, $max, $from) : $conn->Execute($q);
    $collection=array();
    while(!$r->EOF){
        $ue = new admTUsuario;
        $ue->getUser($conn, $r->fields['idUserario']);
        $coleccion[] = $ue;
        $r->movenext();
    }
    $this->total = $r->RecordCount();
    ($json==true) ? $coleccion = json_encode($coleccion) : '';
    $this->tiempo_fin = microtime(true);
    $this->logger->info("Metodo getAllUser Ejecutado en: ".$this->tiempo_fin - $this->tiempo_inicio)." ms");
    return $coleccion;
}

```

**Figura 26. Metodo getAllUser().**

Dicho método retorna a la vista una colección con los usuarios presentes en el sistema, para que sean mostrados en forma de tabla en la aplicación.

Para la eliminación de Usuarios se desarrollo un método llamado deleteUser() (figura 27) el cual permite eliminar toda la información asociada del usuario seleccionado en el sistema.

```

public function deleteUser($conn,$idUserario){
    $this->logger->info("Entro al método: delUser");
    $this->tiempo_inicio = microtime(true);
    try{
        $conn->BeginTrans();
        $q = 'DELETE FROM adm_tusuarios WHERE idUsuario = '.$idUserario.'';
        //die($q);
        $r = $conn->Execute($q);
        $conn->CommitTrans();
        $this->return = 'exito';
        $this->logger->info("la consulta al query: ".$q." Se ejecuto correctamente, retornando: ".$this->return);
    }catch( ADODB_Exception $e ){
        $conn->RollBackTrans();
        $this->logger->error("Ocurrio un Error al ejecutar el query: ".$q." Con el mensaje: ".$e->getMessage());
        $this->return = 'fallo';
    }
    return $this->return;
}

```

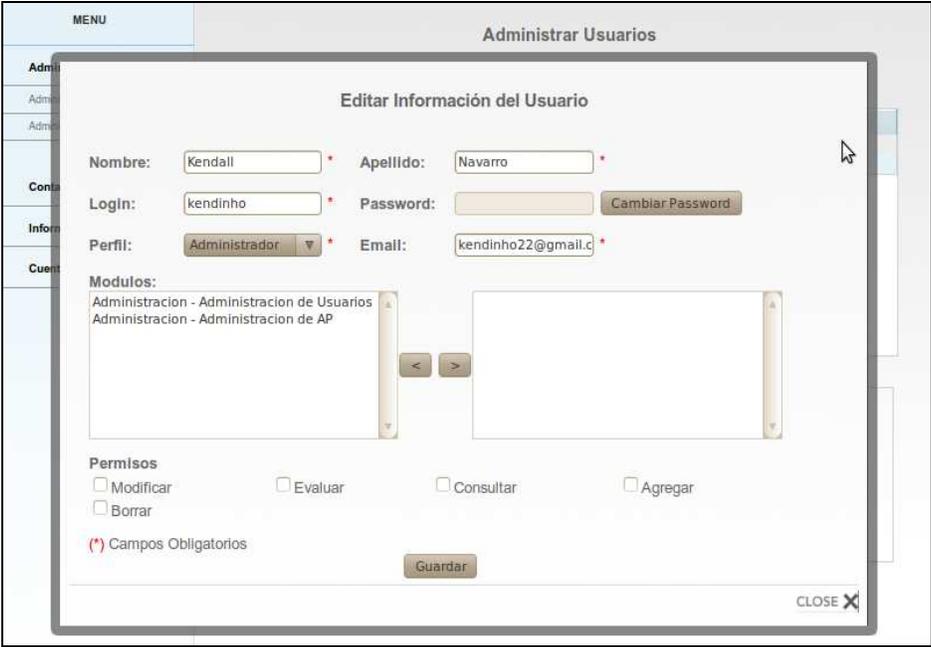
**Figura 27. Metodo deleteUser().**

## Requerimiento 4: Editar información del Usuario.

### Iteración 4

#### Diseño:

Para la interfaz de edición, se rehúsa el diseño generado en la creación, ya que se toma como premisa, que para modificar la información de un usuario en específico, se debe mostrar toda la información asociada para que el usuario modifique solo lo que él desea. En la figura 28 se observa la similitud de la interfaz en Editar Usuario con la interfaz Agregar Usuario.



The screenshot displays a web application interface for user management. The main window is titled 'Administrar Usuarios' and contains a sub-window titled 'Editar Información del Usuario'. The form includes the following elements:

- Form Fields:** 'Nombre' (Kendall), 'Apellido' (Navarro), 'Login' (kendinho), 'Password' (with a 'Cambiar Password' button), 'Perfil' (Administrador), and 'Email' (kendinho22@gmail.c).
- Modulos:** A list of modules including 'Administracion - Administracion de Usuarios' and 'Administracion - Administracion de AP'.
- Permisos:** Checkboxes for 'Modificar', 'Borrar', 'Evaluar', 'Consultar', and 'Agregar'.
- Buttons:** 'Guardar' at the bottom center and 'CLOSE' at the bottom right.
- Footer:** A note '(\*) Campos Obligatorios' is present.

Figura 28. Interfaz para la Edición del Usuario.

#### Construcción.

Para la implementar el requerimiento de edición de Usuarios, se desarrollo un método llamado editUser() el cual realiza una sentencia SQL INSERT en la BD con la información suministrada por el formulario de edición. De esta manera actualiza la información asociada al usuario con la suministrada por el formulario de edición.

```

public function editUser($conn, $idUser, $nombre, $apellido, $login,
    $perfil, $email){
    $q = 'UPDATE adm_tusuarios SET ' ;
    $q.= ' sNombre =\''.$nombre.'\' , ' ;
    $q.= ' sApellido =\''.$apellido.'\' , ' ;
    $q.= ' sLogin =\''.$login.'\' , ' ;
    $q.= ' idPerfil =\''.$perfil.'\' , ' ;
    $q.= ' sCorreo =\''.$email.'\' ' ;
    $q.= 'WHERE idUsuario = \''.$idUser.' ' ;
    //die($q);
    $q = 'INSERT INTO adm_tusuarios_modulos (idUser, idModulo,
sPermisos ) ' ;
    $q.= 'VALUES ('.$idUser.', '.$id_modulo.',
\''.$array_permisos->{$id_modulo}.'\' ) ' ;
    $r = $conn->Execute($q);
    $conn->CommitTrans();
    return $this->return;
}

```

Figura 29. Metodo editUser()

## Requerimiento 5: Agregar AP

### Iteración 5:

#### Diseño.

Para este requerimiento se desarrollo una ventana en donde el administrador puede introducir información acerca del dispositivo que desea agregar (figura 30), dicha información es registrada directamente en la Base de Datos en una tabla llamada router.

The screenshot shows a web application interface for managing access points. The main window is titled "Administrar Puntos de Acceso" and contains a sub-section for "Agregar Nuevo AP". The form includes the following fields:

- Nombre:
- Mac Address:
- marca:
- ubicacion:
- Community Name:
- Dirección IP:  \*
- Frecuencia:
- modelo:
- snmp:  \*

At the bottom of the form is a "Guardar" button. A note below the fields states "(\*) Campos Obligatorios". A "CLOSE X" button is located in the bottom right corner of the window.

Figura 30. Ventana para agregar nuevo Punto de acceso

## Construcción.

Para agregar la información del punto de acceso en la Base de Datos se desarrolló un método llamado `addRouter()`, la cual se puede observar en la figura 31.

```

public function
addRouter($conn,$nameRouter,$dirIp,$macAddress,$frecuencia,$marca,$modelo
,$ubicacion,$snmp,$communityName){
    $this->logger->info("Entro al método: addRouter");
    $this->tiempo_inicio = microtime(true);
    try{
        //$conn->BeginTrans();
        $q = 'INSERT INTO router (nameRouter, dirIp, macAddress,
frecuencia, marca, modelo, ubicacion, snmp, communityName) '
        $q.= 'VALUES
(\''.$nameRouter.'\'','\''.$dirIp.'\'','\''.$macAddress.'\'','\''.$frecuencia.'\'
\'','\''.$marca.'\'','\''.$modelo.'\'','\''.$ubicacion.'\'','\''.$snmp.'\'','\''.$c
ommunityName.'\' ) '
        //die($q);
        $r = $conn->Execute($q);
        $this->return = 'exito';
        $this->logger->info("la consulta al query: ".$q." Se ejecuto
correctamente, retornando: ".$this->return);
    }catch( ADODB_Exception $e ){
        $conn->RollBackTrans();
        $this->logger->error("Ocurrio un Error al ejecutar el query: ".$q."
Con el mensaje: ".$e->getMessage());
        $this->return = 'fallo';
    }
    $this->tiempo_fin = microtime(true);
    $this->logger->info("Metodo addRouter Ejecutado en: " .($this-
>tiempo_fin - $this->tiempo_inicio)." ms");
    return $this->return;
}

```

**Figura 31. Metodo addRouter().**

Este método permite validar la información suministrada por el administrador. La validación se encarga de verificar que dicha información suministrada no tenga redundancia en la Base de Datos, es decir, valida que el usuario no introduzca varias veces la información asociada a un mismo AP.

## Requerimiento 6: Visualizar y Eliminar Punto de acceso

### Iteración 6:

#### Diseño.

Para Visualizar los Puntos de Acceso administrados por la aplicación se desarrollo una interfaz el cual muestra la información en una lista, donde cada elemento de la misma es un Punto de acceso administrado (figura 32). Al seleccionar un AP, la información asociada es mostrada en su totalidad, de esta manera el administrador puede visualizar toda la información referente al Punto de acceso seleccionado.

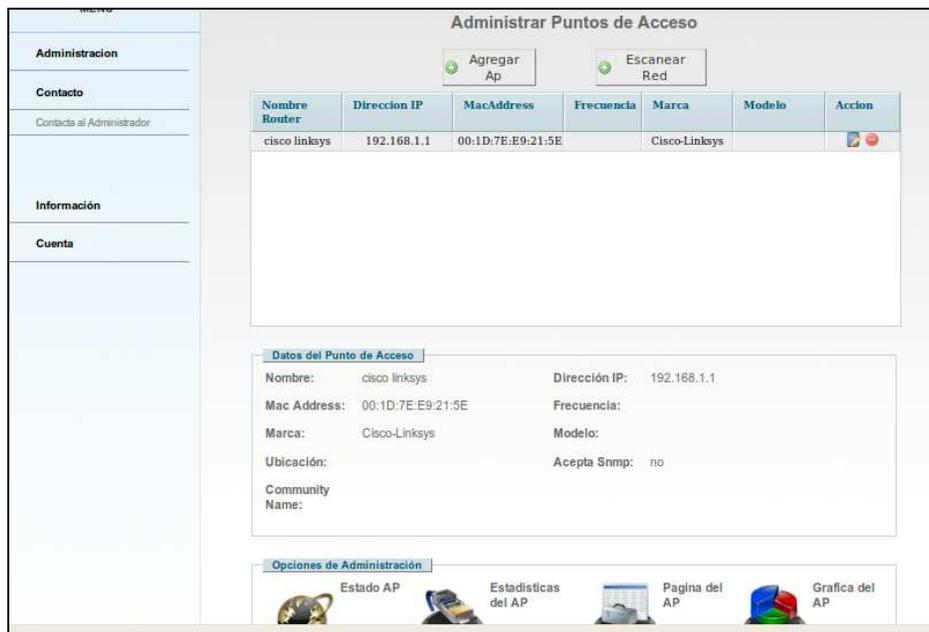


Figura 32. Interfaz de Visualización de Punto de acceso.

Para la eliminación del Punto de acceso seleccionado, se coloca en la lista de Punto de acceso un icono que representa la tarea de eliminación del mismo (figura 33).



Figura 33. Botón para eliminar un Punto de acceso.

## Construcción.

Se desarrollo el método llamado deleteRouter() (figura 34) el cual permite eliminar y el punto de acceso administrado que es seleccionado por el usuario.

```
public function deleteRouter($conn,$idRouter){
    $this->logger->info("Entro al método: deleteRouter");
    $this->tiempo_inicio = microtime(true);

    try{
        $conn->BeginTrans();

        $q = 'DELETE FROM router WHERE idRouter = '.$idRouter.'';
        //die($q);
        $r = $conn->Execute($q);
        $conn->CommitTrans();

        $this->return = 'exito';
        $this->logger->info("la consulta al query: ".$q." Se ejecuto
correctamente, retornando: ".$this->return);

    }catch( ADODB_Exception $e ){
        $conn->RollBackTrans();
        $this->logger->error("Ocurrio un Error al ejecutar el query: ".$q."
Con el mensaje: ".$e->getMessage());
        $this->return = 'fallo';
    }
    return $this->return;
}
```

Figura 34. Método deleteRouter().

La información retornada del metdo getAllRouter()es mostrada en la interfaz, lo cual permite al administrador visualizar todos los Punto de acceso administrados por la aplicación.

```
public function getAllRouter($conn,$orden='idRouter',$json=false){
    $q = 'SELECT idRouter FROM router '. 'ORDER BY '.$orden.' ';
    $r = ($max!=0) ? $conn->SelectLimit($q, $max, $from) : $conn-
>Execute($q);
    $coleccion=array();
    while(!$r->EOF){
        $ue = new admTRouter();
        $ue->getRouter($conn, $r->fields['idRouter']);
        $coleccion[] = $ue;
        $r->movenext();
    }
    $this->total = $r->RecordCount();
    ($json==true) ? $coleccion = json_encode($coleccion) : '';
    return $coleccion;
}
```

Figura 35. Metodo getAllRouter().

## Requerimiento 7: Editar Punto de acceso

## Diseño.

Para editar la información del Punto de acceso se coloca un icono de edición en la lista de Punto de acceso administrados (figura 36). El administrador solo debe seleccionar el Punto de acceso que desea editar para que se muestre la ventana de edición del mismo.

| Nombre Router   | Direccion IP   | MacAddress        | Frecuencia | Marca         | Modelo | Accion  |
|-----------------|----------------|-------------------|------------|---------------|--------|---|
| google          | 74.125.113.106 | 0                 | 50Mhz      | Cisco         | Lynx24 |   |
| Scan            | 192.168.0.30   | 00:15:58:91:8D:CA | 1000Mhz    | Forceconn     | Super  |   |
| Router de Indra | 10.168.8.2     | 00:1D:70:14:93:96 | 54mhz      | Cisco Systems |        |   |
|                 | 10.168.8.3     | 00:09:0F:50:9C:83 |            | Fortinet      |        |   |
| Cisco CC        | 192.168.0.10   | 00:00:00:00       | 2.4GHZ     | Cisco         | WRT54G |   |

Figura 36. Botón “Editar Router”.

## Construcción.

Para la edición del Punto de acceso se desarrollo un método llamado editRouter() (figura 37), el cual realiza una actualización de la información del Punto de acceso seleccionado.

```
public function
editRouter($conn,$idRouter,$nameRouter,$dirIp,$puerto,$macAddress,$frecue
ncia,$marca,$modelo,$ubicacion,$snmp,$communityName){
$this->logger->info("Entro al método: editRouter");
    $this->tiempo_inicio = microtime(true);
    $q = 'UPDATE router SET ' ;
    $q.= ' nameRouter =\''.$nameRouter.'\' , ' ;
    $q.= ' dirIp =\''.$dirIp.'\' , ' ;
    $q.= ' puerto =\''.$puerto.'\' , ' ;
    $q.= ' macAddress =\''.$macAddress.'\' , ' ;
    $q.= ' frecuencia =\''.$frecuencia.'\' , ' ;
    $q.= ' marca =\''.$marca.'\' , ' ;
    $q.= ' modelo =\''.$modelo.'\' , ' ;
    $q.= ' ubicacion =\''.$ubicacion.'\' , ' ;
    $q.= ' snmp =\''.$snmp.'\' , ' ;
    $q.= ' communityName =\''.$communityName.'\' ' ;
    $q.= 'WHERE idRouter = '.$idRouter.' ' ;
    //die($q);
    $r = $conn->Execute($q);

    $this->return = 'exito';
    $this->logger->info("la consulta al query: ".$q." Se ejecuto
correctamente, retornando: ".$this->return);
    return $this->return;
}
```

Figura 37. Método editRouter().

## Requerimiento 8: Escanear Red

### Iteración 7:

## Diseño:

Para poder realizar un escaneo de la red, con la finalidad de visualizar los Punto de accesos que se encuentran en la misma, se diseño una interfaz (figura 38) donde el administrador pueda introducir una IP de la red que desea escanear en notación CIDR con la finalidad de determinar en qué subred se va a realizar el rastreo de Punto de acceso.

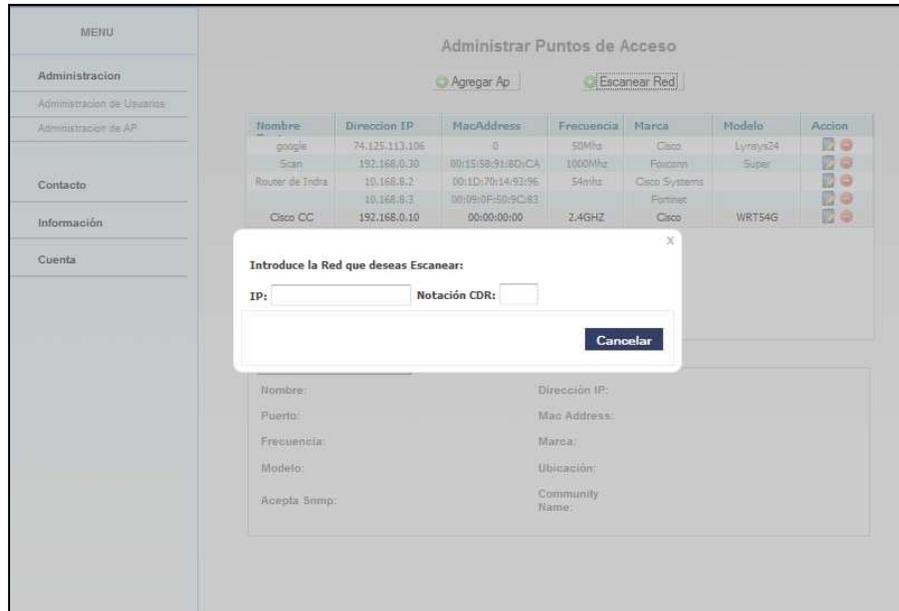


Figura 38. Interfaz de Escanear Red.

## Construcción:

Para el rastreo de Punto de acceso en la Red se desarrollo el método `networkScan()` la cual hace un rastreo de la subred definida por el administrador. Este método verifica que el rastreo realizado no repita IP's de dispositivos que ya estén actualmente registrados como Punto de acceso administrados.

```

public function networkScan ($conn, $notacionCDR, $obt1, $obt2, $obt3,
$obt4, $orden='idRouter', $json=false){
    $dirIp = "$obt1.$obt2.$obt3.$obt4";
    $dirIps = array();
    $scanRed = array(array(),array());
    $existe = 0;
    $i = 0;
    $util = new Utils();
    $dirIps = $util->allColumnTable($conn,'dirIp','router',$orden);

    if($notacionCDR == '0'){
        echo shell_exec("sudo nmap -oX ../../files/scan/networkScan.xml -n
-sT -T Normal -O -sR -F -P0 ". $dirIp . " 2>&1 > /dev/null");
    } else {
        echo shell_exec("sudo nmap -oX ../../files/scan/networkScan.xml -n
-sT -T Normal -O -sR -F -P0 ". $dirIp . "/" . $notacionCDR . " 2>&1 >
/dev/null");
    }

    $xmlObject = new DOMDocument();
    $xmlObject->load('../../files/scan/networkScan.xml');

    $hosts = $xmlObject->getElementsByTagName('host');

    foreach ($hosts as $host){
        //Se realize el studio del archive XML de resultado
    }
    return $scanRed;
}

```

Figura 39. Método networkScan().

### 4.3.3. Implementación.

Para cada proceso desarrollado se muestran a continuación las pruebas realizadas para verificar que la aplicación mantiene un comportamiento correcto:

#### a) Agregar, editar y eliminar usuario:

##### Agregar Usuario:

Para realizar la prueba de esta funcionalidad se crearon 4 usuarios: josmaguirre, amorales, kendall y dvillavicencio. Estos usuarios corresponden al equipo de desarrollo así como el administrador del sistema. En la figura a continuación se muestra la interfaz de Agregar Usuario al momento de realizar la prueba.

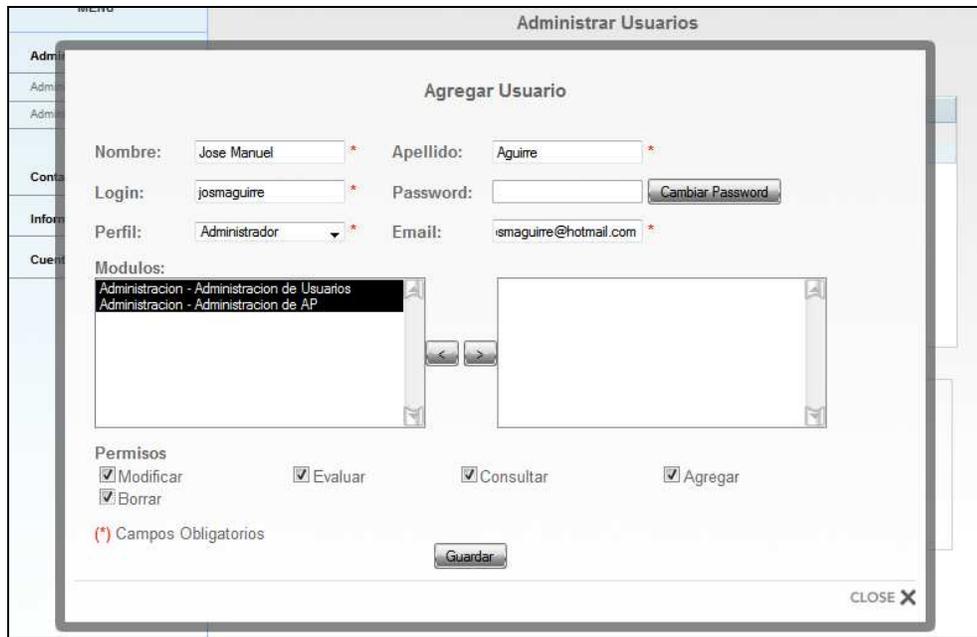


Figura 40. Prueba de Agregar Usuario.

Usuarios agregados: josmaguirre, amorales, kendall, dvillavicencio.

Resultado de las pruebas: Exitoso.

En la figura 41 se observan los usuarios registrados mediante la interfaz de visualización.

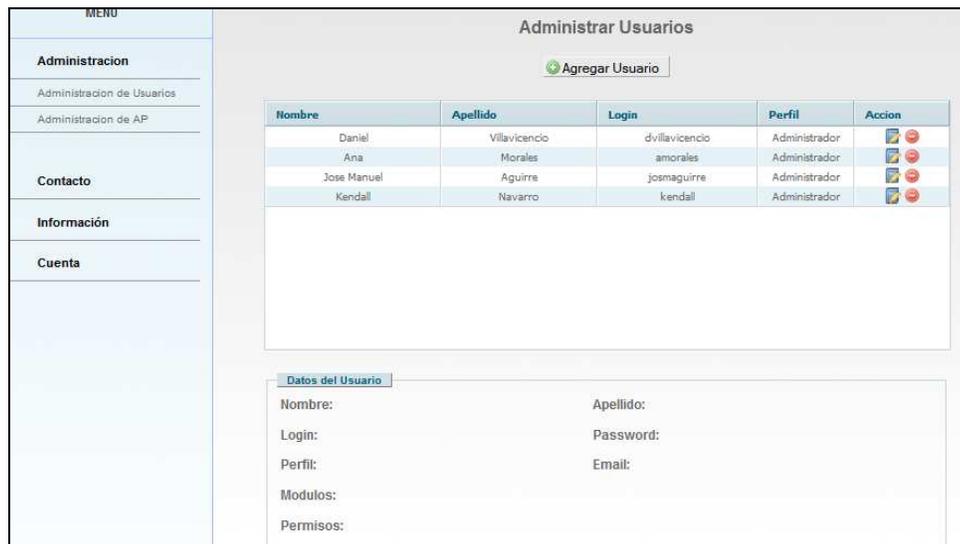


Figura 41. Interfaz de visualización de Usuarios.

### Editar Usuarios:

Para realizar esta prueba se definió realizar la edición de los usuarios josmaguirre y kendall, los cuales corresponden al equipo de desarrollo. Se modifico información como los Modulos que pueden utilizar así como cambio de clave.



Figura 42. Prueba de Editar Usuario.

Usuarios Editados: josmaguirre, kendall.

Resultado de las Pruebas: Exitoso.

En la siguiente figura se muestra la información modificada del usuario josmaguirre:

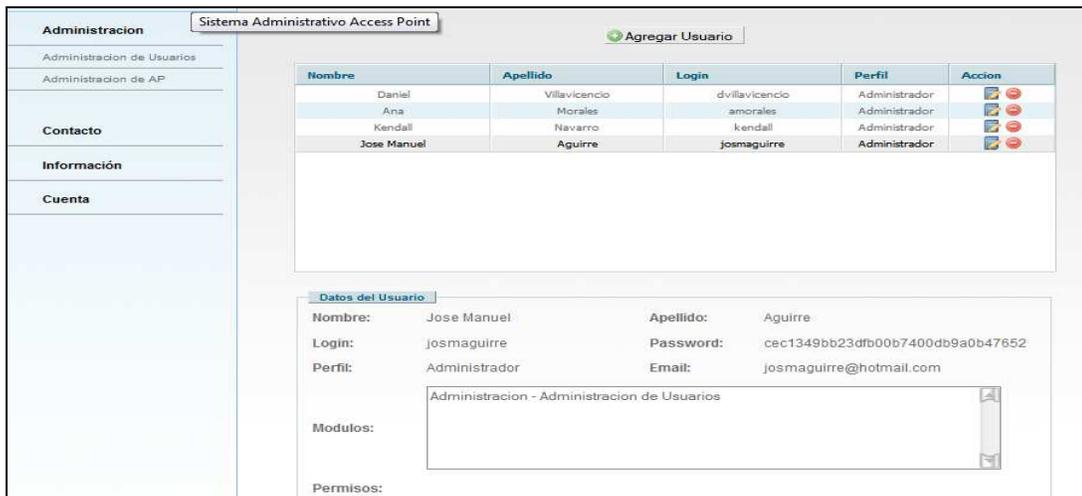


Figura 43. Datos Modificados para el Usuario josmaguirre.

### Eliminar Usuario:

Para realizar la prueba se definió la eliminación del usuario josmaguirre (figura 44). Esta prueba es para verificar que la información asociada al usuario es eliminada en su totalidad del sistema.

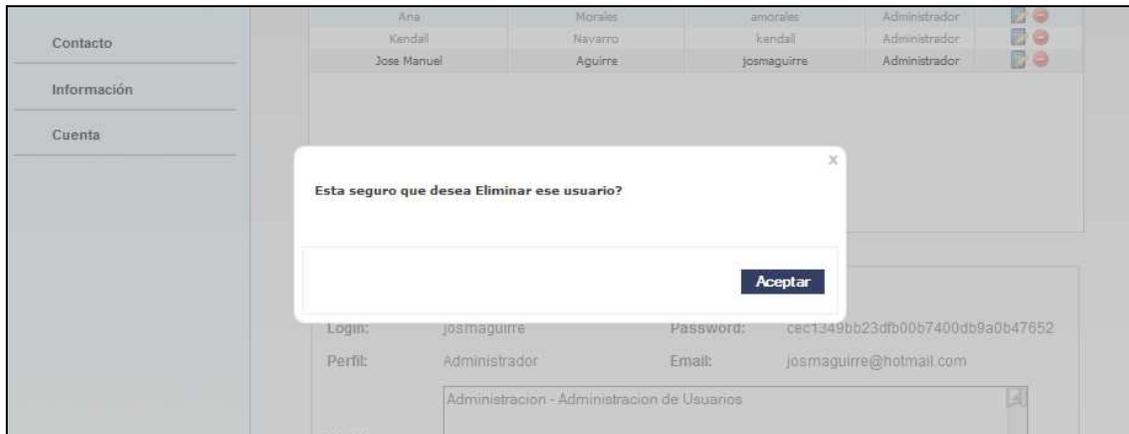


Figura 44. Prueba de Eliminar Usuario.

Usuario eliminado: josmaguirre.

Resultado de la prueba: Exitoso.

En la siguiente figura se muestra la tabla de visualización de usuarios, en la cual no se presenta el usuario eliminado:



Figura 45. Tabla de Visualización de Usuarios.

A nivel de Base de Datos no se encuentra el usuario josmaguirre, como se muestra en la figura 46:

| idUsuario | sLogin         | sPassword                        | sNombre     | sApellido     | idPerfil | sCorreo                          | bloqueada |
|-----------|----------------|----------------------------------|-------------|---------------|----------|----------------------------------|-----------|
| 9         | dvillavicencio | 589b755fe50f8234805f302b77f3d62c | Daniel      | Villavicencio | 1        | daniel.villaciencio@ciens.ucv.ve | (NULL)    |
| 11        | amorales       | 911d9159069558cb8ba42ee8248b452a | Ana         | Morales       | 1        | amorales@ciens.ucv.ve            | (NULL)    |
| 13        | josmaguirre    | 64065543cf0f9d459f25bcb152c85710 | Jose Manuel | Aguirre       | 1        | josmaguirre@hotmail.com          | (NULL)    |

Figura 46. Tabla adm\_tusuarios con los usuarios registrados.

**b) Agregar, editar y eliminar Punto de acceso:**

**Agregar Punto de acceso:**

Para realizar esta prueba se realizo la agregación de un Punto de acceso local de manera manual, según se muestra en la figura 47:

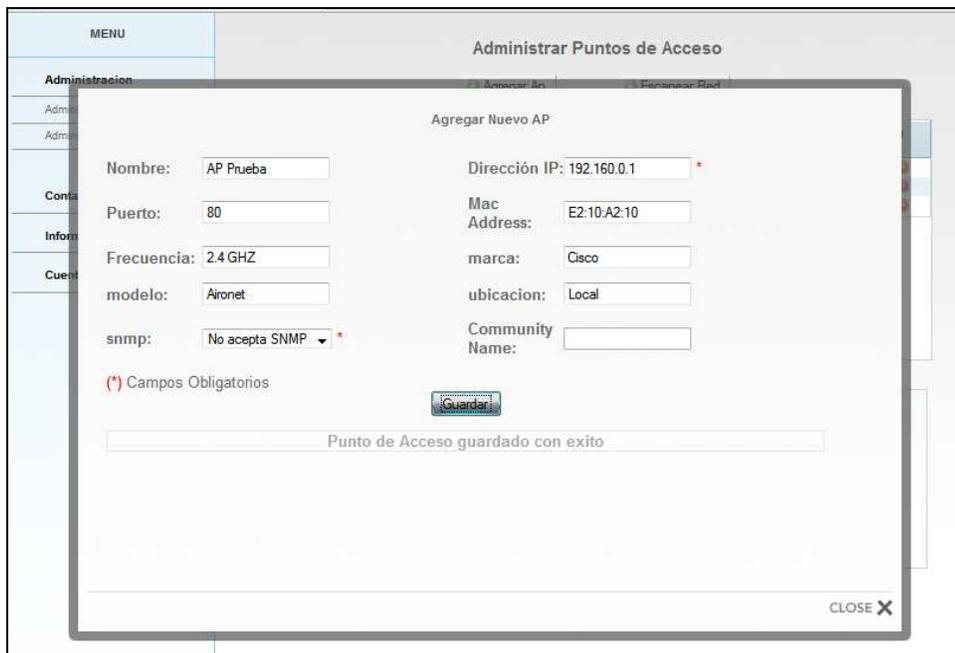


Figura 47. Prueba de Agregar Punto de acceso manual.

**Resultado de la Prueba: Exitoso.**

Luego de realizar la prueba se puede visualizar la información de dicho punto en la Interfaz de visualización de Punto de acceso (figura 48).



Figura 48. Interfaz de Visualización de Punto de acceso.

**Editar Punto de acceso:**

Para esta prueba se realizo la edición manual del Punto de acceso creado “AP Prueba” como se muestra en la figura 49. Se modifico específicamente la MAC address del Punto de acceso.

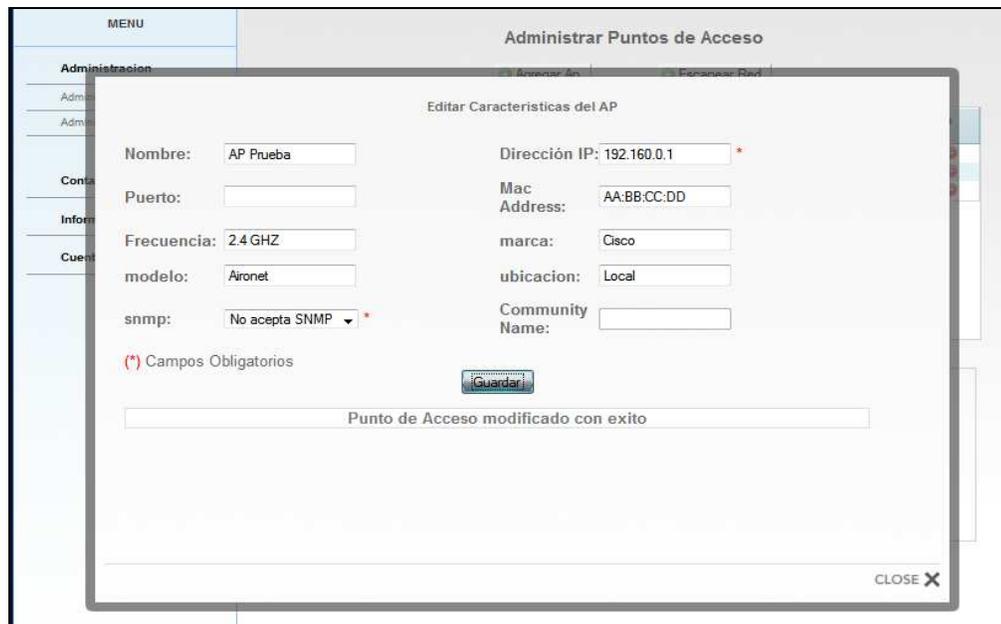


Figura 49. Interfaz de Modificación de Punto de acceso.

**Resultado de la prueba: Exitoso.**

**Eliminar Punto de acceso:**

Para esta prueba se procede a la eliminación del Punto de acceso creado “AP Prueba” como se muestra en la figura 50.

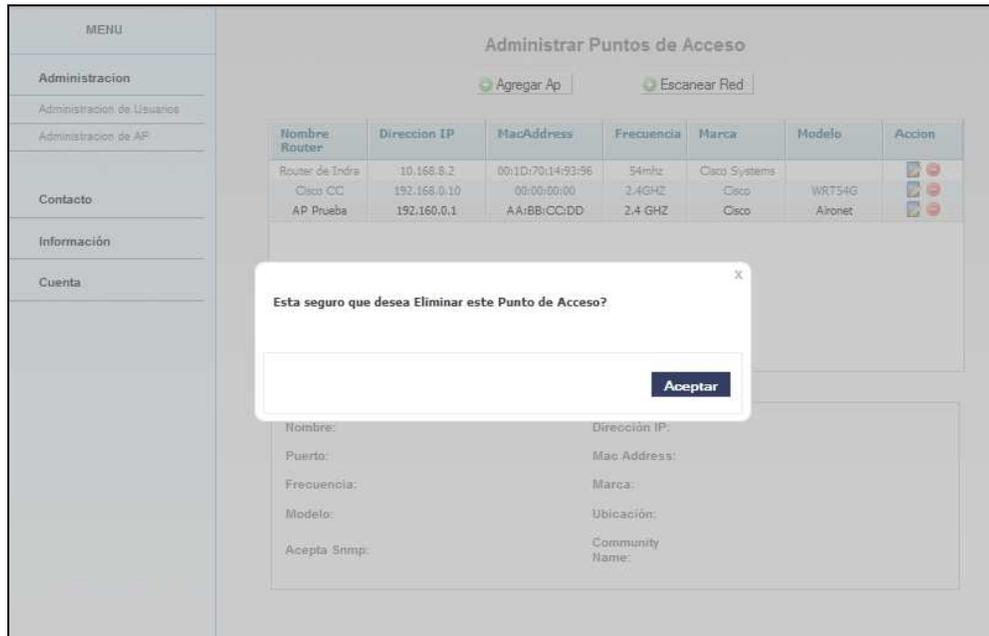


Figura 50. Ventana de Eliminación de Punto de acceso.

**Resultado de la Prueba: Exitoso**

Para verificar que se ha eliminado en su totalidad el Punto de acceso se observa en la figura 51 la tabla router con los router registrados luego de la eliminación.

| ter | nameRouter      | dirIp        | macAddress        | frecuencia | marca         | modelo | ubicacion             | snmp | communityName |
|-----|-----------------|--------------|-------------------|------------|---------------|--------|-----------------------|------|---------------|
| 9   | Router de Indra | 10.168.8.2   | 00:1D:70:14:93:96 | 54mhz      | Cisco Systems |        | Edificio Pioneer      | no   |               |
| 13  | Cisco CC        | 192.168.0.10 | 00:00:00:00       | 2.4GHZ     | Cisco         | WRT54G | Centro de Computacion | no   |               |

Figura 51. Tabla router con los Punto de acceso registrados.

**c) Escanear red:**

Para esta prueba se realizo un rastreo local de la red, especificando la IP del Punto de acceso que se desea encontrar, como se muestra en la figura 52.

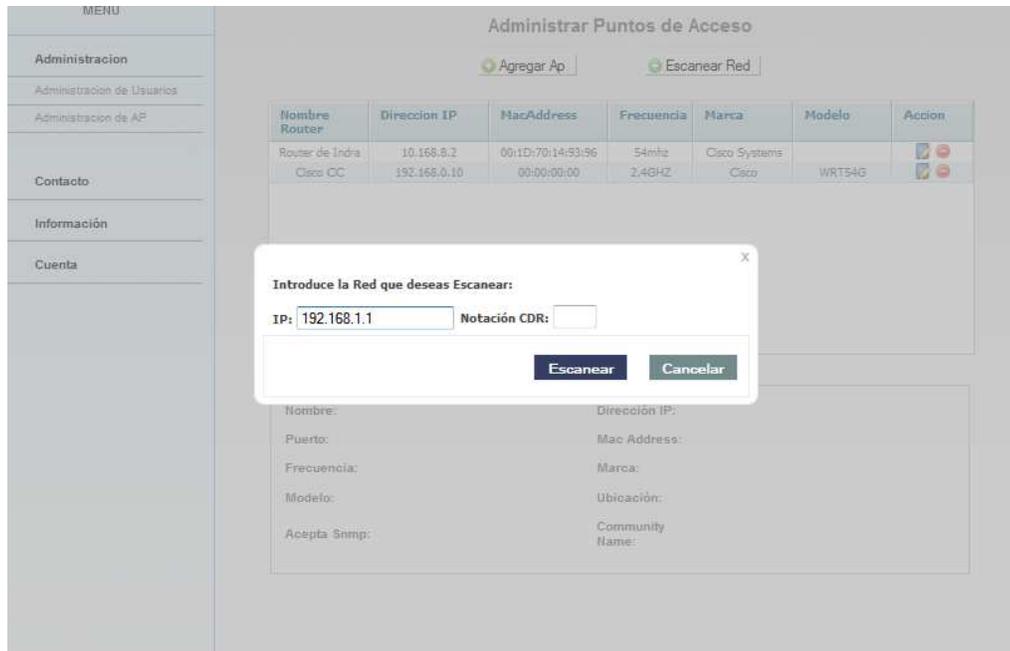


Figura 52. Rastreo de Punto de acceso.

Al realizar el rastreo se tuvo como resultado e siguiente dispositivo:



Figura 53. Resultado del Rastreo del Punto de acceso.

**Resultado de la Prueba: Exitoso.**

Luego de seleccionar el Punto de acceso encontrado se procede a agregarlo a la lista de Puntos de Acceso administrados.

En la siguiente figura se muestra la ventana con la identificación del Punto de acceso agregado satisfactoriamente.

The screenshot displays a web application interface for managing access points. On the left is a navigation menu with categories: MENU, Administracion, Contacto, Información, and Cuenta. The main content area is titled 'Administrar Puntos de Acceso' and includes two buttons: 'Agregar Ap' and 'Escanear Red'. Below these is a table listing three access points. The third entry, 'Rastreo AP', is highlighted. Below the table, a 'Datos del Punto de Acceso' section provides detailed information for the selected 'Rastreo AP'.

| Nombre Router   | Direccion IP | MacAddress        | Frecuencia | Marca         | Modelo | Accion |
|-----------------|--------------|-------------------|------------|---------------|--------|--------|
| Router de Indra | 10.168.8.2   | 00:1D:70:14:93:96 | 54mhz      | Cisco Systems |        |        |
| Cisco CC        | 192.168.0.10 | 00:00:00:00       | 2.4GHZ     | Cisco         | WRT54G |        |
| Rastreo AP      | 192.168.1.1  |                   | 2.4 GHZ    | Cisco         | Cisco  |        |

| Datos del Punto de Acceso |             |
|---------------------------|-------------|
| Nombre:                   | Rastreo AP  |
| Puerto:                   |             |
| Frecuencia:               | 2.4 GHZ     |
| Modelo:                   | Cisco       |
| Acepta Snmp:              | no          |
| Dirección IP:             | 192.168.1.1 |
| Mac Address:              |             |
| Marca:                    | Cisco       |
| Ubicación:                | Laboratorio |
| Community Name:           |             |

Figura 54. Punto de acceso registrado luego del rastreo.

## V. Escenarios de Prueba

A continuación se presentan el conjunto de escenarios de prueba utilizados para comprobar el funcionamiento del sistema. Estas pruebas se enfocaron específicamente en el proceso de rastreo de dispositivos, el cual debe ser comprobado haciendo el rastreo tanto en distintas subredes de la Facultad de Ciencias así como también en el Centro de Computación, el cual tendrá el servidor en el cual la aplicación va a entrar en actividad.

### Escenario 1.

Este escenario se realizó en las instalaciones del Laboratorio ICARO. La finalidad de la prueba era comprobar si el rastreo de dispositivos desarrollado era capaz de encontrar dispositivos inalámbricos que se encontraban en ese momento en la red.

En el ambiente de la Prueba se tuvieron los siguientes dispositivos:

- 1 dispositivo inalámbrico (Router D-LINK, IP 192.168.1.1).
- Una computadora personal, la cual iba a actuar como servidor donde se encontraba contenida la aplicación. Dicha computadora recibe servicio de internet gracias al dispositivo D-LINK mencionado anteriormente.

### Finalidad de la Prueba

La finalidad de la prueba era validar que la aplicación fuera capaz de encontrar dispositivos que se encontraran dentro de la misma red en la cual el servidor donde la aplicación estaba prestando servicio se encontraba.

### Inicio de la Prueba

Para el inicio de la prueba se colocó en la opción “Escanear Red” de la página de administración de punto de acceso, la IP del router. No se colocó ningún número en la notación CIDR, lo que le indica a la aplicación que el rastreo solo se realizara únicamente sobre la IP suministrada.

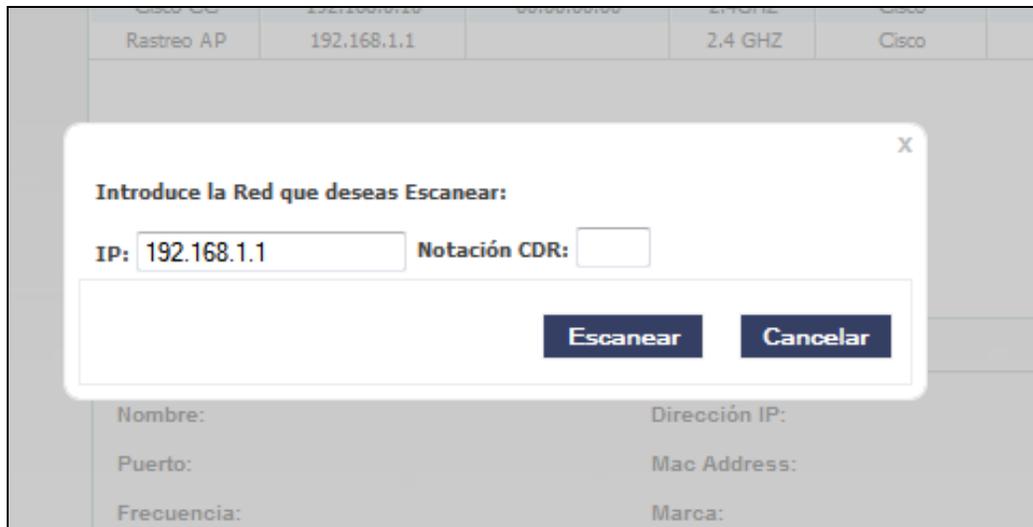


Figura 55. Inicio de la prueba, Escenario 1.

Luego de seleccionar el botón “Escanear” el resultado obtenido es verificado en el archivo *networkScan.xml* el cual es generado por la aplicación luego del rastreo. Es allí donde el resultado del rastreo se observa (figura 55).

```

1 <host starttime="1312338968" endtime="1312338978"><status state="up" reason="arp-response"/>
2 <address addr="192.168.1.1" addrtype="ipv4" />
3 <address addr="00:23:69:95:82:BF" addrtype="mac" vendor="Cisco-Linksys" />
4 <hostnames />
5 <ports><extraports state="filtered" count="98">
6 <extrareasons reason="no-responses" count="98"/>
7 </extraports>
8 <port protocol="tcp" portid="80">
9 <state state="open" reason="syn-ack" reason_ttl="0"/>
10 <service name="http" method="table" conf="3" />
11 </port>
12 <port protocol="tcp" portid="113">
13 <state state="closed" reason="conn-refused" reason_ttl="0"/>
14 <service name="auth" method="table" conf="3" />
15 </port>
16 </ports>
17 <os><portused state="open" proto="tcp" portid="80" />
18 <portused state="closed" proto="tcp" portid="113" />
19 <portused state="closed" proto="udp" portid="31222" />
20 <osclass type="router" vendor="D-Link" osfamily="embedded" accuracy="100" />
21 <osclass type="linux" vendor="D-Link" osfamily="embedded" accuracy="100" />
22 <osmatch name="D-Link DWL Series" accuracy="100" line="643" />
23 </os>
24 <uptime seconds="4092" lastboot="Tue Aug 2 20:58:06 2011" />
25 <distance value="1" />
26 <tcpsequence index="263" difficulty="Good luck!"
27 values="218E42B7,3AC8DE30,AEABED3E,1BCB98B4,F8F9CB36,78804488" />
28 <ipidsequence class="Incremental" values="35D0,35D2,35D4,35D6,35D8,35DA" />
29 <tcptssequence class="100HZ" values="63DE3,63DED,63DF7,63E01,63E0B,63E15" />
30 <times srtt="925" rttvar="306" to="100000" />
31 </host>

```

Figura 56. Resultados del rastreo en el Escenario 1.

En la figura se puede observar que el resultado genera una entrada en el archivo XML en cual define que la IP 192.168.1.1 corresponde a un dispositivo D-Link DWL Series. El resultado mostrado a usuario es el siguiente:

| <b>Resultado del Scanning de la RED</b> |              |                   |               |                   |                           |
|---|--------------|-------------------|---------------|-------------------|---------------------------|
| Nº                                      | Dirección IP | Dirección MAC     | Distribuidor  | Sistema Operativo | Opción                    |
| 1                                       | 192.168.1.1  | 00:23:69:95:82:BF | Cisco-Linksys | D-Link DWL Series | <a href="#">Registrar</a> |

5  Numero de Registros Pagina 1 de 1

Figura 57. Resultado mostrado al usuario en el Escenario 1.

De esta manera se pudo validar que el rastreo de dispositivos que se encuentran en la misma red se realizo con éxito.

## Escenario 2.

Esta prueba tuvo como finalidad validar que el rastreo de dispositivos en la red se realizara de manera efectiva. Para ello se instalo la aplicación en una computadora personal ubicada en el Laboratorio de GROUPWARE, que se encuentra en las instalaciones de la Escuela de Computación de la Facultad de Ciencias.

En el ambiente de la prueba se tuvieron los siguientes dispositivos:

- 1 router inalámbrico (TP-LINK, IP 192.168.0.1).
- Una computadora personal del laboratorio, el cual actuó como servidor donde la aplicación estaba contenida. Dicha computadora personal estaba dentro de la misma red en la cual se encontraba el router TP-LINK mencionado anteriormente.

### Finalidad de la Prueba

La finalidad de la prueba era validar que el rastreo de dispositivos que se encuentran dentro de la misma red se realizara de manera satisfactoria.

## Inicio de la Prueba.

Para el inicio de la prueba se colocó en la opción “Escanear Red” de la página de administración de punto de acceso, la IP del router. No se colocó ningún número en la notación CIDR, lo que le indica a la aplicación que el rastreo solo se realizara únicamente sobre la IP suministrada.



Figura 58. Inicio de la prueba en el Escenario 2.

Luego de seleccionar el botón “Escanear” el resultado obtenido es verificado en el archivo *networkScan.xml* el cual es generado por la aplicación luego del rastreo. Es allí donde el resultado del rastreo se observa (figura 59).

```

1 <host starttime="1312338968" endtime="1312338978"><status state="up" reason="arp-response"/>
2 <address addr="192.168.0.1" addrtype="ipv4" />
3 <address addr="00:23:69:95:82:BF" addrtype="mac" vendor="TP-Link" />
4 <hostnames />
5 <ports><extraports state="filtered" count="98">
6 <extrareasons reason="no-responses" count="98"/>
7 </extraports>
8 <port protocol="tcp" portid="80"><state state="open" reason="syn-ack" reason_ttl="0"/><service
9 <port protocol="tcp" portid="113"><state state="closed" reason="conn-refused" reason_ttl="0"/>
10 </ports>
11 <os><portused state="open" proto="tcp" portid="80" />
12 <portused state="closed" proto="tcp" portid="113" />
13 <portused state="closed" proto="udp" portid="31222" />
14 <osclass type="router" vendor="TP-Link" osfamily="embedded" accuracy="100" />
15 <osclass type="" vendor="TP-Link" osfamily="embedded" accuracy="100" />
16 <osmatch name="TL-WR Series" accuracy="100" line="643" />
17 </os>
18 <uptime seconds="4092" lastboot="Tue Aug 2 20:58:06 2011" />
19 <distance value="1" />
20 <tcpsequence index="263" difficulty="Good luck!" values="218E42B7,3AC8DE30,AEABED3E,1BCB98B4,F8
21 <ipidsequence class="Incremental" values="35D0,35D2,35D4,35D6,35D8,35DA" />
22 <tcpssequence class="100HZ" values="63DE3,63DED,63DF7,63E01,63E0B,63E15" />
23 <times srtt="925" rttvar="306" to="100000" />
24 </host>

```

Figura 59. Resultados de las pruebas en el Escenario 2.

En la figura anterior se puede observar que el resultado genera una entrada en el archivo XML en cual define que la IP 192.168.1.1 corresponde a un dispositivo D-Link DWL Series. El resultado mostrado a usuario es el siguiente:

**Resultado del Scanning de la RED**

| Nº | Dirección IP | Dirección MAC     | Distribuidor | Sistema Operativo | Opción    |
|----|--------------|-------------------|--------------|-------------------|-----------|
| 1  | 192.168.0.1  | 00:23:69:95:82:BF | TP-Link      | TL-WR Series      | Registrar |

5 Numero de Registros Pagina 1 de 1

Figura 60. Resultado mostrado al usuario en el Escenario 2.

### Escenario 3.

Esta prueba tuvo como finalidad validar que el rastreo de dispositivos en la red se realizara de manera efectiva. Para ello se instalo la aplicación en una computadora personal ubicada en el Centro de Computación, que se encuentra en las instalaciones de la Facultad de Ciencias.

En el ambiente de la prueba se tuvieron los siguientes dispositivos:

- 1 router inalámbrico (Cisco Linksys WRT54G, IP 192.168.0.2).
- Una computadora personal del laboratorio, el cual actuó como servidor donde la aplicación estaba contenida. Dicha computadora personal estaba dentro de la misma red en la cual se encontraba el router TP-LINK mencionado anteriormente.

#### Finalidad de la Prueba

La finalidad de la prueba era validar que el rastreo de dispositivos que se encuentran dentro de la misma red se realizara de manera satisfactoria.

#### Inicio de la Prueba.

Para el inicio de la prueba se coloco en la opción “Escanear Red” de la pagina de administración de punto de acceso, la IP del router. No se coloco ningún

numero en la notación CIDR, lo que le indica a la aplicación que el rastreo solo se realizara únicamente sobre la IP suministrada.



Figura 61. Inicio de la prueba en el Escenario 3.

Luego de seleccionar el botón “Escanear” el resultado obtenido es verificado en el archivo *networkScan.xml* el cual es generado por la aplicación luego del rastreo. Es allí donde el resultado del rastreo se observa (figura 62).

```

8      <xmloutputversion="1.03" />
9      <scaninfo type="connect" protocol="tcp" numservices="100"
10         services=
11         "7,9,13,21-23,25-26,37,53,79-81,88,106,110-111,113,119,135,139,143-144,179,199,389,42
12         ,995,1025-1029,1110,1433,1720,1723,1755,1900,2000-2001,2049,2121,2717,3000,3128,3306,3
13         666,5800,5900,6000-6001,6646,7070,8000,8008-8009,8080-8081,8443,8888,9100,9999-10000,1
14         <host starttime="1305069527" endtime="1305069610">
15             <status state="up" reason="arp-response" />
16             <address addr="192.168.0.2" addrtype="ipv4" />
17             <address addr="00:22:55:42:82:84" addrtype="mac" vendor="Cisco Systems" />
18             <hostnames />
19             <ports>
20                 <extraports state="closed" count="97">
21                     <extrareasons reason="conn-refused" count="97" />
22                 </extraports>
23                 <port protocol="tcp" portid="22">
24                     <state state="open" reason="syn-ack" reason_ttl="0" />
25                     <service name="ssh" method="table" conf="3" />
26                 </port>
27                 <port protocol="tcp" portid="23">
28                     <state state="open" reason="syn-ack" reason_ttl="0" />
29                     <service name="telnet" method="table" conf="3" />
30                 </port>
31                 <port protocol="tcp" portid="80">
32                     <state state="open" reason="syn-ack" reason_ttl="0" />
33                     <service name="http" method="table" conf="3" />
34                 </port>

```

Figura 62. Resultados de las pruebas en el Escenario 3.

En la figura anterior se puede observar que el resultado genera una entrada en el archivo XML en cual define que la IP 192.168.0.3 corresponde a un dispositivo Cisco Series. El resultado mostrado a usuario es el siguiente:

**Resultado del Scanning de la RED**

| Nº | Dirección IP | Dirección MAC     | Distribuidor  | Sistema Operativo  | Opción    |
|----|--------------|-------------------|---------------|--|-----------|
| 1  | 192.168.0.2  | 00:22:55:42:82:84 | Cisco Systems | Cisco Aironet 350, 1100, 1200, or 1131AG WAP; or Cisco Catalyst 2600 router (IOS 12.3) | Registrar |

5 Numero de Registros ⏪ ⏩ Página 1 de 1

Figura 63. Resultado mostrado al usuario en el Escenario 3.

## Conclusiones

El objetivo del presente Trabajo Especial de Grado se cumplió satisfactoriamente, el cual consistió en el desarrollo e implementación de una aplicación Web que permita administrar y monitorear los Puntos de Acceso inalámbricos que conforman la red de la Facultad de Ciencias de la Universidad Central de Venezuela.

El resultado obtenido es una aplicación que cumple con todos los requerimientos planteados al principio de este trabajo. Así mismo, con el cumplimiento de los objetivos, se logró minimizar la dificultad que estas tareas generaban, haciendo las tareas de administración y monitoreo de dichos dispositivos sean más sencillas para los administradores de la red de la Facultad de Ciencias.

Aunque nuevos requerimientos fueron surgiendo durante el desarrollo de la aplicación, se lograron abarcar en su totalidad en el resultado final de este trabajo. Ciertas limitantes se encontraron al momento de realizar el taller de diseño, las cuales fueron las siguientes:

- Al momento de realizar el escáner de la red, la aplicación muestra al Administrador de la Red todos los dispositivos (Computadoras Personales, Impresoras de Red, routers, etc.) que se encuentran en la misma. Queda de la mano del Administrador discernir cual dispositivo agregará para ser administrado.
- Es posible hacer un escáner de una IP en específico o de un grupo de IP's dentro de una subred. Queda de parte del Administrador tener conocimiento sobre la notación CIDR para que pueda definir el rango de IP's en las que desea realizar el rastreo.

El método RAD permitió por medio de iteraciones facilitar el desarrollo de la aplicación, puesto que dio flexibilidad para cubrir e implementar cada uno de los requerimientos, así como el poder realizar las pruebas necesarias para asegurar el funcionamiento efectivo de la aplicación.

El uso de PHP como tecnología del lado del servidor agilizo considerablemente el proceso de desarrollo de la aplicación, ya que se aprovecharon al máximo las estructuras y métodos predefinidos que posee el lenguaje, los cuales minimizaron el tiempo de codificación. El uso de la herramienta NMAP ha sido muy importante a la hora de realizar el rastreo de la red, ya que permite conocer la información necesitada por la aplicación de una manera sencilla.

Se espera que con la implementación de esta aplicación, las tareas de administración y monitoreo de la red de la Facultad de Ciencias se realicen de manera más sencilla y efectiva.

## Referencias Bibliográficas

[1] Alegsa, Diccionario de Informática. Consultado el 10 de Junio de 2011.

<http://www.alegsa.com.ar/Dic/aplicacion%20web.php>.

[2] Somerville, Ian. Ingeniería del Software, séptima edición. Consultado el 22 de Junio de 2011.

[3] Morín, Arley T. Implementación del Patrón clásico de diseño web Modelo-Vista-Controlador (MVC) en PHP. Consultado el 22 de Junio de 2011

<http://arley triana.blogspot.com/2009/07/implementacion-del-patron-clasico-de.html>

[4] Cibernetia. Historia de las aplicaciones web. Consultado el 30 de Junio de 2011.

[http://www.cibernetia.com/manuales/introduccion\\_aplicaciones\\_web/3\\_historia\\_aplicaciones\\_web.php](http://www.cibernetia.com/manuales/introduccion_aplicaciones_web/3_historia_aplicaciones_web.php)

[5] Moreno y Villena, Antonio. Historia y características generales. Consultado el 5 de Julio de 2011.

[http://www.aeemt.com/contenidos\\_socios/Informatica/Informac\\_Informat\\_Tecnolog/AMV\\_AGI\\_AEEMT\\_HTML\\_Historia.pdf](http://www.aeemt.com/contenidos_socios/Informatica/Informac_Informat_Tecnolog/AMV_AGI_AEEMT_HTML_Historia.pdf)

[6] Desarrolloweb. Manual de Hojas de Estilo en Cascada. Consultado el 5 de Julio de 2011.

<http://www.desarrolloweb.com/manuales/manual-css-hojas-de-estilo.html>

[7] GS-ZONE. Definición de JavaScript. Consultado el 15 de Julio de 2011.

<http://www.gs-zone.org/>

[8] McLaughlin, B. Introduction to AJAX. Consultado 18 de Mayo de 2010.

<http://www-128.ibm.com/developerworks/web/library/wa-ajaxintro1.html>.

[9] Manual de Referencia JQuery. Consultado el 22 de Julio de 2011.

<http://jquery.com/>

[10] Manual de Referencia de Apache HTTP Server. Consultado el 26 de Julio de 2011.

<http://apache.org>

[11] Gilmore, W. A Programmer's Introduction to PHP 4.0. Consultado el 15 de Mayo de 2010.

<http://books.google.co.ve/>

[12] Manual de Referencia de MySQL. Consultado el 15 de Julio de 2011.

<http://dev.mysql.com/doc/refman/5.5/en/>

[13] Manual de Referencia de NMAP. Consultado el 26 de Mayo de 2010.

<http://nmap.org>

[14] Cisco Systems. "Internetworking Technologies Handbook". Consultado el 13 de Febrero de 2010.

[http://www.cisco.com/en/US/docs/internetworking/technology/handbook/ito\\_doc.html](http://www.cisco.com/en/US/docs/internetworking/technology/handbook/ito_doc.html).

[15] Kendall, K. y Kendall, J. Análisis y Diseño de Sistema. Sexta Edición. Pearson Education, México