



**UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

**Diseño e implementación de un sistema de seguridad  
de control para servicios de red inalámbrica y acceso  
a RADIUS con GNU/Linux y software libre: el caso  
de una IES.**

**TESIS**

Que para obtener el título de  
**Ingeniero en Computación**

**PRESENTA**

De La Cruz Reyes Eduardo

**DIRECTOR DE TESIS**

Ing. Cruz Sergio Aguilar Díaz



Ciudad Universitaria, Cd. Mx. 3 Junio de 2021



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## ÍNDICE GENERAL

<b>CAPÍTULO 1 ANTECEDENTES</b>	9
1.1 Panorama general	10
1.2 Objetivo general	11
1.3 Finalidad de la aplicación	13
<b>CAPÍTULO 2 IDENTIFICACIÓN DE LA PROBLEMÁTICA</b>	15
2.1 Análisis de hardware y de software	18
2.2 Definición de los alcances del sistema	20
2.3 Definición de los requerimientos	21
<b>CAPÍTULO 3 FUNDAMENTOS TEÓRICOS</b>	23
3.1 Introducción	24
3.2 Software	24
3.3 Software libre	27
3.4 Metodología de desarrollo de proyectos	34
3.5 Bases de datos	51
3.6 Redes de datos	57
3.7 Servidores	72
<b>CAPÍTULO 4 ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DEL SISTEMA</b>	78
4.1 Justificación de las herramientas	79
4.2 Diseño del sistema	86
4.3 Implementación y configuración del servidor	89
4.4 Creación de la base de datos	98
4.5 Población de la base de datos	101
<b>CAPÍTULO 5 DOCUMENTACIÓN Y RESULTADOS</b>	103
5.1 Creación de la Documentación del Sistema.	104
5.2 Manual de uso del sistema.	105
5.3 Resultados del Sistema	105
<b>CONCLUSIONES</b>	112
<b>APÉNDICES</b>	115
<b>APÉNDICE 1. INSTALACIÓN DE MariaDB</b>	116
<b>APÉNDICE 2. CONFIGURACIÓN DEL ACCESS POINT</b>	120
<b>APÉNDICE 3. CONFIGURACIÓN DE SERVIDOR FREERADIUS</b>	123
<b>APÉNDICE 4. COMANDOS BÁSICOS EN BASES DE DATOS</b>	127
<b>APÉNDICE 5. USO DEL SISTEMA</b>	130
<b>GLOSARIO</b>	164
<b>BIBLIOGRAFÍA</b>	177
<b>MESOGRAFÍA (Referencias Electrónicas)</b>	179

## ÍNDICE DE TABLAS

Tabla 1. Población total de Licenciatura en la Facultad	17
Tabla 2. Principales problemas que sugieren usuarios del servicio.	17
Tabla 3. Descripción de equipos utilizados y recomendados	22
Tabla 4. Ventajas, desventajas y requisitos mínimos de instalación.	33
Tabla 5. Conceptos básicos de base de datos.	51
Tabla 6. Datos de Médicos que se relacionarán con la tabla Ingresos	53
Tabla 7. Tabla Hospital relacionada con Tabla Médicos mediante su código.	54
Tabla 8. Características de cables de conexión	58
Tabla 9. Características de redes existentes.	62
Tabla 10. Características físicas de equipos más utilizados.	71
Tabla 11. Comparativa de algunos Access Point disponibles en el mercado	80
Tabla 12. Características de instalación básica en distribuciones Linux	84

## ÍNDICE DE FIGURAS

Figura 1. Pasos para una autenticación mediante un Access Point y RADIUS.	14
Figura 2. Funcionamiento básico de un Access Point.	19
Figura 3. Software de acuerdo con el tipo de trabajo realizado	25
Figura 4. Software de acuerdo con el método de distribución	26
Figura 5. Ejemplos de software libre	28
Figura 6. Sistemas operativos de distribución libre.	28
Figura 7. Sistemas operativos de servidores más utilizados	32
Figura 8. Ciclo de Agile	35
Figura 9. Ejemplo de un tablero Kanban	40
Figura 10. Ciclo de vida de la metodología XP	46
Figura 11. Ciclo del modelo en cascada	47
Figura 12. Modelo Agile utilizado en el proyecto.	50
Figura 13. Bases de datos NoSQL y Relacionales	56
Figura 14. Medios de transmisión de una red	57
Figura 15. Medios físicos de conexión en una red	59
Figura 16. Red PAN	59
Figura 17. Red LAN	60
Figura 18. Red MAN	60
Figura 19. Red GAN	61
Figura 20. Red WAN	61
Figura 21. Topología estrella.	63
Figura 22. Topología de árbol.	64
Figura 23. Topología de anillo.	64
Figura 24. Topología de bus.	65
Figura 25. Topología de malla.	66
Figura 26. Topología híbrida.	67
Figura 27. Router inalámbrico marca Cisco, modelo RV110W	68
Figura 28. Repetidor marca Cisco Linksys, modelo RE1000	69
Figura 29. Hub marca Cisco Linksys USB de 4 puertos	69
Figura 30. Bridge marca Cisco Linksys, modelo WES610N	70
Figura 31. Switch marca Cisco Fast Ethernet modelo SF250-24	70
Figura 32. Arquitectura básica cliente-servidor	72

<b>Figura 33. Tipos de servidores</b>	77
<b>Figura 34. Servidor FreeRADIUS.</b>	81
<b>Figura 35. Herramientas de daloRadius.</b>	82
<b>Figura 36. Sistema operativo Debian</b>	85
<b>Figura 37. Elementos a utilizar en el sistema</b>	87
<b>Figura 38. Diagrama general del funcionamiento de red usando servidor Radius</b>	88
<b>Figura 39. Creación de la Base de datos</b>	89
<b>Figura 40. Verificación de la base de datos creada</b>	90
<b>Figura 41. Descarga e instalación de freeRadius</b>	90
<b>Figura 42. Descarga e instalación de freeRadius</b>	91
<b>Figura 43. Verificación de estado de freeRadius</b>	91
<b>Figura 44. Importación de esquema de BD.</b>	91
<b>Figura 45. Creación de enlace simbólico.</b>	92
<b>Figura 46. Comando para cambios de parámetros de conexión</b>	92
<b>Figura 47. Configuración final de archivo</b>	93
<b>Figura 48. Comandos para asignación de permisos</b>	93
<b>Figura 49. Comando para reinicio de servidor</b>	93
<b>Figura 50. Configuración de archivo users</b>	93
<b>Figura 51. Creación de usuario en archivo users</b>	94
<b>Figura 52. Reinicio del servidor freeRadius</b>	94
<b>Figura 53. Prueba de autenticación de usuario en local</b>	94
<b>Figura 54. Descarga de archivo de daloRadius</b>	95
<b>Figura 55. Comando para desempaquetar archivo</b>	95
<b>Figura 56. Modificación de nombre y acceso a directorio</b>	95
<b>Figura 57. Importación de tablas a daloRadius</b>	96
<b>Figura 58. Cambio de ruta del directorio daloRadius</b>	96
<b>Figura 59. Asignación de permisos al directorio</b>	96
<b>Figura 60. Comando para configurar conexión a base de datos</b>	96
<b>Figura 61. Asignación de parámetros para conexión</b>	96
<b>Figura 62. Reinicio de servicios</b>	97
<b>Figura 63. Interfaz inicial de herramienta daloRadius</b>	97
<b>Figura 64. Página inicial de acceso a daloRadius</b>	97
<b>Figura 65. Creación de la base de datos</b>	98
<b>Figura 66. Acceso al Sistema Gestor de Bases de Datos</b>	98
<b>Figura 67. Acceso a la base de datos</b>	99
<b>Figura 68. Creación de tabla administradores</b>	100
<b>Figura 69. Creación de tabla alumnos</b>	100
<b>Figura 70. Inserción de datos en tabla administradores</b>	101
<b>Figura 71. Comprobación de inserción de datos exitosa</b>	101
<b>Figura 72. Inserción de datos en tabla alumnos</b>	102
<b>Figura 73. Comprobación de inserción de datos exitosa</b>	102
<b>Figura 74. Ingreso de datos para alta de usuario</b>	106
<b>Figura 75. Ingreso de datos para alta de usuario</b>	106
<b>Figura 76. Alta de usuario exitosa.</b>	107
<b>Figura 77. Ingreso de datos para modificación de usuarios.</b>	107
<b>Figura 78. Modificación de usuario exitoso.</b>	108
<b>Figura 79. Ingreso de usuario para deshabilitación.</b>	108
<b>Figura 80. Deshabilitación de usuario exitoso.</b>	109

<b>Figura 81. Ingreso de usuario para eliminación.</b>	109
<b>Figura 82. Eliminación de usuario exitosa.</b>	110
<b>Figura 83. Ingreso de datos para conexión a la red mediante Access Point.</b>	110
<b>Figura 84. Conexión exitosa de usuario al Access Point</b>	111
<b>Figura 85. Descarga e instalación de la base de datos MariaDB</b>	117
<b>Figura 86. Modificación de datos por defecto de MariaDB</b>	118
<b>Figura 87. Eliminación de usuarios anónimos.</b>	118
<b>Figura 88. Desactivación de acceso remoto</b>	119
<b>Figura 89. Eliminación de base de datos de prueba.</b>	119
<b>Figura 90. Configuración completa de la base de datos</b>	119
<b>Figura 91. Inicio de sesión del Access Point</b>	121
<b>Figura 92. Configuración de la IP estática para usarse en el Access Point</b>	121
<b>Figura 93. Configuración de datos del servidor freeRadius en el Access Point</b>	122
<b>Figura 94. Asignación de IP estática en el servidor freeRadius</b>	124
<b>Figura 95. Llenado de tabla NAS con datos del Access Point</b>	124
<b>Figura 96. Gráfica de la sección de usuarios</b>	125
<b>Figura 97. Creación de un nuevo usuario</b>	125
<b>Figura 98. Comprobación del usuario agregado</b>	126
<b>Figura 99. Pantalla para inicio de sesión</b>	131
<b>Figura 100. Pantalla de inicio de sesión incorrecto.</b>	131
<b>Figura 101. Menú después de inicio de sesión</b>	132
<b>Figura 102. Menú después de inicio de sesión</b>	132
<b>Figura 103. Inicio de sesión con usuario más alto.</b>	133
<b>Figura 104. Menú al iniciar sesión con usuario más alto.</b>	133
<b>Figura 105. Inicio de sesión con usuario intermedio.</b>	134
<b>Figura 106. Menú al iniciar sesión con usuario intermedio.</b>	134
<b>Figura 107. Inicio de sesión con usuario más bajo.</b>	135
<b>Figura 108. Menú al iniciar sesión con usuario más bajo.</b>	135
<b>Figura 109. Tabla de usuarios conectados en el día.</b>	136
<b>Figura 110. Tabla con registro de veces conectadas por un usuario.</b>	136
<b>Figura 111. Ejemplo de alta de usuario.</b>	137
<b>Figura 112. Ejemplo de alta de usuario.</b>	137
<b>Figura 113. Alta de usuario exitoso.</b>	138
<b>Figura 114. Verificación de alta de usuario.</b>	138
<b>Figura 115. Menú para la opción Modificar Usuario.</b>	139
<b>Figura 116. Instrucciones para modificación de usuario y/o contraseña.</b>	139
<b>Figura 117. Campos para modificación de usuario y/o contraseña.</b>	140
<b>Figura 118. Ejemplo de modificación de usuario.</b>	140
<b>Figura 119. Modificación de usuario exitoso.</b>	141
<b>Figura 120. Verificación de usuario antes de ser modificado.</b>	141
<b>Figura 121. Verificación de usuario después de ser modificado.</b>	142
<b>Figura 122. Verificación de contraseña de usuarios desde terminal.</b>	142
<b>Figura 123. Ejemplo de modificación de contraseña.</b>	143
<b>Figura 124. Cambio de contraseña exitosa.</b>	143
<b>Figura 125. Verificación de cambio de contraseña de usuarios desde terminal.</b>	144
<b>Figura 126. Ejemplo de cambio de usuario y contraseña.</b>	144
<b>Figura 127. Cambio de usuario y contraseña exitosa.</b>	145
<b>Figura 128. Verificación de cambio de usuario y contraseña desde terminal.</b>	145

<b>Figura 129. Búsqueda de usuario por número de cuenta.</b>	146
<b>Figura 130. Muestra de resultados obtenidos.</b>	146
<b>Figura 131. Modificación de apellido materno</b>	147
<b>Figura 132. Modificación de apellido materno exitoso.</b>	147
<b>Figura 133. Confirmación de cambio de apellido materno.</b>	148
<b>Figura 134. Búsqueda de usuario por usuario.</b>	148
<b>Figura 135. Muestra de resultados obtenidos.</b>	149
<b>Figura 136. Modificación de cuenta exitosa.</b>	149
<b>Figura 137. Confirmación de cambio de cuenta exitoso.</b>	150
<b>Figura 138. Búsqueda de usuario por número de cuenta.</b>	150
<b>Figura 139. Muestra de resultados obtenidos.</b>	151
<b>Figura 140. Modificación de semestre exitoso.</b>	151
<b>Figura 141. Confirmación de cambio de semestre exitoso.</b>	152
<b>Figura 142. Búsqueda de usuario por número de cuenta.</b>	152
<b>Figura 143. Muestra de resultados obtenidos.</b>	153
<b>Figura 144. Modificación de carrera exitosa.</b>	153
<b>Figura 145. Verificación de cambio de carrera exitosa.</b>	154
<b>Figura 146. Menú inicial.</b>	154
<b>Figura 147. Página para habilitar / deshabilitar usuarios.</b>	155
<b>Figura 148. Tabla con usuarios bloqueados.</b>	155
<b>Figura 149. Ejemplo de habilitación de usuario.</b>	156
<b>Figura 150. Error en habilitación de usuario.</b>	156
<b>Figura 151. Habilitación de usuario exitoso.</b>	157
<b>Figura 152. Verificación de usuario deshabilitado.</b>	157
<b>Figura 153. Ejemplo para deshabilitar un usuario.</b>	158
<b>Figura 154. Deshabilitar un usuario de manera exitosa.</b>	158
<b>Figura 155. Verificación para desabilitar un usuario.</b>	159
<b>Figura 156. Menú inicial.</b>	159
<b>Figura 157. Página para eliminación de usuarios.</b>	160
<b>Figura 158. Selección de usuario a eliminar.</b>	160
<b>Figura 159. Llenado de campo con usuario a eliminar.</b>	161
<b>Figura 160. Eliminación de usuario exitoso.</b>	161
<b>Figura 161. Comprobación de usuario eliminado.</b>	162
<b>Figura 162. Menú inicial.</b>	162
<b>Figura 163. Página de inicio de sesión.</b>	163

# INTRODUCCIÓN



En el mundo actual, el acceso a Internet se ha vuelto algo vital, prácticamente para cualquier individuo que cuente con algún dispositivo móvil. Nos encontramos en una era donde cualquier información que se necesite, puede ser consultada de manera inmediata a través de los llamados datos móviles o mediante wifi.

Y el uso que se le da hoy en día es muy variado, ya sea para buscar noticias, ver videos, realizar compras por Internet, realizar algún trabajo de investigación, navegar por las redes sociales, etcétera. Con ello, viene la contraparte dada por un incremento masivo de usuarios conectados a Internet y la necesidad de administrarlos adecuadamente para que no exista una sobrecarga de los mismos en los llamados servidores y se le pueda brindar a todos y cada uno de ellos, un servicio eficaz, de conexión rápida, pero sobre todo seguro.

Es por esta razón que los desarrollos de las nuevas tecnologías dedican gran parte de su tiempo a la seguridad y administración de las personas que las utilizan, aplicando diferentes métodos como lo son autenticación, autorización, encriptación, entre otros más, para evitar que la información que viaje a través de esta pueda verse comprometida y con ello se utilice para algún otro fin, ya sea fraudulento, para exponer cosas personales o simplemente por malicia.

Existen varias maneras de realizar esto y también varios fabricantes que han puesto su esfuerzo en ello, existen de igual manera aquellos que son gratuitos y aquellos que tienen un costo, ya sea bajo o elevado. Para lograr lo descrito, en este trabajo se hace uso de un servidor Radius (del inglés Remote Access Dial in User Service), que ofrece un mecanismo de seguridad, flexibilidad, capacidad de expansión y una administración simplificada de las credenciales de acceso a un recurso de red.

Se utiliza un esquema cliente-servidor, esto quiere decir que un usuario con credenciales se conecta contra un servidor que será el encargado de verificar la autenticidad de los datos y determine si el usuario puede o no acceder. Muchos operadores de red hacen uso de esto, pero existen otros lugares como lo son hoteles, aeropuertos, bibliotecas, etcétera. donde puede hacerse uso de esta herramienta.

En el capítulo 1, se describe el panorama actual de la red inalámbrica dentro de la Facultad de Ingeniería, asimismo el objetivo a alcanzar y los elementos a utilizar, para finalmente describir de forma general, como funcionará el servidor.

En el capítulo 2, se describen los elementos a utilizar tanto de software como de hardware, se especifica hacia quiénes está dirigido el proyecto, los pasos a realizar y el tiempo estimado.

En el capítulo 3, se hace una descripción de conceptos vistos en materias durante la carrera, qué son y qué elementos de hardware o software son los que se utilizan, los cuales serán de ayuda en la realización del proyecto.

En el capítulo 4, se realiza la justificación del porqué se utilizarán las herramientas descritas en el capítulo 2, cómo estará compuesto el sistema y se mostrará la instalación y configuración del servidor.

En el capítulo 5, se llevará a cabo la realización de la documentación del sistema, incluyendo el manual de operación del mismo y su respectiva evidencia del funcionamiento correcto.

# **CAPÍTULO 1**

# **ANTECEDENTES**

## 1.1 Panorama general

En la actualidad el uso de las redes de datos y en específico del wifi (wireless fidelity) se ha vuelto una necesidad primordial, es por ello que actualmente la UNAM (Universidad Nacional Autónoma de México) brinda a los estudiantes dos tipos de servicios, el primero de ellos es el acceso a salas de computación, aquí los estudiantes pueden tener acceso a equipos conectados a internet para realizar únicamente sus actividades escolares, asimismo si se requiere de algún software adicional, este puede ser solicitado para su instalación, en estas salas no se requiere algún tipo de registro previo, su acceso es únicamente presentando la credencial vigente de la Facultad.

Por otro lado, se cuenta con una red inalámbrica, conocida como RIU (Red Inalámbrica Universitaria), esta red, brinda a los alumnos la oportunidad de tener acceso a una red de Internet gratuita dentro de la institución que les permita garantizar la conectividad con un ancho de banda aceptable para realizar sus actividades escolares. Para poder acceder a esta red, solo basta con acudir a la DGTIC (Dirección General de Cómputo y de Tecnologías de la Información) a darse de alta, con lo cual se les brindará un usuario y contraseña la cual podrán utilizar en cualquier lugar dentro de Ciudad Universitaria donde se localice alguna señal RIU y así poder conectarse.

Sin embargo, debido a que este tipo de red es a nivel campus, es decir, con un mismo usuario y contraseña uno puede conectarse desde cualquier punto de Ciudad Universitaria, sin importar en qué Facultad, Posgrado o Biblioteca uno se encuentre, esto conlleva a que muchas veces exista una saturación de usuarios en un mismo punto wifi. En el caso particular de Facultad de Ingeniería, los alumnos que estudien en esta facultad podrán conectarse a la RIU si cumplen con las características antes mencionadas, al igual que otros alumnos o profesores que vengan de otros sitios, es por ello que muchas veces es casi imposible poder conectarse a esta red, y en caso de poder hacerlo, la velocidad de navegación es muy baja y en casos extremos existe intermitencia en el servicio.

Es por ello que se busca desarrollar una red inalámbrica de acceso controlado a nivel facultad, cuyo propósito es que solamente pueda ser utilizada por alumnos que estudien aquí, y que al mismo tiempo no se encuentre saturada por tantos dispositivos, esto se logrará, teniendo la limitante de que un mismo usuario no podrá tener más de 3 sesiones activas (dispositivos conectados), ya sea al mismo tiempo o en diferentes momentos durante todo el día, esto se logrará mediante el alta de los usuarios (estudiantes) en el sistema, los datos serán guardados en una base de datos, y se les brindará un usuario y contraseña para que puedan obtener acceso al sistema.

## 1.2 Objetivo general

A partir de una encuesta realizada a estudiantes de la Facultad de Ingeniería (la cual podemos encontrar más adelante en el capítulo 2 Identificación de la problemática) donde se destacan los distintos problemas que afectan el desempeño diario de las actividades de los alumnos a la hora de utilizar las salas de cómputo, así como a la hora de utilizar la Red Inalámbrica Universitaria existente (RIU), observamos que no solo existe un problema de saturación de red inalámbrica, sino que los equipos de las salas actuales en su mayoría, conforme avanza el tiempo van siendo obsoletos en el sentido de que la gran mayoría de ellos, llevan años siendo los mismos, con el mismo sistema operativo, misma capacidad de memoria, mismos monitores, y en muchos casos, también existe una falta de mantenimiento adecuado.

Es por ello que ante la existencia de estas dos opciones con las que cuentan los alumnos, de las cuales una de ellas únicamente sirve a alumnos de Ingeniería (salas de cómputo) y la otra puede ser usada tanto por alumnos de Ingeniería como por personas ajenas a la Facultad de Ingeniería que previamente hayan tramitado su usuario y contraseña ante la DGTIC y que tengan alguna relación con la UNAM, por lo cual, estén donde estén, pueden acceder a la RIU al contar con sus accesos. Por lo tanto, lo que se busca con este proyecto de tesis es, en simples palabras, dotar a los alumnos de una red inalámbrica que cuente con un mayor control sobre los usuarios para así detectar a aquellos que compartan credenciales saturando la red, esto es, impedir que un mismo nombre de usuario aparezca conectado a la red más de 3 veces (que podrían ser más de 3 dispositivos), ya que esto nos indicaría que hay diferentes alumnos con 3 o más dispositivos conectándose a la red utilizando las credenciales de algún compañero.

Otro punto es permitir el acceso a Internet a los alumnos con dispositivos móviles, sean laptops, tabletas o celulares, en áreas de la Institución que no se dispone normalmente de red cableada. Es decir, implementar un servicio de red inalámbrica mediante un Servidor Radius que permita a los estudiantes garantizar la conectividad desde los dispositivos de su preferencia, para que puedan complementar su quehacer académico en el lugar de su preferencia dentro de las instalaciones, evitando el tiempo de traslado y de asignación de una máquina dentro de las salas de cómputo o tener que compartir la señal de red inalámbrica con personas que provengan de otras Facultades o incluso personas ajenas a la UNAM (Universidad Nacional Autónoma de México), pero que al conocer a alguien, pueden utilizar sus credenciales para obtener acceso, y tomando en cuenta que una persona externa a la Facultad o a la UNAM no necesita y no debería de tener acceso a la red.

Se busca crear este sistema de seguridad de control para servicios de red inalámbrica para complementar las opciones de conexión de red existente, no para sustituirla, y esto lo podemos observar ya que el alumno seguirá conservando sus credenciales perteneciente a la RIU, para que pueda seguirse conectando a esa red estando fuera de la Facultad de Ingeniería. El objetivo no es competir con la RIU, sino solamente brindar a los alumnos una red más “privada”, de la cual ellos estarán consientes que los únicos con acceso a ella, serán los mismos alumnos, buscando eliminar a personas ajenas. Sin embargo, los alumnos tendrán completa libertad para elegir la red de su preferencia.

Y más allá de los beneficios que esto busca traer para los estudiantes, se busca hacer uso de herramientas de bajo costo para reducir gastos a la hora de su implementación, sin dejar a un lado que esta red deberá ser lo más eficaz posible aunque se haga uso de herramientas que no sean precisamente de pago. Por ende, los objetivos más destacables en la creación de este sistema son:

- Realizar la implementación de un servidor de bajo costo utilizando software libre y un Access Point.
- Controlar el acceso a la red mediante la autenticación de los usuarios.
- Presentar un servidor con buena conexión de Internet al reducir los usuarios.

Conforme se avance en los capítulos de este proyecto de tesis, se irá profundizando en cada uno de los puntos anteriores, se analizarán cada uno de los componentes de los cuales se compondrá este nuevo sistema, primero de manera individual para posteriormente analizarlos de forma conjunta, tanto de forma escrita como de forma visual. Si bien el proyecto puede ser más fácil de entender para alguna persona con conocimientos previos de los temas aquí mencionados o para algún profesional de la Ingeniería, el objetivo de cada capítulo es que el lector, sea quien sea, al leer este proyecto entienda por qué se está creando este sistema, hacia quién va dirigido en específico y que comprenda cómo es que funcionan cada uno de los elementos para lograr el objetivo deseado y de ser posible, pueda tomar algunas ideas para servir como base en futuros proyectos, ya sea dentro de la misma Universidad o en alguna otra Universidad o Institución de carácter pública o privada.

### 1.3 Finalidad de la aplicación

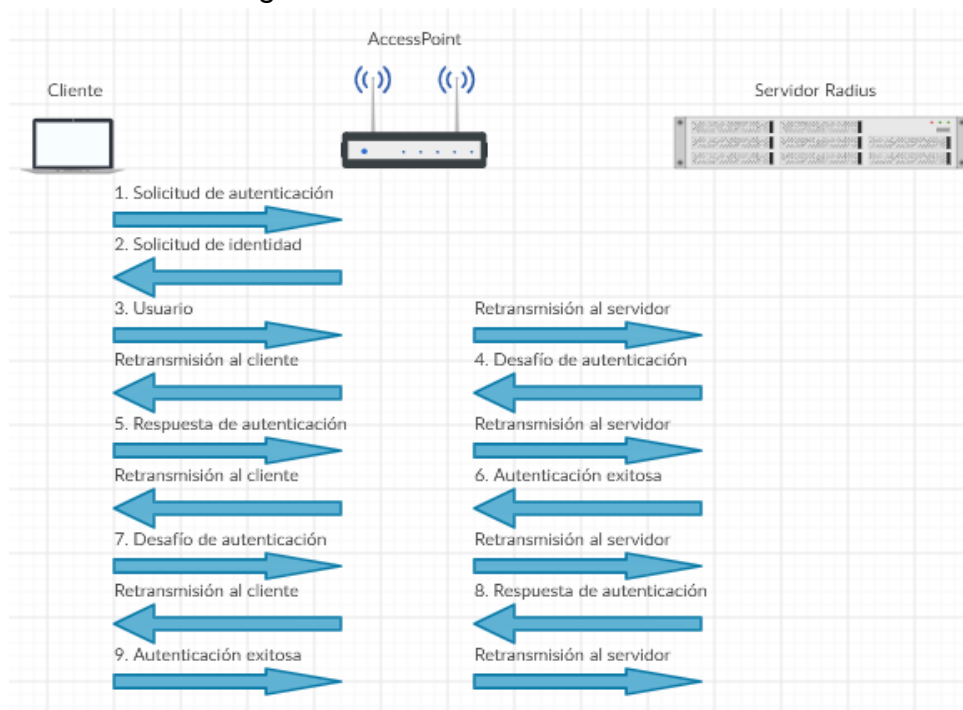
Realizar la configuración de un Servidor Radius, utilizando un servidor de software libre (freeRadius), junto con la creación de una base de datos (MariaDB), y un sistema operativo Linux (Debian) para realizar la autenticación de los usuarios. Además, un Access Point para que la conexión a la red pueda hacerse de forma inalámbrica, funcionando como un complemento de la red actual RIU y de las salas de cómputo, en el caso de la RIU con la diferencia de que esta red solo estará disponible para alumnos de la Facultad de Ingeniería que cuenten con su usuario y contraseña, la cual se les brindará al darse de alta en el sistema. Y en el caso de las salas de cómputo evitar los desplazamientos y tiempos de espera para que una computadora pueda ser asignada, así como seguirles brindando a los alumnos la libertad de ocupar sus propios equipos pero esta vez usando una red más a nivel facultad.

De este modo se contará con una mayor seguridad al evitar que otra persona pueda tomar prestada las credenciales de algún otro alumno registrado para acceder con ellas a la red y se dé el caso de que un mismo usuario se conecte 3 o más veces a la red y vuelva la misma problemática de antes, que es la saturación de la red actual, debido a la gran cantidad de usuarios conectados desde sus equipos. Con ello, al mantener un monitoreo constante de los usuarios que se conectan a la red inalámbrica, se podrá tener un mayor orden para así conocer que alumno es el que cae en acciones indebidas al conceder su usuario y contraseña a otros compañeros para que estos eviten el trámite para darse de alta con sus propios datos.

Por lo tanto, la finalidad de este proyecto de tesis es brindarle al alumno otra opción de conectividad, ya que muchas veces por la desidia de ir a tramitar sus accesos a la DGTIC, prefieren entrar a las salas de cómputo sin importar las problemáticas mencionadas así como sin tener que estar lidiando con problemas de desconexión en la red inalámbrica existente, y esta vez, al tener un monitoreo constante del sistema y poder observar que alumno concede sus credenciales a otros, existiría un bloqueo temporal de sus accesos al sistema, con el cual el alumno deberá presentarse para aclarar la situación y en dado caso, volver a habilitar sus accesos, y en caso de reincidir, podría existir la expulsión definitiva del sistema. Esto conllevaría a un compromiso por parte de los alumnos a utilizar la red inalámbrica de una manera más responsable para que de esta manera todos los alumnos que la utilicen puedan realizar sus actividades.

A continuación, se describe en términos generales el proceso de interacción entre el usuario con el Access Point y el Access Point con el Servidor (Véase Figura 1).

- 1) El cliente (en nuestro caso sería el dispositivo del alumno), nota la red disponible (en este caso el nombre del Access Point) y solicita conectarse.
- 2) El Access Point recibe la solicitud por parte del cliente y solicita las credenciales de acceso.
- 3) El cliente introduce sus credenciales de acceso, y el Access Point transmite estos datos hacia el servidor.
- 4) El servidor inicia lo que se conoce como un desafío de autenticación hacia el cliente pasando por el Access Point, el cual consiste básicamente en una “pregunta” en específico para ese cliente.
- 5) El cliente envía la respuesta de autenticación a la “pregunta” formulada por el servidor, pasando de nueva cuenta a través del Access Point.
- 6) Si la respuesta es la esperada por el servidor, entonces se realiza una autenticación exitosa, que va de vuelta al cliente.
- 7) Ahora el cliente inicia su propio desafío de autenticación que es transmitido hacia el servidor.
- 8) El servidor envía su respuesta hacia el cliente.
- 9) Si ambos desafíos son exitosos, el cliente puede acceder a la red y comenzar a navegar.



**Figura 1. Pasos para una autenticación mediante un Access Point y RADIUS.**  
**Fuente: Imagen diseñada en Internet<sup>1</sup>**

<sup>1</sup> Imagen realizada con:  
 app.creately.com



# **CAPÍTULO 2**

# **IDENTIFICACIÓN**

# **DE LA**

# **PROBLEMÁTICA**

Actualmente, la Facultad de Ingeniería cuenta con dos salas de cómputo, las cuales sirven de apoyo para la realización de las actividades de los alumnos, sin embargo, al llevarse a cabo una encuesta que englobó a las 15 carreras de Licenciatura y un Posgrado en donde se imparten maestrías y doctorados en 8 campos del conocimiento, además de 12 especialidades en diferentes áreas, las cuales en su totalidad (hasta el año 2018) conforman una población cercana a los 15000 alumnos (Véase Tabla 1). Para identificar los distintos problemas que afectan el desempeño diario de sus actividades en estas salas, se les solicitó que proporcionarían según su experiencia, los motivos por los cuales se abstendrían de utilizar las salas de cómputo y/o la RIU, mencionando tales como que los equipos cada vez son más obsoletos, que las condiciones para permanecer en las salas no es el adecuado (hace mucho calor), que los equipos tienen virus, que el tiempo de asignación de un equipo es tardado debido a que ya se encuentran ocupados, que el internet es muy lento o que el proceso de impresiones es muy tardado (Véase tabla 2), siendo el principal motivo por que los alumnos cada vez asisten menos a estas salas es que los equipos cada vez son más obsoletos, es decir, no cumplen con las características deseadas, o que al estar ocupando el equipo, muchas veces se congele la pantalla o definitivamente no responda y por parte de la RIU que el Internet es demasiado lento.

Por otro lado, dentro de la UNAM, se cuenta con una red de Internet global conocida como RIU, es decir, cualquier persona que se encuentre dentro del campus, puede tener acceso a ella, usando su usuario y contraseña (la cual fue tramitada previamente mediante DGTIC). Existen diversos puntos dentro del campus para acceder a ella, ya sea estando en bibliotecas, comedores, facultades, posgrados, incluso estando en áreas verdes.

Esto ha sido otra gran opción para alumnos en el desarrollo de proyectos y/o tareas a lo largo de su carrera, sin embargo, como se mencionó, este tipo de red es a nivel campus, con lo cual si alguna otra persona proveniente de otra facultad o posgrado ingresara a la Facultad de Ingeniería esta podría conectarse a la RIU debido a que contaría con el usuario y contraseña necesarios para realizar la conexión. Esto conlleva una problemática, ya que la red se satura si ingresan más usuarios que los propios alumnos que existen en la Facultad, y por ende la velocidad de la red disminuye y en casos extremos, existe pérdida de conexión de los dispositivos, para finalmente no poder llevar a cabo las actividades.

Entonces si tomamos en cuenta estas dos opciones con las que cuentan los alumnos, podemos observar que por un lado, en las salas los equipos ya no son los adecuados según la perspectiva de los alumnos y por el otro, la RIU se satura muy rápido haciendo que muchas veces los alumnos no puedan ni siquiera conectarse. Es por ello que ante esta problemática, al crear este sistema de seguridad de control para servicios de red inalámbrica, brindaremos a los alumnos de una opción más para que puedan ser libres de elegir los dispositivos desde los

cuales quieran trabajar y a su vez contar con una red inalámbrica mucho más estable al reducir drásticamente el número de usuarios conectados ya que los registros y el uso de la red estará limitado únicamente a alumnos vigentes dentro de la Facultad.

*Tabla 1 Población total de Licenciatura en la Facultad*

Ciclo Escolar	Primer Ingreso			Reingreso			Total
	Hombres	Mujeres	Total	Hombres	Mujeres	Total	
2014-2015	1,974	570	2,544	8,227	2,166	10,393	12,937
2015-2016	1,813	594	2,407	8,448	2,279	10,727	13,134
2016-2017	1,778	581	2,359	8,298	2,376	10,674	13,033
2017-2018	1,822	588	2,410	7,561	2,26	9,821	12,231

Fuente: Dirección General de Planeación<sup>2</sup>.

*Tabla 2 Principales problemas que sugieren usuarios del servicio.*

Disminución de calidad de servicios de cómputo	Datos recolectados	Porcentaje	Porcentaje acumulado
Los equipos ya están obsoletos / Pocas características.	112	57.44%	57.44%
Hace mucho calor en las Salas / No sirve el aire acondicionado.	34	17.44%	74.87%
Los equipos a veces tienen virus.	13	6.67%	81.54%
No te permiten la entrada sin credencial o Credencial sin resello.	10	5.13%	86.67%
No cuentan con el software adecuado / No te permiten instalar.	8	4.10%	90.77%
Las impresiones se tardan mucho / Pocas impresiones.	7	3.59%	94.36%
Los equipos están ocupados o en curso.	5	2.56%	96.92%
El Internet es muy lento / Trabajo con RIU	4	2.05%	98.97%
Deberían de tener Scanner e impresoras de color.	2	1.03%	100.00%

Fuente: Encuestas aplicadas a los alumnos en Salas de Cómputo en 2019.

<sup>2</sup> Dirección General de Planeación, UNAM. Página: <https://www.planeacion.unam.mx/>

## **2.1 Análisis de hardware y de software**

A continuación se realizará una breve descripción de los elementos de hardware y software posibles a utilizar en la creación del sistema de control de red inalámbrica. Se mencionará brevemente qué son y cómo es que funcionan, así como algunas de sus características más relevantes. Cabe destacar que estos elementos podrían variar en función de las necesidades que se presenten más adelante, es decir, se puede llegar a ocupar otro software o hardware más sofisticado que los que se mencionan a continuación.

### **ANÁLISIS DE HARDWARE**

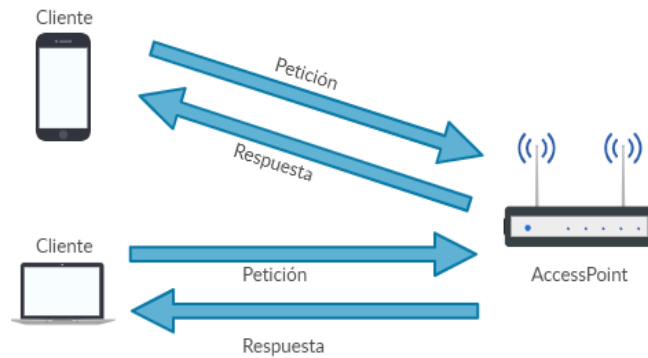
Para llevar a cabo la implementación de este proyecto se requiere de un Access Point que permita llevar el control de los dispositivos que se conectaran (en conjunto con un servidor) mediante unas credenciales de acceso. Las características y funcionalidades de este dispositivo se muestran de forma breve a continuación.

#### **Access Point**

Un Access Point es un dispositivo que permite que dos o más computadoras se comuniquen sin cables para compartir información. Estos dispositivos se encargan de administrar el acceso a archivos de datos, música, fotografías e incluso el mismo servicio de internet que pueden compartir varios dispositivos de forma simultánea como, por ejemplo: un Smart TV, computadora, tableta, Smartphone etcétera (Véase Figura 2). Normalmente un Access Point crea una red inalámbrica pero también puede conectarse a una red cableada, y pueden transmitir datos entre los dispositivos conectados a la red cableada y los dispositivos inalámbricos.

El Access Point se puede conectar a un Router o Switch que proyecta una señal wifi en un área designada, un solo Access Point puede soportar a varios usuarios y pueden funcionar en un rango de 30 hasta más de 100 metros. La tecnología de comunicación con que cuentan se conforma por ondas de radio capaces de traspasar muros, sin embargo, entre cada obstáculo esta señal puede perder la fuerza y reducir su cobertura.

Para este caso, haremos uso de un Access Point, el cual en conjunto con nuestro servidor nos permitirá conceder o denegar un acceso a un usuario a la red.



**Figura 2. Funcionamiento básico de un Access Point.**  
 Fuente: Imagen diseñada en Internet<sup>3</sup>

## ANÁLISIS DE SOFTWARE

Para la implementación del servidor se requiere de un sistema operativo que permita la instalación y configuración de freeRadius junto con una base de datos para el almacenamiento de la información de los alumnos de la Facultad de Ingeniería.

### a) Servidores

En cuanto a la parte de Servidores, al igual que con la parte de los Sistemas Operativos, Windows y Linux son de los más utilizados. Windows con su famoso Windows Server el cual es un software privado, lo que implica que su código no es público y las actualizaciones dependen de la empresa, que puede o no cobrar por ellas. Existen varias versiones o distribuciones de Windows Server, entre las cuales se encuentran Windows 2000 Server, Windows Server 2012, Windows Essential Business Server o Windows Server 2016. Por el lado de Linux, existen un poco más de versiones, entre las cuales aparecen Ubuntu Server, Red Hat Enterprise Linux Server, Debian, Fedora Server, CentOS, etcétera.

Como se podrá observar, no todos llevan la palabra 'Server' y esto es porque no son como tal un sistema dedicado única y exclusivamente a atender servidores, sino más bien, sistemas operativos capaces de manejar servidores. Para este caso en específico se trabajará en conjunto con algo conocido como Remote Authentication Dial-In User Service o RADIUS Server, este servidor en especial nos permitirá realizar autenticaciones de los usuarios antes de que estos se puedan conectar a la red. Todo esto sin perder de vista que siempre buscaremos la parte de reducción de costos, y que en este caso lo lograremos utilizando el ya

<sup>3</sup> Imagen realizada con:  
 app.creately.com

mencionado Radius Server, ya que al igual que el sistema operativo y la base de datos a utilizar, el uso de este servidor será totalmente libre.

## **b) Bases de datos**

Para la parte de Bases de Datos, hoy en día existen infinidad de opciones, entre las Base de Datos más utilizadas existen las relacionales y las no relacionales (NoSQL), ambas con sus ventajas y sus desventajas si hablamos de eficiencia, rendimiento, escalabilidad, complejidad, etcétera. Algunos ejemplos de bases de datos relacionales son MySQL, MariaDB, SQLite, PostgreSQL, Microsoft SQL y Oracle. Por el lado de las bases de datos no relacionales tenemos como ejemplos a MongoDB, Redis y Cassandra, por solo mencionar las más populares.

Dependiendo de la cantidad de datos a manejar y a almacenar, asimismo como de la relación que exista entre la información almacenada (dependencia entre tablas) será como se elija una u otra base de datos. Para el proyecto a desarrollar, al no existir demasiadas tablas que se deban de relacionar unas con otras ni tampoco cantidades enormes o complejas de información, no existiría la necesidad de pagar por licencias para acceder a características muy complejas de la base de datos. Como se ha mencionado, este proyecto busca ser lo más reducido en costos posible, así mismo se seguirá esta filosofía al escoger una base de datos que no requiera el pago de alguna licencia o de complejas configuraciones para operar, también que los requerimientos de software no sean tan exigentes para la instalación de la misma, todo esto sin perder de vista el hecho de lo estable que tenga que ser la base de datos para evitar en lo más posible fallas a la hora de acceder a los datos al trabajar en conjunto con nuestro servidor.

## **2.2 Definición de los alcances del sistema**

Se busca realizar la implementación de servicios de red inalámbricos mediante un Servidor Radius para la comunidad estudiantil, tomando en cuenta que se utilizará únicamente software libre, esto con la finalidad de reducir costos y poder dotar a los estudiantes de una red segura, se empezará por definir la situación actual de la red dentro de la institución, después se buscarán las herramientas de software libre a utilizar, se realizará la instalación y configuración de las mismas y finalmente se realizarán las pruebas finales para verificar que el servidor se encuentre activo y funcionando correctamente, todo esto dentro de un lapso máximo de seis meses.

## 2.3 Definición de los requerimientos

Dentro de los elementos que se requieren para desarrollar este sistema de red inalámbrica, a continuación se hará una descripción de dos tipos:

- 1) Equipo a utilizar para verificar su funcionamiento en pruebas locales
- 2) Equipo recomendado para el momento en que el sistema de red inalámbrica se encuentre en operación.

En lo que respecta a la parte del hardware, de inicio, para comprobar el correcto funcionamiento del sistema de red inalámbrica, no será necesario contar con un equipo de gran capacidad o rendimiento, tampoco un Access Point que soporte gran cantidad de usuarios o que tenga un alcance de muchos metros, por lo cual, el equipo inicial con el que se iniciarán las pruebas para verificar el correcto funcionamiento, sería el siguiente:

- Para la parte del equipo de cómputo se hará uso de un equipo DELL modelo 14 7000, de 16 Gb de memoria RAM, con DD (Disco Duro) de 105 Gb y un procesador Intel Core i5. (Véase Tabla 3)
- Para la parte del Access Point se hará uso de un Access Point marca TP-Link, modelo TL-WR840N con velocidad de transmisión de 300 Mbit/s y dos antenas externas. (Veáse Tabla 3).

Más adelante, al momento de implementar este sistema en la Facultad, todo lo mencionado será de gran prioridad, para lo cual, se propone otro tipo de equipo y de Access Point tales como:

- Un equipo DELL modelo Vostro 3000, de 8 Gb de memoria RAM, DD (Disco Duro) de 1 TB y un procesador Intel i7 (Véase Tabla 3).
- Un Access Point marca Cisco, modelo WAP150, con velocidad de transmisión de 1200 Mbit/s y antena interna 3 dBi.

En lo que respecta a la parte del software, que incluyen el sistema operativo y al servidor, lo único que podría cambiar si así se quisiera sería el sistema operativo, sin embargo más adelante se analizarán en profundidad estos detalles, desde que son, como funcionan, hasta sus ventajas y desventajas.

Tabla 3. Descripción de equipos utilizados y recomendados

Equipo a utilizar para pruebas	Equipo recomendado en producción
<p data-bbox="221 300 715 405">Equipo DELL Inspiron 14 7000, 16 Gb RAM, DD 105 Gb, Procesador Intel Core i5-7500U, 2.70 GHz</p> 	<p data-bbox="778 300 1406 405">Equipo DELL Vostro 3000, 8 Gb RAM, DD 1 TB 7200 RPM, Procesador Intel i7-9700, up to 4.7GHz</p> 
<p data-bbox="212 696 727 801">Access Point TP Link TL-WR840N , velocidad de 300 Mbit/s, 2.4 GHz, 2 antenas externas de 2dBi,</p> 	<p data-bbox="770 696 1417 801">Access Point Cisco WAP150, 1200 Mbit/s, 2.4 GHz, Antena interna 3 dBi, 256 MB RAM, Memoria Flash 128 MB</p> 
<p data-bbox="209 1176 730 1211">Conexión a internet mínima de 5 Mb</p>	<p data-bbox="823 1176 1362 1211">Conexión a internet mínima de 20 Mb</p>
<p data-bbox="217 1249 722 1323">Sistema Operativo Debian 9 Buster X64</p>	<p data-bbox="807 1249 1378 1285">Sistema Operativo Debian 9 Buster X64</p>
<p data-bbox="325 1361 614 1393">Servidor freeRadius</p>	<p data-bbox="951 1361 1236 1393">Servidor freeRadius</p>

Fuente: Información obtenida de Internet<sup>4</sup>

<sup>4</sup> Información obtenida de:  
<https://www.cisco.com/c/en/us/products/wireless/wap150-wireless-ac-n-dual-radio-access-point-poe/index.html>  
<https://www.tp-link.com/mx/business-networking/outdoor-radio/cpe220>



# **CAPÍTULO 3**

# **FUNDAMENTOS**

# **TEÓRICOS**

### 3.1 Introducción

En este apartado se revisan fundamentos a utilizar para la creación de este proyecto, fundamentos que en su mayoría fueron vistos en materias tomadas durante la carrera, se abordan temas relacionados con sistemas operativos, bases de datos, metodologías, servidores y redes de datos, mostrando sus características y algunas ventajas y desventajas de las mismas.

### 3.2 Software

Se denomina software a todos los componentes intangibles de una computadora, es decir, todos aquellos programas que ejecutan una tarea en específico, como tal podría ser el caso de un sistema operativo o un procesador de textos.<sup>5</sup>

En las ciencias de la computación y la ingeniería de software, el software es toda la información procesada por los sistemas informáticos: programas y datos.

El software se clasifica de dos maneras:

1. De acuerdo con el tipo de trabajo realizado (Véase Figura 3).
2. De acuerdo con el método de distribución (Véase Figura 4).

#### Clasificación del software de acuerdo con el tipo de trabajo realizado

##### 1. Software de Sistema

El software de sistema son los programas básicos el cual controla a la computadora, también llamado sistema operativo el cual tiene tres grandes funciones: coordina y manipula el hardware de la computadora, como la memoria, las unidades de disco; organiza los archivos en diversos dispositivos de almacenamiento y gestiona los errores de hardware y del mismo software.

Los sistemas operativos pueden ser de tarea única o multitarea. Los sistemas operativos de tarea única, los más primitivos, solo pueden manejar un proceso en cada momento.

##### 2. Software de aplicación

Permite a los usuarios llevar a cabo una o varias tareas más específicas, en cualquier campo de actividad susceptible de ser automatizado o asistido, con especial énfasis en los negocios, nos ayudan a la elaboración de una determinada tarea, este tipo de software es diseñado para facilitar al usuario en la realización de un determinado tipo de trabajo.

---

<sup>5</sup> Información obtenida de la liga:

<https://docs.google.com/document/edit?id=15SWNVY7vRs26ABwceZSQURbJgrH5pyn4bMBQODDh6L8&hl=es>

El software de aplicación resulta una solución informática para la automatización de ciertas tareas complicadas como puede ser la contabilidad y gestión de una organización.

### 3. Software de desarrollo

Este software recibe varios nombres, como software de programación o lenguaje de programación del software. El software de programación proporciona herramientas para ayudar al programador a escribir programas informáticos y a usar diferentes lenguajes de programación de forma práctica, entre los lenguajes de programación más utilizados podemos mencionar: C++, Java, C#, Visual Basic, etcétera.



**Figura 3. Software de acuerdo con el tipo de trabajo realizado**  
Fuente: Imagen tomada de Internet<sup>6</sup>

### Clasificación del software de acuerdo con el método de distribución.

#### 4. Freeware

Es aquel software de computadora que se distribuye sin cargo. A veces se incluye el código fuente, pero no es lo usual. El Freeware suele incluir una licencia de uso, que permite su redistribución, pero con algunas restricciones, como no modificar la aplicación en sí, ni venderla, y dar cuenta de su autor. Programa computacional cuyo costo económico para

<sup>6</sup> Imagen obtenida de:  
<https://sites.google.com/site/softwareeriosyromano/software-de-aplicacion>  
<https://okhosting.com/blog/herramientas-de-desarrollo-de-software/>

el usuario final es cero, independiente de las condiciones de distribución y uso que tenga.

## 5. Software multimedia

Se refiere a los programas utilizados para presentar de una forma integrada textos, gráficos, sonidos y animaciones, este tipo de software es considerado como una nueva tecnología. Este tipo de software suele utilizarse para el desarrollo de proyectos específicos multimedios, utilizar software multimedia requiere de tiempo, capacidades, dedicación y recursos.

## 6. Software de uso general

El software de uso general, es aquel que sirve para resolver problemas muy variados del mismo tipo, de muy diferentes empresas o personas, con adaptaciones realizadas por un usuario, ejemplos: procesadores de texto, manejadores de bases de datos, hojas de cálculo, etcétera.

## 7. Software de uso específico

Software desarrollado específicamente para un problema específico de alguna organización o persona, utilizar este software requiere de un experto en informática para su creación o adaptación.



**Figura 4. Software de acuerdo con el método de distribución**

Fuente: Imagen tomada de Internet<sup>7</sup>

A continuación se hablará de un tema que suele ser muchas veces controversial, el software libre, y esto es porque muchas veces se piensa que por ser “libre” no es de calidad o que no cuenta con el soporte necesario por si algo sale mal, sin embargo con el paso del tiempo, se ha demostrado que no es así, y como ejemplo podemos ver que este tipo de software muchas veces cuenta con más y mejor soporte, ya que no solo las personas que lo crean son las encargadas de darle el

<sup>7</sup> Imagen obtenida de:

<https://sites.google.com/site/lobasicodeinformatica/clasificacion-del-software->  
<https://www.winrar.es/descargas>

mantenimiento o ayudan a corregir los errores, si no que detrás existe toda una comunidad de usuarios que siempre están a la expectativa para corregir errores.

También es usual que se piense que por ser libre, se puede hacer lo que uno desee con el código del programa, lo cual no es cierto, ya que como todo, también tiene sus excepciones. Y todo eso se aborda en los puntos que aparecen a continuación.

### **3.3 Software libre**

El software libre es un tipo de programa de computadora que una vez obtenido puede ser usado, copiado, estudiado, modificado y redistribuido libremente (Véase Figura 5). Estos permisos son otorgados por sus autores, o quién sea que ostente los derechos de propiedad intelectual sobre el producto, a través de un acuerdo de licencia, una forma de contrato, que acompaña a cualquier programa.

De forma más precisa, el software libre es aquel cuya licencia ofrece a los usuarios las siguientes cuatro libertades:

1. La libertad de usar el programa, con cualquier propósito. (libertad 0)
2. La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades. (libertad 1)
3. La libertad de distribuir copias, con lo que puedes ayudar a tu vecino. (libertad 2)
4. La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3)

Para que las libertades 1 y 3 sean posibles es necesario que los usuarios puedan tener acceso al código fuente de los programas. Este código fuente es el conjunto de líneas de texto que indican las instrucciones que debe seguir la computadora para ejecutar dicho programa. Es decir, en el código fuente de un programa está escrito por completo su funcionamiento. Así que tener acceso a él es la única forma de poder estudiarlo y modificarlo con garantías. Debemos tener presente que, aunque mucho software libre es gratuito, esto no tiene por qué ser necesariamente así. La libertad del software está relacionada con los permisos que su autor ofrece y no con su precio.



**Figura 5. Ejemplos de software libre**  
Fuente: Imagen tomada de Internet<sup>8</sup>

### Motivos para escoger software libre

Muchos destacan entre sus ventajas que suele tener menor costo económico, ya que es un hecho que muchos de estos programas son gratuitos. Sin embargo, sin duda una de sus características más destacables es su capacidad para fomentar la libre competencia. Por ejemplo, con software libre (Véase Figura 6) es factible que cualquiera de soporte a un producto, aunque no lo haya desarrollado ya que, al ser el software libre de código abierto, junto con el programa también se puede obtener el código fuente. Por otro lado, hay que tener en cuenta que, aunque no se obtenga directamente dinero por el software, las implantaciones de muchas herramientas profesionales suelen ir acompañada de otros costos, por ejemplo, la integración, mantenimiento o formación, distintos al pago por la licencia.



**Figura 6. Sistemas operativos de distribución libre.**  
Fuente: Imagen tomada de internet<sup>9</sup>

<sup>8</sup> Imagen obtenida de:  
<https://www.pinterest.com/pin/413838653253347772/>

<sup>9</sup> Información tomada de:  
<https://karolvanesaep.wixsite.com/tecnologia/single-post/2015/11/22/SOFTWARE-LIBRE>

Dentro del mundo del software libre existen las siguientes categorías:

- **Software de código abierto (Open Source)**

Algunas personas utilizan la expresión software de código abierto para referirse más o menos a la misma categoría a la que pertenece el software libre. De todos modos, hay muy poca diferencia entre lo que abarca una y otra categoría. Se prefiere la expresión 'software libre', debido a que es una cuestión de libertad, no de precio, cosas que no sucede con la expresión 'Open Source'.

- **Software de dominio público**

El software de dominio público es aquel que no tiene derechos de autor. Si el código fuente es de dominio público, se trata de un caso especial de software libre sin copyleft, lo que significa que algunas copias o versiones modificadas pueden no ser libres en absoluto. En algunos casos, un programa ejecutable puede ser de dominio público, pero no disponer libremente del código fuente.

- **Software con copyleft**

El software con copyleft es software libre cuyos términos de distribución garantizan que todas las copias de todas las versiones tengan aproximadamente los mismos términos de distribución. Esto significa, por ejemplo, que las licencias copyleft generalmente no permiten que terceros le agreguen requisitos adicionales al software (aunque puede estar permitido agregar un conjunto limitado de requisitos que se consideran seguros) y exigen que el código fuente esté disponible.

- **Software sin Copyleft**

Los programas publicados sin copyleft vienen con permiso de redistribución y modificación, como así también con el permiso de agregarle restricciones. Si un programa es libre pero no tiene copyleft, es posible que algunas copias o modificaciones no sean libres en absoluto.

- **Software con licencia permisiva, laxa**

Estas licencias permiten utilizar el código de cualquier manera, inclusive la distribución de binarios privativos con o sin modificaciones del código.

- **Software con licencia GPL**

La Licencia Pública General de GNU (General Public License - GNU GPL) consiste en un conjunto específico de cláusulas de distribución para publicar programas con copyleft.

- **Sistema operativo GNU**

El sistema GNU incluye todos los paquetes oficiales de GNU. También incluye muchos otros paquetes, como el sistema X Window y TeX, que no son software de GNU. Debido a que el objetivo de GNU es ser libre, cada uno de los componentes del sistema operativo GNU es software libre.

Existen muchas distribuciones en el mundo Linux, cada una enfocada en un nicho de mercado diferente, en el caso de servidores, la distribución más utilizada es Ubuntu con el 37% del mercado, seguido por Debian con el 23% y CentOS con el 18%<sup>10</sup> (Véase Figura 7).

A continuación se hablará de cada una de estas distribuciones mencionando características generales y qué tipo de usuario es capaz de manejar estos sistemas.

### 1) **Ubuntu Server**

Ubuntu es una distribución GNU/Linux que ofrece un sistema operativo predominantemente enfocado a equipos de escritorio, aunque también proporciona soporte para servidores. Ubuntu Server es una variante de Ubuntu que sale con cada versión y está dedicada especialmente para su uso en servidores. El uso de Ubuntu como servidor se ha extendido mucho en los últimos años, sobre todo para el uso de servidores web, de un modo tanto particular como profesional.

Ubuntu Server es un Sistema Operativo sin entorno gráfico (aunque podemos instalarlo) lo que quiere decir que todas las acciones se realizan mediante consola, y normalmente ni si quiera a través del propio servidor, sino desde una conexión remota, esto quiere decir que está dirigido para usuarios con experiencia previa en manejo de comandos. El manejo de Ubuntu Server es muy similar al de cualquier otro Sistema Linux, pero con las particularidades de Ubuntu (como el sudo).

Cuando estamos instalando Ubuntu Server nos hace una serie de preguntas sobre qué tipo de servicios queremos instalar, entre una lista de los más típicos, y nos pregunta los parámetros necesarios para su configuración. De este modo podemos instalar de una forma fácil y sencilla un servidor acorde a nuestras necesidades en unos pocos minutos.

Es muy habitual encontrarnos Ubuntu Server como sistema operativo en muchos de los VPS que podemos contratar en la mayoría de compañías, aunque también nos suelen dar a elegir otras distribuciones Linux.

---

<sup>10</sup> Información tomada de:  
<https://hostingdiario.com/centos-linux/>



## 2) Debian<sup>11</sup>

Debian es especialmente popular entre los usuarios avanzados debido a su excelencia técnica y a sus comités siempre atentos a las necesidades y expectativas de la comunidad Linux. Debian también introdujo muchas características a Linux, que ahora son comunes. Fue la primera distribución de Linux en incluir un sistema de gestión de paquetes para una fácil instalación y desinstalación del software. Además, también fue la primera que podía actualizarse sin necesidad de reinstalarla.

Lo que más distingue a Debian de otras distribuciones GNU/Linux es su sistema de gestión de paquetes. Estas herramientas otorgan al administrador de un sistema Debian total control sobre los paquetes instalados, incluyendo la capacidad de instalar un sólo paquete o actualizar el sistema operativo por completo.

Para proteger su sistema contra “caballos de Troya” y otros programas malévolos, los servidores de Debian verifican que los paquetes provienen de sus auténticos encargados.

Los desarrolladores de Debian también ponen gran cuidado en configurarlos de forma segura. La atención que pone Debian a los detalles, nos permite producir una distribución de alta calidad, estable y escalable.

## 3) CentOS<sup>12</sup>

Es una distribución hecha por y para la comunidad, tiene detrás una de las comunidades más grandes de Linux lo que garantiza su continuidad. Es un sistema muy estable por diseño, es un excelente sistema para montar cualquier plataforma y si suma una cuidadosa configuración de seguridad y endurecimiento (hardening) de rendimiento se obtendrá un sistema estable, confiable, seguro y de alto rendimiento, libre de errores y caídas.

Soporta una amplia gama de hardware y eso incluye el hardware habitual de los centros de datos que difiere bastante del hardware normal de cualquier pc, desde los procesadores hasta las tarjetas de red y placas base. Soporta los principales métodos de virtualización desde VMWare, XEN, KVM, contenedores como Docker u orquestadores como Kubernetes, Mesos y Swarm.

CentOS no suele considerarse como una de las primeras opciones dentro de los sucesores más potentes de Linux. Sin embargo, esta distribución

---

<sup>11</sup> Información tomada de:

<https://www.debian.org/releases/etch/hppa/ch01s03.html.es>

<sup>12</sup> Información obtenida de la liga:

Datos obtenidos de <https://hostingdiario.com/centos-linux/>

cuenta con todo lo necesario para suplir las necesidades de su sector objetivo: el corporativo y de los servidores.



**Figura 7. Sistemas operativos de servidores más utilizados**  
*Fuente: Imagen tomada de Interne*

A continuación, a modo de resumen, se muestra una tabla con las ventajas y desventajas, así como algunas otras características de instalación mínimas de cada uno de los sistemas operativos mencionados:

*Tabla 4. Ventajas, desventajas y requisitos mínimos de instalación.*

	<b>Ventajas</b>	<b>Desventajas</b>	<b>Memoria RAM</b>	<b>Disco Duro</b>	<b>Procesador</b>
<b>CentOS</b>	<ul style="list-style-type: none"> <li>* Basado en Red Hat, a decir, mismo SO pero sin costo.</li> <li>* Estable</li> <li>* Eficaz</li> <li>* Actualización de seguridad rápida.</li> </ul>	<ul style="list-style-type: none"> <li>* Para escalar, hay que añadir servidores.</li> <li>* Si falla el servidor, todo falla.</li> <li>* Límites en audio y video sincronizado.</li> </ul>	192 MB	850 MB (Mínimo) 2 GB (Recomendado)	Intel Pentium I/II/III/IV/Celeron, AMD K6/II/III, AMD Duron, AMD Athlon/XP/MP
<b>Debian</b>	<ul style="list-style-type: none"> <li>* Instalación sencilla.</li> <li>* Rápido y ligero en memoria.</li> <li>* Estabilidad.</li> <li>* Gran cantidad de software</li> <li>* Mantenido por usuarios.</li> <li>* Software de seguridad</li> </ul>	<ul style="list-style-type: none"> <li>* Gran tiempo entre nuevas versiones</li> <li>* Poca compatibilidad con paquetes de software populares.</li> <li>* Dificultad al realizar una configuración.</li> </ul>	Consola 128 MB (Mínimo) 512 MB (Recomendado)  Gráfico 256 MB (Mínimo) 1 GB (Recomendado)	Consola  2 GB  Gráfico  10 GB	Intel Pentium IV
<b>Ubuntu Server</b>	<ul style="list-style-type: none"> <li>* Costo menor comparado con Red Hat.</li> <li>* Amplia documentación.</li> <li>* Gran soporte técnico.</li> </ul>	<ul style="list-style-type: none"> <li>* Más inestable tras actualización en comparación con CentOS.</li> <li>* Curva de aprendizaje más lenta.</li> </ul>	Consola 256 MB (Mínimo)  Gráfico 512 MB (Mínimo)	Consola 2 GB  Gráfico 4 GB	AMD Optaron y Intel EM64T Xeon

Fuente: Información obtenida de Internet<sup>13</sup>

<sup>13</sup> Información obtenida de la liga:

<https://hostingdiario.com/centos-linux/>

<http://www.ubuntufacil.com/2013/04/ubuntu-server/>

[https://www.debian.org/intro/why\\_debian.es.html](https://www.debian.org/intro/why_debian.es.html)

### 3.4 Metodología de desarrollo de proyectos

En la actualidad existen muchas variantes de metodologías de proyectos, entendiendo el concepto de metodología de gestión de proyectos brindado por el PMI (Project Management Institute) el cual lo define como:

*‘Una metodología es un sistema de prácticas, técnicas, procedimientos y reglas utilizadas por quienes trabajan en una disciplina.’*

En términos generales, hay muchas maneras diferentes de entregar proyectos, estas formas son nuestras metodologías: aplicar diferentes principios, temas, marcos, procesos y estándares para ayudar a estructurar la forma en que entregamos los proyectos.

Algunas metodologías de gestión de proyectos simplemente definen principios, como Agile, y algunos son muy ligeros y simplemente definen el proceso, como Scrum. Como una visión general de los tipos de metodologías de administración de proyectos, existen los siguientes:

#### 1. Agile

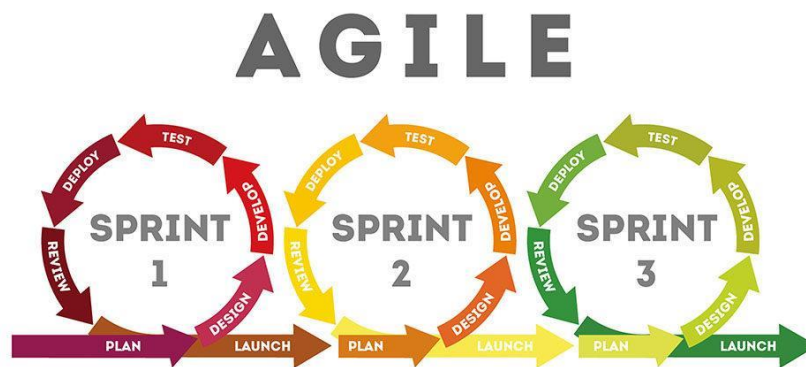
Agile es una metodología de desarrollo flexible de un proyecto, mediante el cual este se divide en pasos separados para crear así un esquema de trabajo práctico y funcional (véase Figura 8). Los requisitos para cada proyecto se deciden rápidamente, y la interacción entre especialistas se lleva a cabo en el marco de un equipo de proyecto auto organizado.

Entre las ventajas de Agile destacan las siguientes:

- a) El equipo puede dividir el proyecto en etapas y así centrarse en cada una de forma individual. Esto permite trabajar más rápido.
- b) Permiten adaptar el proyecto a medida que avanza.
- c) Es muy sencillo volver a organizar el equipo en relación con los nuevos objetivos.
- d) Identificación de las tareas más interesantes en cada momento.
- e) Facilidad de encontrar los errores de forma sencilla.
- f) El cliente puede ver resultados de manera mucho más sencilla en cualquier punto del proceso.
- g) Retroalimentación más rápida de los usuarios finales.
- h) Flexibilidad para definir funciones prioritarias y establecer objetivos.

Entre las desventajas de Agile destacan las siguientes:

- a) Al inicio del proyecto, es difícil determinar con precisión la cantidad de tiempo y dinero que se necesitará para completarlo, debido a los requisitos en constante cambio.
- b) El equipo necesitar tener una base sólida y habilidades.
- c) Se requiere un alto nivel de interacción entre el cliente y los desarrolladores.
- d) La falta de atención a la documentación puede dificultar que los nuevos miembros del equipo accedan a la misma.



**Figura 8. Ciclo de Agile**  
Fuente: Imagen tomada de internet<sup>14</sup>

## 2. Scrum

La metodología Scrum establece un marco (Framework) que se aplica con el propósito de mejorar continuamente tanto el método de trabajo como el producto. El marco está formado por roles, eventos, artefactos y reglas. En Scrum, los clientes (o contratantes) y los usuarios adoptan una posición importante y el desarrollo se basa en sus necesidades.

Existen dos términos en Scrum cuyo papel resulta clave:

- a) Iterativo: El trabajo describe un ciclo que va desde la planificación hasta la evaluación, pasando por el desarrollo y las pruebas, y vuelve desde ahí a la fase de planificación. Por consiguiente, Scrum no solo se ocupa de la creación inicial, sino también del desarrollo posterior.

---

<sup>14</sup> Información obtenida de la liga:  
<https://ebf.com.es/blog/ventajas-y-desventajas-de-las-metodologias-agiles-y-su-aplicacion-en-el-trabajo/>

- b) Incremental: Se basa en la idea de que un producto se crea por pasos y estos pasos son concebidos como objetivos intermedios. Para obtener una mayor flexibilidad, los proyectos más grandes se dividen en diversos proyectos pequeños.

Si se combinan ambos términos, el resultado es un procedimiento iterativo-incremental, lo que quiere decir que el proyecto se desarrolla de forma gradual y continua. Por un lado, se obtiene un proceso más transparente y por el otro, se mejora la calidad de los productos, dado que su funcionalidad se comprueba continuamente.

Dentro de un equipo Scrum hay tres roles definidos:

- a) El **Team** (equipo)  
Hace referencia a los desarrolladores del producto y el grupo está definido de tal manera que puede organizarse por sí mismo y establecer cómo se puede alcanzar un objetivo acordado. El tamaño del equipo debe erigirse de tal manera que pueda actuar con rapidez y flexibilidad sin dejar de ser eficiente, se recomiendan que tenga entre 3 y 9 miembros.
- b) El **Product Owner** (dueño de producto)  
Es el responsable de la calidad del producto y del trabajo, por lo que dicha persona adopta la función de control, organizan el product backlog (lista del producto), una lista que determina cuáles son los requisitos del proyecto. Para garantizar la mayor productividad posible, el equipo debe confiar en las decisiones del dueño del producto.
- c) El **Scrum master** (líder de Scrum).  
Es el responsable de integrar e introducir Scrum en el proceso de trabajo, organiza las reuniones y ejerce de moderador. Se ocupa de que todos los participantes comprendan los procesos correctamente, ofreciendo consejos para crear las listas con el trabajo pendiente (backlogs), organizando el equipo y, en general, eliminando los obstáculos que puedan ir surgiendo, para ello también puede obtenerse la certificación de Scrum master

Otro aspecto importante a tomarse en cuenta son los ciclos de trabajo (**sprints**), con las siguientes características:

- a) Un sprint comienza con el sprint planning (planificación del ciclo de trabajo).
- b) Participan todos los roles del equipo Scrum

- c) El sprint es corto para que, a la hora de formular los objetivos, se mantenga la simplicidad.
- d) Durante una reunión de hasta un máximo de ocho horas, las partes implicadas hablan del próximo incremento de producto: ¿qué debe incluir el incremento y cómo quiere lograrse?
- e) El punto de partida para la planificación es el product backlog (pila de producto).
- f) Esta corta duración determina la realización de un análisis del desarrollo al menos una vez al mes.
- g) En el sprint tiene lugar un daily Scrum (Scrum diario) todos los días, siempre a la misma hora y en el mismo lugar, lo que ahorra esfuerzos organizativos.
- h) En cada reunión, de 15 minutos, el equipo de desarrollo conversa sobre las tareas pendientes del día y lo que se ha logrado el día anterior.
- i) La comparación diaria garantiza que las personas implicadas no pierdan de vista los objetivos, lo que también aumenta la productividad.

Mientras que en la revisión el elemento central es el producto y el avance del proyecto, en la retrospectiva prevalece el trabajo en equipo. El objetivo de esta segunda revisión es mejorar el trabajo entre las partes y sortear los problemas internos. Cuando acaba un sprint, el siguiente empieza con el sprint planning (planeación del ciclo de trabajo).

### **Artefactos Scrum**

Los objetos que desempeñan un papel importante en la metodología Scrum reciben el nombre de artefactos y en ellos se engloban:

- a) **Product backlog** (pila de producto)  
En el product backlog el responsable es el product owner: un backlog es una lista clasificada de todos los elementos decisivos para el producto que recoge tanto las funciones del producto como las correcciones y las mejoras.
- b) **Sprint backlog** (lista de pendientes del sprint)  
Se genera a partir del anterior. Este último recoge todas las entradas del product backlog seleccionadas en el sprint planning para el próximo sprint. Para ello, el backlog presenta todos los pasos hasta el objetivo del sprint correspondiente.
- c) **El incremento terminado**  
Se define como el resultado de la fase de desarrollo. En el incremento deben estar incluidas todas las entradas del sprint backlog y todos los incrementos de sprint anteriores. Además, todo incremento debe

poderse entregar directamente y debe estar listo para usarse, aun cuando no se haya planeado la entrega del incremento de producto

En la revisión de los ciclos de trabajo (sprint review), el equipo presenta el incremento y así el dueño del producto (product owner) y las partes interesadas (Stakeholders) pueden valorar el resultado.

La metodología Scrum ofrece algunas ventajas

**a) Comunicación**

Scrum hace mucho hincapié en que los trabajadores tengan una estrecha comunicación entre sí. La reunión diaria, a primera hora de la mañana, hace que ningún desarrollador trabaje más que los otros y en la última fase de revisión también se tratan los problemas internos del equipo.

**b) Flexibilidad**

Dado que entre dos reuniones de planificación transcurre máximo un mes, el proyecto puede modificarse a corto plazo y adaptarse a las nuevas exigencias.

**c) Transparencia**

La comunicación continua de todos los miembros del equipo, así como la creación de artefactos Scrum, favorecen una transparencia elevada.

**d) Control de calidad**

La corrección de errores y los desarrollos pertenecen también a la lista de todo el trabajo pendiente (product backlog). Asimismo, el dueño del producto y las partes interesadas tienen la oportunidad de convencerse de la calidad, con lo que puede reaccionarse con mayor celeridad a los errores en el desarrollo que detectando una función defectuosa al final de un proyecto.

**e) Creatividad**

Debido a que el marco de Scrum es, en comparación, abierto y no tiene muchas normas, cada uno de los componentes del equipo puede aportar sus propias ideas.

**f) Efectividad**

En comparación con la gestión de proyectos clásica, en el método Scrum existe una obligación de documentación mucho menor, ya que el foco de la atención está en el producto en funcionamiento y no en la documentación en sí, que requiere tiempo y recursos



Entre las desventajas de Scrum se encuentran las siguientes:

**a) Falta de perspectiva general**

Una planificación a muy corto plazo puede dar lugar a que se pierda de vista la perspectiva general en los proyectos de mayor envergadura.

**b) Comunicación laboriosa**

Los daily Scrum (Scrum diarios) habituales impiden en tales casos la productividad, pues se dedica mucho tiempo a la organización y la comunicación en lugar de trabajar en el producto.

**c) Dudas respecto a la planificación y la competencia**

Scrum prevé la propia organización del equipo. Esto puede ser ventajoso, pero en algunos campos y sectores esta jerarquía plana puede causar incertidumbre en la planificación y confusión con respecto a la competencia.

**d) No integrable**

En algunas estructuras empresariales solo es posible integrar Scrum realizando grandes cambios, casos en los que se plantea la pregunta de si no se pierde más efectividad de la que se gana con Scrum.

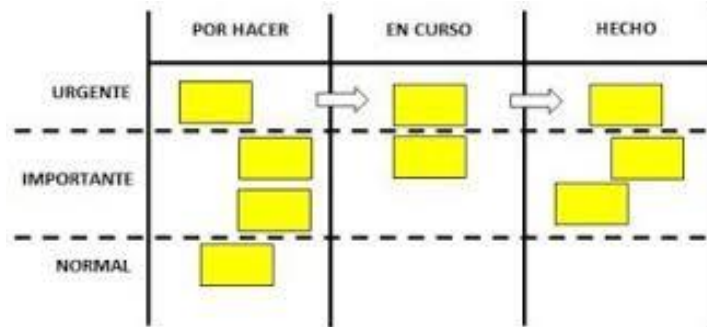
### 3. Kanban

Kanban es originario de Japón, desarrollado por Toyota en el año 1947 y cuya traducción aproximada es tarjeta visual. El objetivo de la implantación de Kanban consiste en mejorar el flujo de trabajo de un equipo, aumentando al mismo tiempo la productividad y la calidad del producto final. Se engloba dentro de la denominada metodología ágil y, como tal, otorga una gran flexibilidad a los procesos de trabajo.

Las tareas se dividen en pequeñas fases que se realizan de forma consecutiva y con base en el siguiente lema: stop starting – start finishing. A diferencia de otros, Kanban puede integrarse en los procesos de trabajo sin hacer grandes modificaciones, por lo que la adaptación al sistema resulta muy sencilla y deja abierta otras posibilidades, como la de aplicar, junto con Kanban, otras metodologías de gestión de tareas (como Scrum). El elemento central del método es el tablero Kanban (ya sea físico o digital), y con ello, la visualización del flujo de trabajo. Las distintas tareas se especifican en tarjetas de colores (por ejemplo en notas o fichas) y se colocan en el tablero de forma ordenada y clara. El tablero se divide en varias columnas (tres, como mínimo). A la izquierda, se encuentra la lista de tareas pendientes (el backlog), en la que, tal y como su nombre indica, se especifican todas las tareas que se deben llevar a cabo.

A continuación, en la siguiente columna, se dispondrán todas las tareas en curso, por lo que esta columna recibe el nombre de Work in Progress

(WiP) o, simplemente, En curso. Si la tarea cuenta con varias fases, esta parte puede dividirse a su vez en varias columnas. También puede crearse una columna para revisiones y pruebas. Las tarjetas se irán moviendo de izquierda a derecha hasta llegar a la columna final de tareas concluidas (Véase Figura 9).



**Figura 9. Ejemplo de un tablero Kanban**

Fuente: Imagen tomada de Internet<sup>15</sup>

Gracias a esta exposición del trabajo, se consigue fácilmente una mayor transparencia. Pero mediante Kanban no solo se consigue una mejor visualización del flujo de trabajo, sino que además se está eligiendo una metodología que limita el número de tareas en curso. Antes de implantar Kanban en la producción, se determina la cantidad de tareas que deben realizarse simultáneamente. Así, un equipo solo podrá llevar a cabo el trabajo de dos tarjetas al mismo tiempo, ya que la realización simultánea de tareas provoca una demora en la ejecución.

Además, la limitación de tareas permite dividir de forma más eficiente la capacidad del equipo. Cuando una tarea se compone de distintas fases, puede sobrevenir un embotellamiento con facilidad, de manera que, si la primera estación trabaja con rapidez y en la segunda estación surge un problema, los miembros de la primera estación no deberán continuar trabajando, según establece Kanban. Sin embargo, deberán emplear su capacidad disponible para ayudar a los miembros de la segunda estación a solucionar el problema.

Por su parte, el intercambio de información entre compañeros de trabajo es una parte esencial del método Kanban, puesto que se trata de mejorar continuamente el flujo de trabajo y el producto. Partiendo de la situación real, el equipo deberá introducir mejoras poco a poco en lugar de llevar a cabo un gran cambio de organización.

---

<sup>15</sup> Imagen tomada de:  
<https://www.pinterest.es/pin/425027283583294861/>

En total, se distinguen seis prácticas diferentes en el método Kanban:

1. Visualización  
El tablero Kanban permite tener una visión general de los flujos de trabajo. Lo único importante es que las estaciones estén claramente definidas y cada columna tenga determinado su límite.
2. Limitación  
Cada columna puede contar con un número máximo de tareas. Esto hace que el flujo de trabajo sea más eficiente.
3. Gestión  
Durante el proceso de trabajo, pueden surgir bloqueos y saturaciones. En ese caso, el equipo centrará su atención en solventar estos problemas.
4. Regulación  
Las reglas sirven para diseñar los flujos de trabajo de forma más clara y transparente. Entre estas reglas, se cuentan la determinación de límites, pero también la definición de cuándo se considera terminada una tarea.
5. Feedback (Retroalimentación)  
El intercambio de información es parte ineludible de los flujos de trabajo porque solo así se podrá mejorar. Para ello, se planean reuniones con regularidad, las denominadas cadencias. Al contrario que Scrum, Kanban no establece normas fijas para celebrar esas reuniones.
6. Kaizen (Mejora)  
El trabajo en equipo con Kanban debe mejorarse continuamente. Por tanto, la teoría parte de la base de que nunca se llegará a un estado óptimo; por el contrario, siempre se trabajará para alcanzar nuevas mejoras.

Entre las ventajas de Kanban se encuentran:

- a) Versatilidad.
- b) Más transparencia.
- c) Mejora continuada.
- d) Puede aplicarse en muchas situaciones.
- e) Sencilla implementación.

Entre las desventajas de Kanban se encuentran:

- a) La falta de planificación del tiempo puede causar problemas con las fechas de entrega.
- b) El trabajo debe poder dividirse en distintas fases.

#### **4. XP (Extreme Programming o Programación Extrema)**

Se considera como la aplicación más radical del desarrollo ágil de software. En este contexto, la Programación Extrema se diferencia concretamente del desarrollo en cascada en que implica múltiples problemas. Está orientado a las necesidades del cliente. Además, trata de fomentar la creatividad de los desarrolladores y acepta los errores como una parte natural del trabajo.

La programación Extrema rompe con la tradición de completar un proyecto durante meses de principio a fin para que al final el resultado no sea el adecuado. En lugar de esto, se hacen comprobaciones, se habla y se publica constantemente en ciclos cortos (Véase Figura 10). De esta forma se pueden determinar y eliminar los fallos rápidamente. Se basa en distintos valores, principios y técnicas, además, se asignan roles definidos para que las tareas se puedan asignar de forma clara.

A continuación se explicará cuáles son y en qué consisten cada uno de estos valores, principios y técnicas, así como el beneficio que aportan a la metodología.

##### **Valores**

La programación extrema trata de modificar el enfoque general del trabajo de programación con cinco valores. La idea es que el equipo como conjunto siga una mentalidad determinada para poder colaborar de la mejor manera y entregar un producto de primer nivel.

##### **1) Comunicación**

No solo importa la comunicación entre los miembros del equipo, sino también la comunicación entre los desarrolladores y los clientes. Solo si todos los implicados están en contacto de forma permanente se pueden evitar y detectar rápidamente los malos entendidos. En este contexto, se prioriza el diálogo cara a cara a la comunicación escrita.

##### **2) Sencillez**

La sencillez implica varias ventajas: si solo te concentras en los factores necesarios, no te distraes con cuestiones secundarias. Un código de programación simple también facilita la comprensión, otra ventaja a favor del valor de la comunicación. Si todo el equipo es capaz de entender el texto fuente, es más fácil intercambiar impresiones.

### **3) Feedback (retroalimentación)**

Es importante que el cliente tenga numerosas oportunidades para expresar sus críticas. El equipo trabaja en ciclos cortos, comprueba el código cada poco e incluso le presenta el avance al cliente en intervalos cortos. Así, las modificaciones y correcciones de errores se pueden llevar a cabo rápidamente.

### **4) Valentía**

Entiende la valentía como la disposición a decir la verdad, incluso cuando es poco agradable. Si hay errores en el producto, hay que señalarlos, aunque sean responsabilidad de uno mismo.

### **5) Respeto**

El respeto también implica que un desarrollador no realice modificaciones que tengan un impacto negativo en el trabajo de un compañero. Incluso fuera del equipo, la importancia de los valores es primordial hasta llegar al cliente.

## **Principios**

Los principios se ubican entre los valores y las prácticas, es decir, son el puente entre lo abstracto y lo concreto. Los principios se pueden derivar en mayor o menor medida de los valores citados.

### **1) Feedback inmediato (retroalimentación inmediata)**

En este contexto, la idea es que el feedback del propio sistema (al comprobar el código) se aplique en cuestión de segundos, en lugar de recopilar primero todas las impresiones. El feedback de los clientes, en cambio, debe solicitarse y aplicarse en intervalos de días o semanas.

### **2) Tendencia a lo sencillo**

El principio de la sencillez se corresponde con la base del valor con el mismo nombre, pero incluye instrucciones concretas de aplicación. En este sentido se usan dos métodos:

- a) You ain't gonna need it (YAGNI): mientras una función no se solicite expresamente, no se debe implementar para no realizar trabajo en vano.
- b) Don't repeat yourself (DRY): debe evitarse repetir tareas y diseñar el código de manera que los cambios no tengan que aplicarse en varios puntos, sino una sola vez, en la medida de lo posible.

### **3) Modificaciones incrementales**

Las modificaciones siempre se realizan en pasos pequeños. Así, el equipo reacciona más rápido y es más fácil entender la causa de las

modificaciones. Los cambios en el diseño o en la estructura del equipo también se deben llevar a cabo en pasos pequeños e incrementales.

#### **4) Aceptación de modificaciones**

La programación Extrema gira alrededor del cliente, sus peticiones gozan de la máxima importancia. Por ello, todo el equipo debe aceptarlas como algo positivo y no oponerse a su realización.

#### **5) Trabajo de alta calidad**

El equipo debe realizar un trabajo de alta calidad. No obstante, para poder realizar un trabajo de calidad, hace falta contar con la dirección. Si los factores son los adecuados y el equipo puede estar contento con el trabajo realizado, dicho equipo va a gozar de un excelente estado de ánimo.

### **Técnicas**

Las prácticas de la programación Extrema son instrucciones de actuación y métodos de trabajo muy concretos. Mientras que los valores y principios presentados también se usan en otros métodos de trabajo ágiles, las técnicas concretas en la programación Extrema son elementos claramente distintivos.

#### **1) Feedback fino (retroalimentación fina)**

Los programadores trabajan con ciclos extremadamente cortos. De esta forma se puede comprobar el código escrito una y otra vez. El denominado test-driven development hace que los desarrolladores escriban primero un entorno de prueba antes de comenzar a crear el código fuente en sí. En el momento en el que un código no supere esta prueba, no se puede seguir desarrollando.

#### **2) Proceso continuo**

La idea es que la revisión (refactoring) mejore el texto fuente y elimine repeticiones y componentes innecesarios. Un código optimizado de esta forma es más fácil de entender. El cliente tiene la oportunidad continua de probar directamente los últimos avances del desarrollo e implementar críticas y sugerencias.

#### **3) Comprensión conjunta**

Si el diseño es simple todo el mundo puede entender el código. Es importante que los desarrolladores puedan trabajar en los códigos de sus compañeros y que sepan qué cambios ha realizado cada uno. Así, los trabajadores nuevos que se suman al proyecto entenderán todo mucho más rápido.

#### **4) Bienestar de los desarrolladores**

El bienestar del equipo es importante para el éxito del proyecto. Para garantizar este bienestar, la programación Extrema impone una semana de 40 horas. Las horas extra deben evitarse a toda costa y solo se permiten si la semana siguiente no se suman más horas.

#### **Roles**

Aquí los roles sirven para distribuir todas las competencias entre los implicados, tanto desarrolladores como clientes.

##### **1) Cliente**

La programación Extrema gira en torno al cliente, hasta el punto que el cliente se considera como parte del equipo y siempre hay al menos un representante presente (on-site customer). El cliente marca las exigencias del producto, pero esta responsabilidad también implica expresar sus propios deseos de manera comprensible.

##### **2) Desarrollador**

Aquí no solo se incluyen programadores, sino también otras personas implicadas en la creación, en función de los requisitos del proyecto.

Entre los derechos de los desarrolladores está la posibilidad de buscar recursos necesarios, es decir, solicitar a la dirección medios adicionales

##### **3) Director**

El director es el elemento de comunicación entre los desarrolladores y los clientes. En este contexto, el director debe prestar atención a que se cumplan las normas acordadas previamente y las convenciones generales de un debate constructivo. El director también debe actuar como mediador si es necesario.

##### **4) Coach**

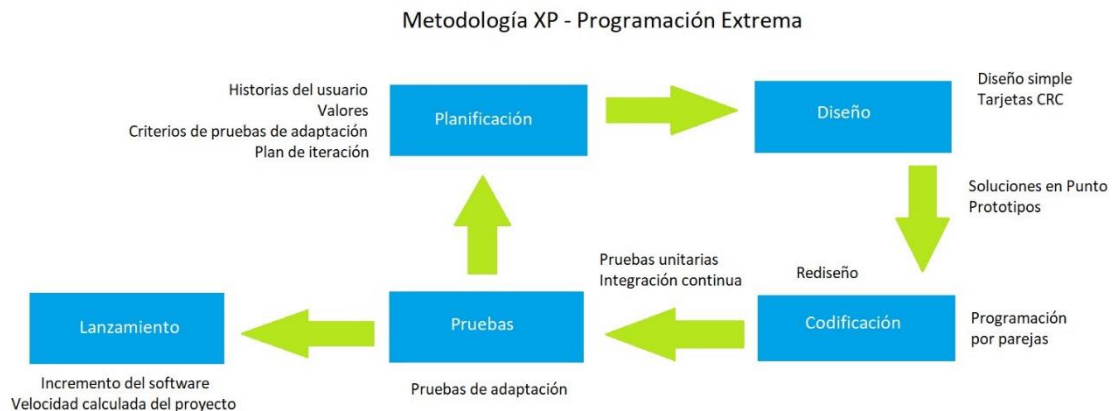
El coach no participa en el desarrollo del producto en sí, sino que solo brinda su apoyo como ayudante externo, de forma similar a un Scrum master. En el mejor de los casos, el coach acompaña al equipo a lo largo de todas las fases del desarrollo y ayuda a resolver dudas.

Entre las principales ventajas de la programación Extrema se encuentran:

- a) Relación estrecha con el cliente.
- b) Ausencia de trabajos de programación innecesarios.
- c) Software estable debido a continuas pruebas.
- d) Menos errores gracias a la programación en pareja.
- e) Ausencia de horas extra, gestión propia del tiempo.
- f) Aplicación rápida de cambios.

Entre las principales desventajas de la programación Extrema destacan:

- a) Mayor esfuerzo de trabajo.
- b) Requiere mucho tiempo.
- c) Relativamente caro.
- d) Requiere control de versiones.
- e) Requiere autodisciplina en la aplicación.



**Figura 10. Ciclo de vida de la metodología XP**

Fuente: Imagen tomada de Internet<sup>16</sup>

## 5. Modelo en cascada (Waterfall model)

El desarrollo en cascada es un procedimiento lineal que se caracteriza por dividir los procesos de desarrollo en sucesivas fases de proyecto. Al contrario que en los modelos iterativos, cada una de estas fases se ejecuta tan solo una vez. Los resultados de cada una de las fases sirven como hipótesis de partida para la siguiente. El modelo en cascada se utiliza, especialmente, en el desarrollo de software.

El desarrollo del modelo se atribuye al teórico de la informática Winston W. Royce. El cual propone un modelo compuesto por siete fases (Véase Figura 11) que se ha de ejecutar en diversas iteraciones:

1. Requisitos de sistema
2. Requisitos de software
3. Análisis
4. Diseño
5. Implementación
6. Prueba
7. Servicio

<sup>16</sup> Imagen tomada de:

<https://www.diegocalvo.es/metodologia-xp-programacion-extrema-metodologia-agil/>



El procedimiento popularmente conocido como modelo en cascada se basa en las fases definidas por Royce, pero solo prevé una iteración. En ocasiones, las fases 1, 2 y 3 definidas por Royce se integran en una sola fase de proyecto a modo de análisis de los requisitos.

1. Análisis: planificación, análisis y especificación de los requisitos.
2. Diseño: diseño y especificación del sistema.
3. Implementación: programación y pruebas unitarias.
4. Verificación: integración de sistemas, pruebas de sistema y de integración.
5. Mantenimiento: entrega, mantenimiento y mejora.

La siguiente imagen explica por qué el procedimiento lineal se denomina modelo en cascada.



**Figura 11. Ciclo del modelo en cascada**  
Fuente: Imagen tomada de Internet<sup>17</sup>

A continuación, se describen las fases del desarrollo en cascada.

### 1. Análisis

Todo proyecto de software comienza con una fase de análisis que incluye un estudio de viabilidad y una definición de los requisitos. En el estudio de viabilidad se evalúan los costos, la rentabilidad y la factibilidad del proyecto de software. Se establece que los problemas complejos se dividen en pequeñas tareas secundarias y se elaboran las correspondientes estrategias de resolución.

<sup>17</sup> Imagen tomada de:  
<https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/>

## **2. Diseño**

La fase de diseño sirve para formular una solución específica con base en las exigencias, tareas y estrategias definidas en la fase anterior, dando como resultado un borrador preliminar con el plan de diseño del software, así como planes de prueba para los diferentes componentes.

## **3. Implementación**

La arquitectura de software concebida en la fase de diseño se ejecuta en la fase de implementación. Los diversos componentes se desarrollan por separado, se comprueban a través de las pruebas unitarias y se integran poco a poco en el producto final. Dando como resultado un producto de software que se comprueba por primera vez como producto final en la siguiente fase (prueba alfa).

## **4. Prueba (Verificación)**

La fase de prueba incluye la integración del software en el entorno seleccionado. Los productos de software se envían en primer lugar a los usuarios finales seleccionados en versión beta (pruebas beta). Aquellos productos de software que superan con éxito las pruebas beta están listos para su lanzamiento.

## **5. Servicio (Mantenimiento)**

Una vez que la fase de prueba ha concluido con éxito, se autoriza la aplicación productiva del software. La última fase del modelo en cascada incluye la entrega, el mantenimiento y la mejora del software.

Entre las ventajas que ofrece el modelo en cascada destacan las siguientes:

- a) Una estructura sencilla gracias a unas fases de proyecto claramente diferenciadas.
- b) Buena documentación del proceso de desarrollo a través de unos hitos bien definidos.
- c) Los costos y la carga de trabajo se pueden estimar al comenzar el proyecto.
- d) Aquellos proyectos que se estructuran con base en el modelo en cascada se pueden representar cronológicamente de forma sencilla.

Entre las desventajas del modelo en cascada destacan las siguientes:

- a) Los proyectos más complejos o de varios niveles no permiten su división en fases de proyecto claramente diferenciadas.

- b) Poco margen para realizar ajustes a lo largo del proyecto debido a un cambio en las exigencias.
- c) El usuario final no se integra en el proceso de producción hasta que no termina la programación.

La metodología en cascada se suele emplear, especialmente, en aquellos proyectos cuyos requisitos y procesos se pueden describir de forma precisa durante la fase de planificación, en los que cabe suponer que las hipótesis no van a sufrir una gran variación durante el transcurso del proyecto.

### **Metodología elegida para este proyecto.**

Para la implementación de la IES se optó por seguir la metodología tipo Agile (Véase Figura 12), por los siguientes motivos:

- a) No creamos un plan detallado al 100% por adelantado.
- b) No hay necesidad de entregar el proyecto al 100% para demostrar su funcionalidad, ya que esto con cada fase que se vaya terminando podemos ir haciendo pruebas.
- c) Los interesados o involucrados en el proyecto, pueden realizar observaciones de cómo va el proyecto hasta ese momento, pudiendo mejorarlo para la siguiente etapa.
- d) No es necesario terminar una etapa para poder avanzar a la segunda, pueden ser hechos simultáneamente, para después observarse en conjunto.
- e) Los cambios pueden ser muy frecuentes, pudiendo ser en este caso en particular, que el número de usuarios en la IES aumenten o sean menores, que el alcance del Access Point sea mayor o menor, o incluso que tenga más o menos nivel de seguridad.
- f) Usualmente conlleva a una mejor satisfacción del cliente, ya que en todo momento estuvo involucrado en el proyecto, realizando observaciones para el mejoramiento del mismo.

A grandes rasgos, necesitamos sincronizar solamente actividades relacionadas como lo son el código, diseño, presupuesto, pruebas de testing, documentación, mantenimiento.



**Figura 12. Modelo Agile utilizado en el proyecto.**  
Fuente: Imagen tomada de Internet<sup>18</sup>

---

<sup>18</sup> Imagen tomada de:  
<https://www.progressalean.com/metodologia-agile/>

### 3.5 Bases de datos

En su definición más sencilla y clara, una base de datos se entiende como una recopilación de información, relacionada y almacenada en forma organizada. También se puede entender a una base de datos como una versión electrónica de alguna guía telefónica, un recetario, registro de calificaciones de un estudiante, direcciones de contacto, en fin, cualquier conjunto de información puede convertirse en una base de datos.

Las bases de datos ofrecen ventajas frente a los almacenamientos físicos, entre los cuales podemos destacar:

- Facilitan el almacenamiento de grandes cantidades de información
- Facilitan la recuperación y la búsqueda rápida de información
- Facilitan la reorganización de la información

Como principales desventajas, encontramos las siguientes:

- Instalación costosa.
- Requiere personal cualificado.
- Implantación larga y difícil.

Los conceptos básicos importantes a destacar de una base de datos son los siguientes:

*Tabla 5. Conceptos básicos de base de datos.*

Concepto	Definición
Tabla	Es aquella que se encarga de alojar datos de la base de datos.
Tupla	Se consideran tuplas a cada una de las filas de una tabla.
Atributo	Son las columnas de una tabla que contiene las propiedades o características que están asociadas a una entidad o relación.
Grado	Número de atributos por los que está compuesta una tabla.
Cardinalidad	Corresponde al número de tuplas de una tabla.
Dominio	Permite especificar los posibles valores válidos para un atributo.
Relaciones	Son aquellas que permiten evitar los datos redundantes.
Consultas	Permiten ver, modificar y analizar datos.
Clave Primaria	Clave designada que permite identificar únicamente las filas de la relación, es decir esta será la clave principal.
Clave Foránea	Es aquel atributo de una tabla que se encuentra relacionado con una clave de otra tabla.
Entidad	Se define a una entidad como un objeto del cual se desea almacenar información en la base de datos.

**Fuente: Ordoñez Mariuxi, Tapia Joofre, Asanza Wilmer. (2015). Fundamentos de bases de datos. Ecuador: UTMACH.**

## **Bases de datos existentes**

Existen diversos tipos de base de datos, pueden clasificarse de diversas maneras.

### **1. Según su flexibilidad de modificación**

#### **a) Bases de datos dinámicas**

Son aquellas donde los datos pueden actualizarse o incluso modificarse. La mayoría puede ser actualizada en tiempo real.

#### **b) Bases de datos estáticas**

Son bases de datos de consulta cuyos datos no pueden modificarse.

### **2. Según su forma de organización**

Las bases de datos jerárquicas, son aquellas organizadas en forma de un árbol al revés. Almacenan la información en forma de registros dentro de una estructura jerárquica, es de aquí que proviene su nombre. Cada registro de este "árbol" es llamado nodo. Nodos son registros que contienen alguna información de interés y a partir del nodo raíz son enlazados los otros nodos descendientes: padres e hijos. Cada nodo padre puede tener varios nodos hijos, pero cada nodo hijo solo puede tener un solo nodo padre. Este tipo de base de datos se recomienda para administrar grandes volúmenes de informaciones, pero actualmente no se utiliza mucho.

Las principales características de la base de datos jerárquica son:

- Diferentes usuarios pueden accederla y compartir información.
- Los datos son independientes.
- Es una estructura rígida por lo que es difícil modificar.
- Requiere gran conocimiento de las unidades de información.
- Los nodos distantes del nodo raíz son de difícil acceso por lo que se requiere tiempo.

## **Bases de datos de red**

Esta base de datos es una variación de la anterior. La diferencia está en que en la base de datos jerárquica un nodo hijo no puede tener varios padres y aquí sí. Las características de estas bases de datos son semejantes a las de las bases de datos jerárquicas, aunque estas son mucho más potentes y complejas.

## Bases de datos relacionales

Actualmente de las más usadas para administrar datos de forma dinámica (Véase Figura 13). Permite crear todo tipo de datos y relacionarlos entre sí. Los datos son almacenados en registros que son organizados en tablas, de esta forma pueden asociarse los elementos entre sí muy fácilmente, además se pueden cruzar sin ninguna dificultad. Sus principales características son:

- Pueden ser utilizadas por cualquier persona.
- Son de fácil gestión.
- Se pueden acceder a los datos con rapidez.
- Garantiza la total consistencia de los datos, sin posibilidad de error.

Una base de datos relacionales está formada por una o más tablas. Una tabla es una colección de datos sobre un tema en común. El diseñador de la base de datos es quien determina si los datos correspondientes a cada categoría se almacenan o no en tablas separadas. Las tablas organizan los datos en filas y columnas, las filas se conocen como registros y las columnas como campos. Cada tabla de una base de datos contiene una colección de registros con datos específicos sobre una persona, un producto o un evento. Veamos el siguiente ejemplo:

*Tabla 6. Datos de Médicos que se relacionarán con la tabla Ingresos*

Código de identificación del médico	Nombre	Apellidos	Especialidad	Cargo
AMG1	Alejandro	Martí Giménez	Medicina General	Adjunto
FPO1	Francisco	Palacios Ortega	Intensivos	Jefe de Sección
MRSN	María Rosa	Sánchez Navarro	Análisis Clínicos	Adjunto
MSM1	Manuel	Samaniego Muñoz	Pediatría	Jefe de Planta

Fuente: Información tomada de Internet<sup>19</sup>

Como se mencionó, cada columna se denomina campo, y en estos campos se puede almacenar diversos tipos de datos; por ejemplo, la tabla Médicos contendría los siguientes campos: Código de identificación del médico, Nombre, Apellidos, Especialidad y Cargo. Los datos de cada tabla deben estar relacionados con los de otras tablas por medio de un campo común al cual se le conoce como clave primaria.

Para este ejemplo, la tabla Médicos está relacionada con la tabla Ingresos por medio del campo 'Código de identificación del médico', el cual es un campo común entre las dos.

---

<sup>19</sup> Información tomada de:

Sáenz Pérez. (2011). Práctica 2. Ejemplo de diseño de bases de datos relacionales.

Tabla 7. Tabla Hospital relacionada con Tabla Médicos mediante su código.

Número de historial clínico	Fecha de ingreso	Código de identificación del médico	Número de planta	Número de cama	Diagnóstico
3	01/02/2004	MSM1	1	603	Neumonía
2	12/04/2004	MRSN	3	312	Insuficiencia renal
1	11/11/2004	FPO1	1	105	Infarto
5	03/03/2004	AMG1	3	310	Infarto

Fuente: Información tomada de Internet<sup>20</sup>

### Bases de datos no relacionales (NoSQL)

Están diseñadas para modelos de datos específicos y tienen esquemas flexibles para crear aplicaciones modernas. Las bases de datos NoSQL son ampliamente reconocidas porque son fáciles de desarrollar, por su funcionalidad y el rendimiento a escala. Las bases de datos NoSQL utilizan una variedad de modelos de datos para acceder y administrar datos (Véase Figura 13).

Estos tipos de bases de datos están optimizados específicamente para aplicaciones que requieren grandes volúmenes de datos, baja latencia y modelos de datos flexibles, lo que se logra mediante la flexibilización de algunas de las restricciones de coherencia de datos en otras bases de datos. Considere el ejemplo de modelado del esquema para una base de datos simple de cualquier tema:

- En una base de datos relacional, un registro a menudo se enmascara (o "normaliza") y se almacena en tablas separadas, y las relaciones se definen mediante restricciones de claves primarias y foráneas. El modelo relacional está diseñado para permitir que la base de datos aplique la integridad referencial entre tablas en la base de datos, normalizada para reducir la redundancia y, generalmente, está optimizada para el almacenamiento.
- En una base de datos NoSQL, el registro de un libro generalmente se almacena como un documento JSON. En este modelo, los datos están optimizados para un desarrollo intuitivo y escalabilidad horizontal.

Las bases de datos NoSQL se adaptan perfectamente a muchas aplicaciones modernas que requieren bases de datos flexibles, escalables, de alto

---

<sup>20</sup> Información tomada de:

Fernando Sáenz Pérez. (18 de Enero de 2011). Práctica 2. Ejemplo de diseño de bases de datos relacionales. 18 de Enero de 2011, de MTIG Sitio web: [https://www.fdi.ucm.es/profesor/fernan/MTIG\\_/Pr%C3%A1ct.%202.%20Dise%C3%B1o.pdf](https://www.fdi.ucm.es/profesor/fernan/MTIG_/Pr%C3%A1ct.%202.%20Dise%C3%B1o.pdf)



rendimiento y altamente funcionales para proporcionar excelentes experiencias de usuario.

- **Flexibilidad:** Generalmente ofrecen esquemas flexibles que permiten un desarrollo más rápido y más iterativo. El modelo de datos flexible hace que las bases de datos NoSQL sean ideales para datos semiestructurados y no estructurados.
- **Escalabilidad:** Generalmente están diseñadas para escalar usando clústeres distribuidos de hardware en lugar de escalar añadiendo servidores caros y sólidos.
- **Alto rendimiento:** Están optimizada para modelos de datos específicos y patrones de acceso que permiten un mayor rendimiento que el intento de lograr una funcionalidad similar con bases de datos relacionales.
- **Altamente funcional:** Proporcionan API's altamente funcionales y tipos de datos que están diseñados específicamente para cada uno de sus respectivos modelos de datos.

### **Bases de datos deductivas o lógicas**

Se utilizan generalmente en buscadores, pero pueden usarse de otras formas. Permiten almacenar los datos y consultarlos a través de búsquedas que utilizan reglas y normas previamente almacenadas. Sus principales características son:

- Permite expresar consultas por medio de reglas lógicas.
- Soporta conjuntos de datos complejos.
- Se puede inferir información a través de los datos almacenados.
- Usan fórmulas matemáticas o algoritmos lógicos.

### **Bases de datos multidimensionales.**

Estas bases de datos utilizan conceptualmente la idea de un cubo de datos. Donde las informaciones se almacenan en la intersección de tres o más atributos. Esta concepción puede ser algo compleja pero su uso es bastante simple. Algunas de sus principales características son:

- No emplean ninguna jerarquía.
- Facilita tanto la búsqueda como la modificación posterior.
- Utiliza un espacio menor de almacenamiento.
- Tiene acceso a grandes cantidades de información.

## Sistema Gestor de Bases de Datos

Un sistema gestor de bases de datos (SGBD) es una aplicación que permite a los usuarios definir, crear y mantener una base de datos, y proporciona acceso controlado a la misma. En general, un SGBD proporciona los siguientes servicios:

- Permite la definición de la base de datos mediante el lenguaje de definición de datos (DDL – Data Description Language).
- Permite la inserción, actualización, eliminación y consulta de datos mediante el lenguaje de manejo o manipulación de datos (DML - Data Manipulation Language).
- Proporciona un acceso controlado a la base de datos mediante:
  - Un sistema de seguridad, de modo que los usuarios no autorizados no puedan acceder a la base de datos, mediante el lenguaje de control de datos (DCL - Data Control Language).
  - Un sistema de integridad que mantiene la integridad y la consistencia de los datos.
  - Un sistema de control de recuperación que restablece la base de datos después de que se produzca un fallo del hardware o del software.

La principal herramienta de un SGBD es la interfaz de programación con el usuario. Este lenguaje comúnmente se denomina SQL (Structure Query Language), está estandarizado por la ISO 1, es decir, todos los manejadores de BD que soportan SQL deben tener la misma sintaxis a la hora de aplicar el lenguaje.



**Figura 13. Bases de datos NoSQL y Relacionales**

Fuente: Imagen tomada de Internet<sup>21</sup>

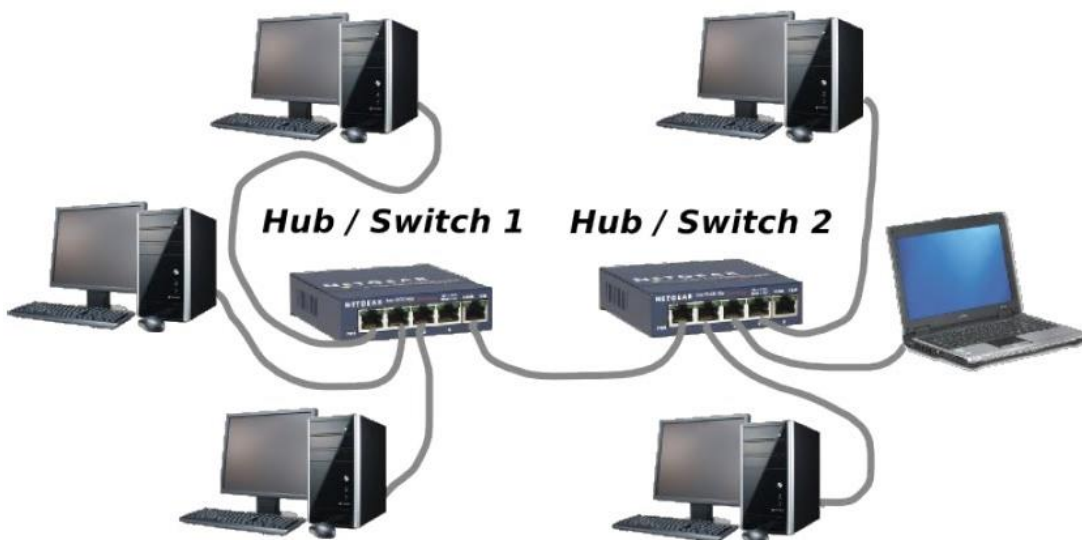
---

<sup>21</sup> Imagen tomada de:

<https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>

### 3.6 Redes de datos

Una red de computadoras es un sistema de interconexión entre equipos que permite compartir recursos e información; para ello, es necesario contar no sólo con las computadoras, también con tarjetas de red, cables de conexión (Véase Tabla 8), dispositivos periféricos y el software conveniente. Las redes se clasifican de acuerdo con su alcance geográfico en PAN, LAN, MAN y WAN. Una red de área local está formada por computadoras, periféricos y los elementos de conexión de los mismos. Las computadoras pueden desarrollar dos funciones: como servidores o estaciones de trabajo. Los elementos de conexión son los cables, tarjetas de red y los dispositivos de interconectividad como los hubs y switches (Véase Figura 14).



**Figura 14. Medios de transmisión de una red**  
Fuente: Imagen tomada de internet<sup>22</sup>

---

<sup>22</sup> Imagen tomada de:  
<https://sites.google.com/site/tema4principiosderedes18/estructura-fisica-de-una-red-local/red-cableada-con-dos-hubs-o-switches>

Dentro de los cables de conexión usualmente utilizados en este tipo de redes se tienen:

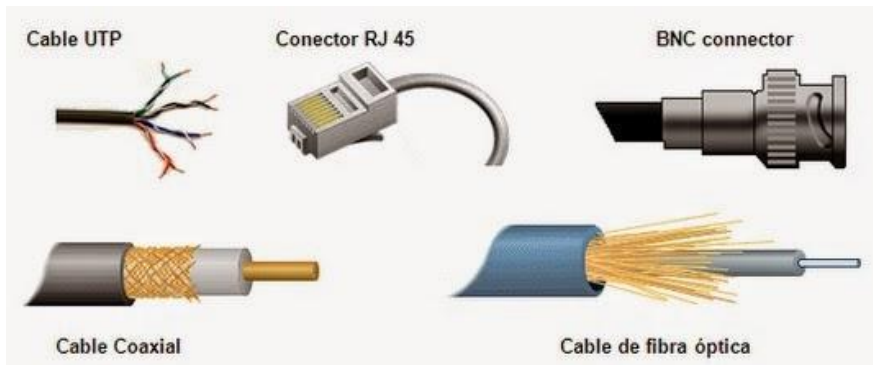
Tabla 8. Características de cables de conexión

Tipo de cable	Descripción física	Características de transmisión	Conectividad y alcance geográfico de acuerdo a IEEE 802.3
<b>UTP</b>	Cada cable de este tipo está compuesto por una serie de pares de cables trenzados. Los pares se trenzan para reducir la interferencia entre pares adyacentes. Normalmente una serie de pares se agrupan en una única funda de color codificado para reducir el número de cables físicos que se introducen en un conducto.	Acrónimo de Unshielded Twisted Pair o Cable trenzado sin apantallar. Son cables que se utilizan para diferentes tecnologías de red local. Existen: 100 BASE T: Velocidad 100 Mbps 10 BASE T: Velocidad 10 Mbps	Distancia máxima por segmento: 100 m  Número de nodos máxima por segmento: 30  Diámetro: .4 y .9  Conectores: RJ45
<b>STP</b>		Acrónimo de Shielded Twisted Pair o Par trenzado apantallado. Son cables de cobre aislados dentro de una cubierta protectora. Existen: <ul style="list-style-type: none"> <li>• 100 BASE T: Velocidad 100 Mbps</li> <li>• 10 BASE T: Velocidad 10 Mbps</li> </ul>	
<b>Coaxial</b>	Compuesto de un hilo conductor central de cobre rodeado por una malla de hilos de cobre. El espacio entre el hilo y la malla lo ocupa un conducto de plástico que separa los dos conductores y mantiene las propiedades eléctricas.	Hay dos tipos de cable coaxial:  Cable fino (Thinnet). 10 base 2  Cable grueso (Thicknet). 10 base 5 10 BASE T : Velocidad 10 Mbps	<b>Para el cable coaxial 10base 5 tenemos:</b> Longitud Máxima: 500 mts. Impedancia: 50 Ω. Diámetro del conductor: 2.17 mm <b>Para el cable coaxial 10base 2 tenemos:</b> Longitud máxima de 185 mts. Impedancia de 50 Ω, RG58 Diámetro del conductor de 0.9 mm
<b>Fibra óptica</b>	Constituido por uno o más hilos de fibra de vidrio, cada fibra de vidrio consta de: Un núcleo central de fibra con un alto índice de refracción. Una cubierta que rodea al núcleo, de material similar, con un índice de refracción ligeramente menor. Una envoltura que aísla las fibras y evita que se produzcan interferencias entre fibras adyacentes, a la vez que proporciona protección al núcleo.	Ancho de banda: 2 GHz  Capacidad Máxima: 2 Gbps	Velocidades muy altas, superiores al GHz.  Bc muy grande (50Ghz máx., 2Ghz típico)  Rmax enorme (2Gbps máx.)

Fuente: Rangel Licea, Macías Ramos. (2009). Apuntes de las asignaturas de Redes de Datos I y Redes de Datos II. México: DGAPA-PAPIME.

Los dispositivos de interconexión (Véase Figura 15) proporcionan la capacidad de extender la distancia de cobertura de una LAN, interconectar redes distantes o distintas y acceder a recursos centralizados; de la misma manera, reducen los dominios de colisión y mejoran el rendimiento de las redes.

#### Medios físicos de conexión



**Figura 15. Medios físicos de conexión en una red**  
Fuente: Imagen tomada de Internet<sup>23</sup>

#### 1) PAN

Las redes PAN's (Personal Area Networks o redes de área personal) están destinadas para comunicaciones entre dispositivos que pertenecen a un solo propietario a través de distancias cortas, suelen ser por lo regular de 10 metros como máximo. Por citar algunos ejemplos, están las comunicaciones entre computadoras, celulares, impresoras y equipos de televisión.



**Figura 16. Red PAN**  
Fuente: Imagen tomada de Internet<sup>24</sup>

<sup>23</sup> Imagen tomada de:  
<http://cipumadayana.blogspot.com/2014/10/medios-de-transmision.html>

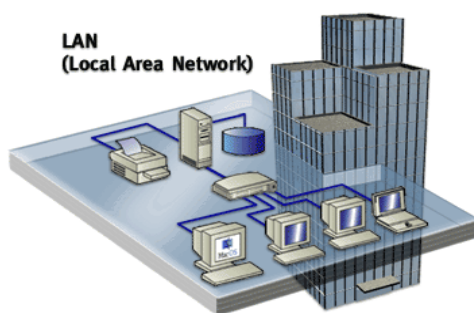
<sup>24</sup> Imagen tomada de:  
<https://sites.google.com/site/asfavnujqwef/tipos-de-redes/red-de-area-personal>

## 2) LAN

Las redes LAN's (Local Area Networks o redes de área local) pueden variar en términos de:

- Tamaño de área a cubrir
- Cantidad de usuarios conectados
- Cantidad y tipos de servicios disponibles

Este tipo de redes generalmente cubren una cierta área geográfica, proporcionando servicios a personas dentro de una estructura organizacional común, como una empresa o una escuela. Por lo general este tipo de redes son administradas por una organización única, el control administrativo que rige las políticas de seguridad y control de acceso está implementado en el nivel de red.

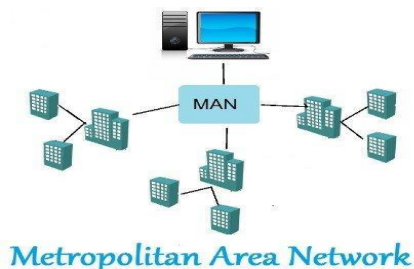


**Figura 17. Red LAN**

Fuente: Imagen tomada de Internet<sup>25</sup>

## 3) MAN

Mejor conocidas como Metropolitan Area Networks o redes de área metropolitana, en este tipo de red, organizaciones individuales alquilan conexiones a través de una red de proveedores de servicios de telecomunicaciones. Las redes MAN, son redes cuyo diámetro no va más allá de 50 km, y responde a la necesidad de un sistema de comunicación de tamaño intermedio con beneficios mayores que aquellos que pueden brindar las redes LAN o WAN.



**Figura 18. Red MAN**

Fuente: Imagen tomada de Internet<sup>26</sup>

---

<sup>25</sup> Imagen tomada de:

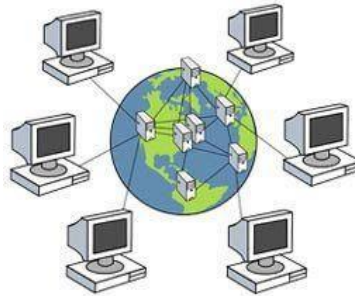
<https://sites.google.com/site/pagedraxpaquetes4/redes/lan>

<sup>26</sup> Imagen tomada de:

<https://pc-solucion.es/2018/03/29/man/>

#### 4) GAN

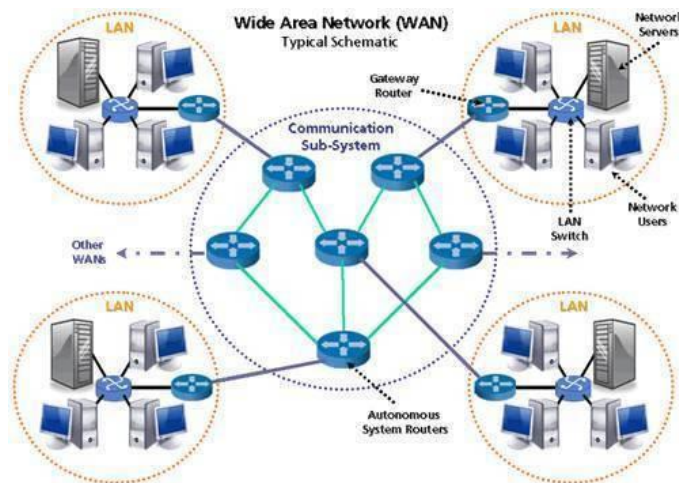
Las Global Area Networks permiten la conexión de una o varias LAN pertenecientes a diferentes países. Es un servicio de comunicación móvil que ofrece datos, voz y fax de alta calidad a velocidades de hasta 64 kbps.



**Figura 19. Red GAN**  
Fuente: Imagen tomada de Internet<sup>27</sup>

#### 5) WAN

Las Wide Area Networks o Redes de área amplia, son redes que conectan LAN's en ubicaciones separadas geográficamente, utilizan dispositivos de red diseñados específicamente para realizar las interconexiones entre las LAN. Dada la importancia de estos dispositivos para la red, la configuración, instalación y mantenimiento de éstos son aptitudes complementarias de la función de una red de la organización.



**Figura 20. Red WAN**  
Fuente: Imagen tomada de Internet<sup>28</sup>

<sup>27</sup> Imagen tomada de:  
<https://www.pinterest.es/pin/631418810236034448/>

<sup>28</sup> Imagen tomada de:  
<https://www.adslfaqs.com/que-es-una-wan-definicion-caracteristicas-ejemplos/>

A manera de resumen, se muestra la siguiente tabla con algunas características de las redes mencionadas.

*Tabla 9. Características de redes existentes.*

Tipo de red	Equipos usados en esta red	Estándar	Banda de frecuencia	Rango nominal	Máxima velocidad transmisión
<b>WPAN</b>	Bluetooth	IEEE 802.15.1	2.4GHz	10 m	720 Kbps
	IrDA	IrDA	850-900 nm	1 m	16 Mbps
	ZigBee	IEEE 802.15.4	868 MHz, 900 MHz, 2.4 GHz	10 m	250 Kbps
	UWB	IEEE 802.15.3	3.1-10.6 GHz (USA) 3.4-4.8 GHz & 6-8.5 GHz (Europea)	10 m	480 Mbps
<b>WLAN</b>	wifi	IEEE 802.11	2.4 / 5 GHz	100 m	1 Mbps
		IEEE 802.11a	5 GHz	100 m	48 Mbps
		IEEE 802.11b	2.4 GHz	100 m	11 Mbps
		IEEE 802.11g	2.4 GHz	100 m	54 Mbps
		IEEE 802.11n	2.4 / 5 GHz	250 m	600Mbps
		IEEE 802.11ac	5 GHz	50 km	1.3 Gbps
<b>WMAN</b>	WiMAX	IEEE 802.16	2-11 GHz y 10-66 GHz	50 km	70 Mbps
<b>WMAN</b>	Móvil	AMPS, GSM, GPRS, UMTS, HSDPA, LTE	700 MHz, 850 MHz, 900 MHz, 1800 MHz, 1900 MHz, 2100 MHz, 2600 MHz	> 50 km	1 Gbps
	Satélite	DVB-S2	3-30 GHz	> 50 km	60 Mbps

**Fuente: Jordi Salazar. (2005). Redes Inalámbricas. Republica Checa: Erasmus+.**

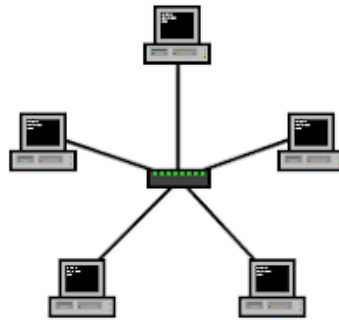
Dentro de las redes de datos, existe otro término muy importante denominado como topología de red, o simplemente topología. El término topología se refiere a la forma en la que está diseñada la red, esto es, tanto física como lógicamente. Dos o más dispositivos se conectan a un enlace; dos o más enlaces forman una topología.

La topología de una red es la representación geométrica de la relación entre todos los enlaces y los dispositivos que los enlazan entre sí (habitualmente llamados nodos). Hay cinco posibles topologías físicas básicas: malla, estrella, anillo, árbol y bus además existe la posibilidad de combinarlas lo que se conoce como topología híbrida, las cuales se describen a continuación.



## 1) Estrella

La propiedad más importante de la topología de estrella es que cada estación se enlaza en forma radial a un nodo central a través de una conexión directa de punto a punto. En la configuración de estrella, una transmisión de una estación entra al nodo central, de donde se retransmite a todos los enlaces de salida. Por consiguiente, aunque el arreglo físico del circuito se asemeje a una estrella, se configura lógicamente como un bus, es decir, las transmisiones desde cualquiera de las estaciones las reciben todas las demás estaciones.



**Figura 21. Topología estrella.**  
Fuente: Imagen tomada de Internet<sup>29</sup>

Entre las ventajas de este tipo de topología encontramos:

- Si este tipo de red falla, solamente el enlace con problemas se verá afectado, todos los demás enlaces permanecen activos.
- Este tipo de topología es más barata, ya que cada dispositivo necesita solamente un enlace y un puerto de entrada/salida para conectarse a cualquier número de dispositivos.

Por otro lado entre las desventajas podemos mencionar:

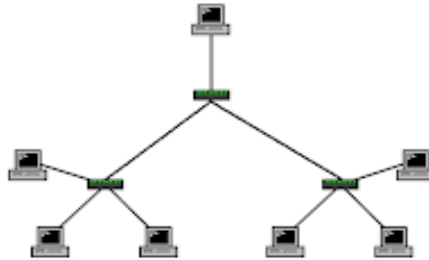
- La red sólo es tan confiable como el nodo central. Cuando falla el nodo central, falla el sistema.
- Cuando es crítica la falla de cualquier entidad dentro de la red, hasta el grado de interrumpir el servicio de toda la red, a esa entidad se le llama recurso crítico. Así, el nodo central en una configuración de estrella es un recurso crítico.

---

<sup>29</sup> Imagen tomada de:  
<https://sites.google.com/site/telecomunicacionaa/topologia-de-estrella>

## 2) Árbol

Una topología árbol es una configuración jerárquica. Como en la estrella, los nodos del árbol están conectados a un concentrador central que controla el tráfico de la red. Sin embargo, no todos los dispositivos se conectan directamente al concentrador central. La mayoría de los dispositivos se conectan a un concentrador secundario que, a su vez, se conecta al concentrador central.



**Figura 22. Topología de árbol.**  
Fuente: Imagen tomada de Internet<sup>30</sup>

Entre las ventajas de este tipo de topología encontramos:

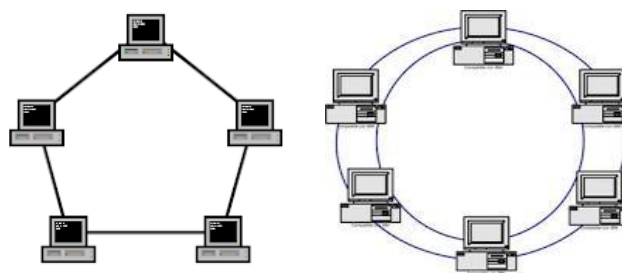
- Permite que se conecten más dispositivos a un único concentrador central.
- Incrementar la distancia que puede viajar la señal entre dos dispositivos.
- Permite aislar y priorizar las comunicaciones de distintas computadoras.

Por otro lado entre las desventajas podemos mencionar:

- La red sólo es tan confiable como el nodo central. Cuando falla el nodo central, falla el sistema.

## 3) Anillo

En la topología en anillo cada dispositivo tiene una línea de conexión dedicada y punto a punto solamente con los dispositivos que están a sus lados. La señal pasa a lo largo del anillo en una dirección, o de dispositivo a dispositivo, hasta que alcanza su destino. Cada dispositivo del anillo incorpora un repetidor. Cuando un dispositivo recibe una señal para otro dispositivo, su repetidor regenera los bits y los retransmite al anillo.



**Figura 23. Topología de anillo.**  
Fuente: Imagen tomada de Internet<sup>31</sup>

<sup>30</sup> Imagen tomada de: <https://sites.google.com/site/telecomunicacionaa/topologia-de-arbol>

<sup>31</sup> Imagen tomada de: <https://sites.google.com/site/telecomunicacionaa/topologia-de-anillo>

Entre las ventajas de este tipo de topología encontramos:

- Es relativamente fácil de instalar y reconfigurar una topología en anillo.
- Para añadir o quitar dispositivos, solamente hay que mover dos conexiones.
- Las fallas se pueden aislar de forma sencilla.

Por otro lado entre las desventajas podemos mencionar:

- El tráfico unidireccional puede ser una desventaja. En anillos sencillos, una rotura del anillo (como por ejemplo una estación inactiva) puede inhabilitar toda la red. Esta debilidad se puede resolver usando un anillo dual o un conmutador capaz de puentear la rotura.

#### 4) Bus

Una topología bus es una topología de difusión. En esta topología todos los nodos están conectados al mismo canal. Es necesario conectar terminadores a cada extremo de la topología para absorber las señales que se reflejen. Si se usa cable coaxial sin terminadores, las señales que se reflejen se repetirán por la red, dejando la red inutilizable.

La propiedad más característica de una topología bus es que el control se distribuye entre todos los nodos interconectados.



**Figura 24. Topología de bus.**  
Fuente: Imagen tomada de Internet<sup>32</sup>

Entre las ventajas de este tipo de topología encontramos:

- El costo y la facilidad de instalación. Dado que esta topología utiliza una distribución de cableado sencilla, cuesta menos y es más fácil de implementar que otras topologías.

Por otro lado, entre las desventajas podemos mencionar:

- Si un segmento de cable o de backbone se rompe o falla, la red también fallará.
- Sólo un nodo puede transmitir datos a la vez. Si dos o más nodos tratan de enviar datos al mismo tiempo, se producirá una colisión; esto requerirá un procedimiento de recuperación, con lo que se alentará la red.
- Cuando se produce una colisión, todos los datos deben ser reenviados.

---

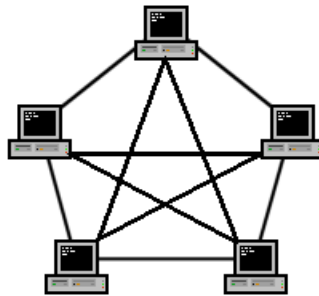
<sup>32</sup> Imagen tomada de:

<https://sites.google.com/site/telecomunicacionaa/topologia-de-bus>

- Un proceso llamado acceso múltiple con detección de portadora y detección de colisiones (CSMA/CD) impide que se produzca otra colisión. CSMA/CD es un proceso en virtud del cual cada nodo espera su turno para transmitir datos.

## 5) Malla

En una topología en malla, cada dispositivo tiene un enlace punto a punto y dedicado con cualquier otro dispositivo. El término dedicado significa que el enlace conduce el tráfico únicamente entre los dos dispositivos que conecta. Por tanto, una red en malla completamente conectada necesita  $n(n-1)/2$  canales físicos para enlazar  $n$  dispositivos.



**Figura 25. Topología de malla.**  
Fuente: Imagen tomada de Internet<sup>33</sup>

Entre las ventajas de este tipo de topología encontramos:

- El uso de los enlaces dedicados garantiza que cada conexión sólo debe transportar la carga de datos propia de los dispositivos conectados, eliminando el problema que surge cuando los enlaces son compartidos por varios dispositivos.
- Una topología en malla es robusta. Si un enlace falla, no inhabilita todo el sistema.
- Otra ventaja es la privacidad o la seguridad. Cuando un mensaje viaja a través de una línea dedicada, solamente lo ve el receptor adecuado.

Por otro lado, entre las desventajas podemos mencionar:

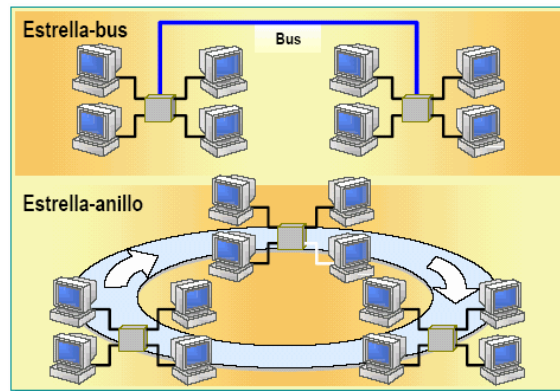
- La cantidad de cable y el número de puertos de entrada/salida necesarios.
- La instalación y reconfiguración de la red es difícil, debido a que cada dispositivo debe estar conectado a cualquier otro.
- La masa de cables puede ser mayor que el espacio disponible para acomodarla (en paredes, techos o suelos).
- Finalmente, el hardware necesario para conectar cada enlace (puertos de E/S y cables) puede ser extremadamente caro.
- El mantenimiento de la red puede ser muy complejo.

<sup>33</sup> Imagen tomada de:

<https://sites.google.com/site/topologiasdered708/home/topologia-de-red-de-maya>

## 6) Híbridas

A menudo, una red combina varias topologías mediante subredes enlazadas entre sí para formar una topología mayor. Por ejemplo, un departamento de una empresa puede decidir usar una topología de bus mientras otro puede tener un anillo. Ambas pueden ser conectadas entre sí a través de un controlador central mediante una topología en estrella.



**Figura 26. Topología híbrida.**

Fuente: Imagen tomada de Internet <sup>34</sup>

Entre las ventajas de este tipo de topología encontramos:

- Si falla algún nodo, no afecta a toda la red.
- Rápido acceso a la red.

Por otro lado, entre las desventajas podemos mencionar:

- Si el concentrador principal se cae, cae toda la red.
- Su costo de implementación es muy elevado.

Para que todos los tipos de topologías mencionadas puedan funcionar, son necesarios diversos tipos de equipo que se mencionarán a continuación.

### Equipos de comunicación

#### a) Router o enrutador

El router es una computadora diseñada para fines especiales que desempeña una función clave en el funcionamiento de cualquier red de datos. Los routers son los principales responsables de la interconexión de redes por medio de: la determinación de la mejor ruta para enviar paquetes el envío de paquetes a su destino. Los routers envían paquetes al aprender sobre redes remotas y al mantener la información de enrutamiento. El router es la unión o intersección que conecta múltiples redes IP.

---

<sup>34</sup> Imagen tomada de:

<https://sites.google.com/site/tecnologia4a16/tipos-de-redes/redes-lan/conceptos-necesarios/topologia-hibrida>

La principal decisión de envío de los routers se basa en la información de Capa 3, la dirección IP de destino. La tabla de enrutamiento del router se utiliza para encontrar la mejor coincidencia entre la dirección IP de destino de un paquete y una dirección de red en la tabla de enrutamiento. La tabla de enrutamiento determinará finalmente la interfaz de salida para enviar el paquete y el router lo encapsulará en la trama de enlace de datos apropiada para dicha interfaz de salida.



**Figura 27. Router inalámbrico marca Cisco, modelo RV110W**  
Fuente: Imagen tomada de Internet<sup>35</sup>

#### **b) Repetidor**

Un repetidor es un dispositivo electrónico que recibe una señal débil o de bajo nivel y la retransmite a una potencia o nivel más alto, de tal modo que se puedan cubrir distancias más largas sin degradación o con una degradación tolerable. El término repetidor se creó con la telegrafía y se refería a un dispositivo electromecánico utilizado para regenerar las señales telegráficas.

El uso del término ha continuado en telefonía y transmisión de datos. En telecomunicación el término repetidor tiene los siguientes significados normalizados:

1. Un dispositivo analógico que amplifica una señal de entrada, independientemente de su naturaleza (analógica o digital).
2. Un dispositivo digital que amplifica, conforma, re temporiza o lleva a cabo una combinación de cualquiera de estas funciones sobre una señal digital de entrada para su retransmisión. En el modelo de referencia OSI el repetidor opera en el nivel físico.

---

<sup>35</sup> Imagen tomada de:  
[https://www.cisco.com/c/es\\_mx/support/routers/rv110w-wireless-n-vpn-firewall/model.html](https://www.cisco.com/c/es_mx/support/routers/rv110w-wireless-n-vpn-firewall/model.html)



**Figura 28. Repetidor marca Cisco Linksys, modelo RE1000**  
Fuente: Imagen tomada de Internet<sup>36</sup>

**c) Hub**

Un hub o concentrador es un equipo de redes que permite conectar entre sí otros equipos y retransmite los paquetes que recibe desde cualquiera de ellos a todos los demás. Los hubs han dejado de ser utilizados, debido al gran nivel de colisiones y tráfico de red que propician.



**Figura 29. Hub marca Cisco Linksys USB de 4 puertos**  
Fuente: Imagen tomada de Internet<sup>37</sup>

**d) Bridge**

Un bridge o puente es un dispositivo que se utilizaba con mayor frecuencia en los inicios de la LAN para conectar dos segmentos de red física. Los switches pueden utilizarse para realizar esta operación, a la vez que permiten la conectividad del dispositivo final con la LAN. Muchas otras tecnologías se desarrollaron en torno a los switches LAN.

---

<sup>36</sup> Imagen tomada de:  
<https://www.redeszone.net/analisis/repetidores-wifi/re1000/>

<sup>37</sup> Imagen tomada de:  
<https://www.amazon.com.mx/Linksys-USBHUB04-cisco-linksys-USB-4-Port/dp/B00004Z6PI>



**Figura 30. Bridge marca Cisco Linksys, modelo WES610N**  
Fuente: Imagen tomada de Internet<sup>38</sup>

#### e) Switch

Un conmutador o switch es un dispositivo digital de lógica de interconexión de redes de computadores que opera en la capa 2 (nivel de enlace de datos) del modelo OSI. Su función es interconectar dos o más segmentos de red, de manera similar a los puentes (bridges), pasando datos de un segmento a otro de acuerdo con la dirección MAC de destino de las tramas en la red. Al igual que los puentes, dado que funcionan como un filtro en la red, mejoran el rendimiento y la seguridad de las LAN's (Local Area Network- Red de Área Local).

Los switches permiten la segmentación de la LAN en distintos dominios de colisiones. Cada puerto de un switch representa un dominio de colisiones distinto y brinda un ancho de banda completo al nodo o a los nodos conectados a dicho puerto. Una LAN puede tener un switch centralizado que conecta a hubs que todavía brindan conectividad a los nodos. O bien, una LAN puede tener todos los nodos conectados directamente a un switch.



**Figura 31. Switch marca Cisco Fast Ethernet modelo SF250-24**  
Fuente: Imagen tomada de Internet<sup>39</sup>

---

<sup>38</sup> Imagen tomada de:  
<https://www.redeszone.net/2011/08/19/cisco-linksys-wes610n-el-bridge-doble-banda-con-4-puertos-ethernet-10100/>

<sup>39</sup> Imagen tomada de:  
<https://www.cyberpuerta.mx/Computo-Hardware/Redes/Switches/Switch-Cisco-Fast-Ethernet-SF250-24-24-Puertos-10-100Mbps-Puertos-SFP-12-8Gbit-s-8000-Entradas-Gestionado.html>



A continuación, de manera general, se muestra un resumen de las características de estos equipos:

*Tabla 10. Características físicas de equipos más utilizados.*

Equipo	Puertos o Interfaces	Medios de transmisión usados	Utilizados en red
<b>Router</b>	Generalmente interfaces FastEthernet para conexiones a diferentes LAN y distintos tipos de interfaces WAN para conectar una variedad de enlaces seriales, entre ellos T1, DSL e ISDN	Cable consola Cable serial	MAN
<b>Repetidor</b>	Como mínimo una salida Ethernet para el cable amarillo y otra para teléfono	Cable cruzado Fibra óptica	LAN MAN
<b>Hub</b>	Varios puertos RJ45 integrados, desde 4, 6, 8, 12, 16 y hasta 24.	Cable UTP.	LAN
<b>Bridge</b>	LAN	Cable Ethernet	LAN MAN
<b>Switch</b>	Varios puertos integrados de tipo RJ45 (desde 4 puertos, 8 puertos, 16 P, 32 P y hasta 52 Puertos).	Interconectan las redes por medio de cables UTP.	LAN

Fuente: Información tomada de Internet<sup>40</sup>

---

<sup>40</sup> Información tomada de:  
<https://sites.google.com/site/wikiejemploalcccna2017/caracteristicas-basicas-de-los-swichts>  
<http://welinfocto.blogspot.com/2012/03/113-puente.html>  
<https://www.ecured.cu/Repetidor>  
<https://sites.google.com/site/redestelematicas2sti/1a-evaluacion/tema-2-routers/2-3-interface-router?tmpl=%2Fsystem%2Fapp%2Ftemplates%2Fprint%2F&showPrintDialog=1>

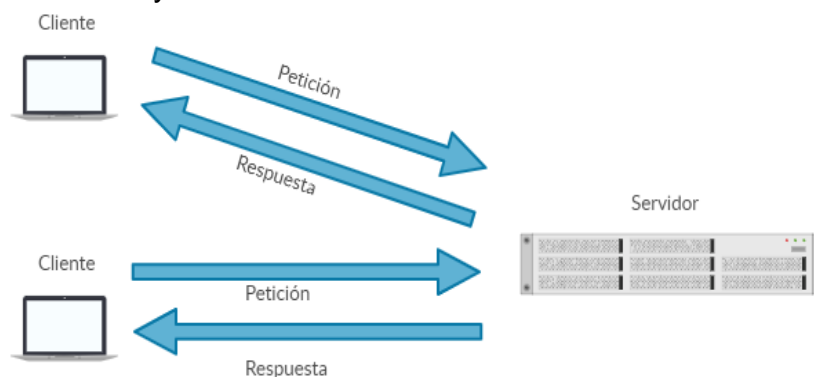
### 3.7 Servidores

El término servidor tiene dos significados en el ámbito informático. El primero hace referencia a la computadora que pone recursos a disposición a través de una red, y el segundo se refiere al programa que funciona en dicha computadora. En consecuencia aparecen dos definiciones de servidor:

- **Definición Servidor (hardware):** un servidor basado en hardware es una máquina física integrada en una red informática en la que, además del sistema operativo, funcionan uno o varios servidores basados en software. Una denominación alternativa para un servidor basado en hardware es "host" (término inglés para "anfitrión"). En principio, toda computadora puede usarse como "host" con el correspondiente software para servidores.
- **Definición Servidor (software):** un servidor basado en software es un programa que ofrece un servicio especial que otros programas denominados clientes (clients) pueden usar a nivel local o a través de una red. El tipo de servicio depende del tipo de software del servidor. La base de la comunicación es el modelo cliente-servidor y, en lo que concierne al intercambio de datos, entran en acción los protocolos de transmisión específicos del servicio.

#### Funcionamiento básico de un servidor.

La puesta a disposición de los servicios del servidor a través de una red informática se basa en el modelo cliente-servidor (Véase Figura 32), concepto que hace posible distribuir las tareas entre las diferentes computadoras y hacerlas accesibles para más de un usuario final de manera independiente. Este es el único modo de asegurar que los clientes como el navegador web o los clientes de correo electrónico siempre tengan la posibilidad de acceder al servidor activamente y de usar el servicio en función de sus necesidades.



**Figura 32. Arquitectura básica cliente-servidor**

Fuente: Imagen tomada de internet<sup>41</sup>

<sup>41</sup> Imagen creada con:  
app.creately.com

Existen gran cantidad de tipos de servidores o roles que estos pueden desempeñar (Véase Figura 33), a continuación, se describen algunos.

- **Servidor de archivos**  
Un servidor de archivos es aquel que se encarga de almacenar los datos a los que acceden los diferentes clientes a través de una red.
- **Servidor de Directorio Activo/Dominio**  
Un servidor de dominio es una computadora especialmente diseñada con arquitectura de alto rendimiento, en el cual se instala un sistema operativo de servidor y un software que aplica políticas para autorizar o denegar características del sistema operativo, en ciertos equipos de cómputo conectados en las redes (LAN, VLAN, WAN, VPN, WLAN, etcétera.).
- **Servidor de Impresión**  
Un Servidor de Impresión o Servicio de impresión a gran escala (Print Server) es un concentrador, o más bien un servidor, que conecta una impresora a red para que cualquier PC pueda acceder a ella e imprimir trabajos.
- **Servidor de Correo**  
Un servidor de correo electrónico consta de varios módulos de software cuya interacción hace posible la recepción, el envío y el reenvío de correos electrónicos, Por regla general funciona mediante el protocolo de transferencia simple de correo (SMTP).
- **Servidor de Fax**  
Un servidor de fax (en inglés Fax Server) es un sistema instalado en una red de área local (LAN) que permite a los usuarios de las computadoras que están conectados a la LAN enviar y recibir mensajes de fax.
- **Servidor de juegos**  
Servidores creados específicamente para juegos multijugador online. Estos servidores gestionan los datos del juego online y permiten la interacción sincrónica con el mundo virtual.

- **Servidor Proxy**

El servidor proxy sirve como interfaz de comunicación en las redes informáticas. En su papel de intermediario, el servidor proxy recibe las solicitudes de red y las transmite a través de su propia dirección IP.
- **Servidor Web**

La tarea principal de un servidor web es la de guardar y organizar páginas web y entregarlas a clientes como navegadores web o crawlers. La comunicación entre servidor (software) y cliente se basa en HTTP, es decir, en el protocolo de transferencia de hipertexto o en HTTPS, la variante codificada.
- **Servidor de Base de Datos**

Un servidor de base de datos es un programa informático que posibilita que otros programas puedan acceder a uno o varios sistemas de bases de datos a través de una red. Las soluciones de software con una elevada cuota de mercado son Oracle, MySQL, Microsoft SQL Server, PostgreSQL y DB2.
- **Servidor DNS**

El servidor DNS o servidor de nombres permite la resolución de nombres en una red. Los servidores DNS son de vital importancia para la red informática mundial (WWW), ya que traducen los nombres de host como `www.example.com` en la correspondiente dirección IP.
- **Servidor DHCP**

El protocolo de configuración dinámica de host (en inglés: Dynamic Host Configuration Protocol, también conocido por sus siglas de DHCP) es un protocolo de red de tipo cliente/servidor mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo en una red para que puedan comunicarse con otras redes IP. Así los clientes de una red IP pueden conseguir sus parámetros de configuración automáticamente
- **Servidor FTP (File Transfer Protocol)**

Es un programa especial que se ejecuta en un servidor conectado normalmente en Internet. La función del mismo es permitir el desplazamiento de datos entre diferentes servidores / computadoras.

- Servidor Radius

Radius proviene del acrónimo Remote Access Dial in User Service. Se trata de un protocolo cliente-servidor AAA (autenticación, autorización y contabilidad) que se encarga de gestionar el acceso de los usuarios a las redes a través de un dispositivo NAS.

La comunicación entre un servidor de acceso de red (NAS) y el servidor RADIUS se basa en el protocolo de datagrama de usuario (UDP). Generalmente, el protocolo RADIUS se considera un Servicio sin conexión. Los problemas relacionados con la disponibilidad de los servidores, la retransmisión y los tiempos de espera son tratados por los dispositivos activados por RADIUS en lugar del protocolo de transmisión.

El RADIUS es un protocolo cliente/servidor. El cliente RADIUS es típicamente un NAS y el servidor de RADIUS es generalmente un proceso de daemon. Los servidores de RADIUS reciben las peticiones de conexión del usuario, autentican al usuario, y después devuelven la información de la configuración necesaria para que el cliente entregue el servicio al usuario. Un servidor RADIUS puede funcionar como cliente proxy para otros servidores RADIUS u otro tipo de servidores de autenticación.

Comúnmente, el proceso de autenticación y autorización es el siguiente: El ingreso de un usuario al sistema consiste en un pedido (Solicitud de acceso) desde el NAS hacia el servidor RADIUS y de una correspondiente respuesta (Aceptación de acceso o Rechazo de acceso) desde el servidor. El paquete access-request contiene el nombre de usuario, la contraseña encriptada, la dirección IP NAS, y el puerto.

Cuando el servidor de RADIUS recibe el pedido de acceso del NAS, busca una base de datos para el nombre de usuario enumerado. Si el nombre de usuario no existe en la base de datos, se carga un perfil predeterminado o el servidor RADIUS inmediatamente envía un mensaje Access-Reject (acceso denegado). Este mensaje de acceso rechazado puede estar acompañado de un mensaje de texto que indique el motivo del rechazo.

En RADIUS, la autenticación y la autorización están unidas. Si se encuentra el nombre de usuario y la contraseña es correcta, el servidor RADIUS devuelve una respuesta de acceso-aceptar e incluye una lista de

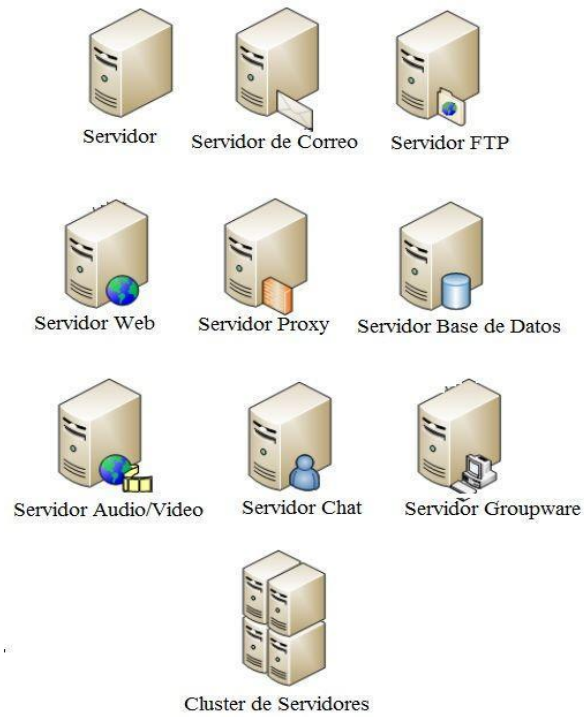
pares de atributo-valor que describe los parámetros que deben usarse en esta sesión. Los parámetros comunes incluyen el tipo de servicio (shell o entramado), el tipo de protocolo, la dirección IP para asignar el usuario (estática o dinámica), la lista de acceso a aplicar o una ruta estática para instalar en la tabla de ruteo de NAS. La información de configuración en el servidor RADIUS define qué se instalará en el NAS.

Las funciones de contabilidad del protocolo RADIUS pueden emplearse independientemente de la autenticación o autorización de RADIUS. Las funciones de contabilidad de RADIUS permiten que los datos sean enviados al inicio y al final de las sesiones, indicando la cantidad de recursos (por ejemplo, tiempo, paquetes, bytes y otros) utilizados durante la sesión. Un Proveedor de servicios de Internet (ISP) puede usar RADIUS para el control de acceso y un software de contabilidad para satisfacer necesidades especiales de seguridad y facturación.

Las transacciones entre el cliente y el servidor RADIUS son autenticadas mediante el uso de un secreto compartido, que nunca se envía por la red. Además, las contraseñas del usuario se envían cifradas entre el cliente y el servidor de RADIUS para eliminar la posibilidad que algún curioso dentro de una red insegura pudiera determinar una contraseña de usuario.

¿Qué ventajas tiene usar Radius?

1. Mejora la seguridad de la red.
2. Mejora el seguimiento de los accesos a la red basado en los nombres de usuarios de clientes.
3. Permite aplicar restricciones a un usuario concreto.
4. Los accesos a la red individuales están encriptados.
5. Es posible desconectar a un único usuario o dispositivo sin afectar al resto, sin cambiar la clave del resto.



**Figura 33. Tipos de servidores**  
Fuente: Imagen tomada de Internet<sup>42</sup>

---

<sup>42</sup> Imagen tomada de:  
<https://www.areatecnologia.com/informatica/servidor-y-tipos.html>

# **CAPÍTULO 4**

# **ANÁLISIS, DISEÑO**

# **E**

# **IMPLEMENTACIÓN**

# **DEL SISTEMA**



## 4.1 Justificación de las herramientas

Al llevar a cabo el análisis de las encuestas mostradas en el capítulo 2 Identificación de la problemática, al apreciarse que el principal problema de los alumnos al utilizar el servicio de Internet dentro de la Facultad es que los equipos van siendo más obsoletos conforme avanza el tiempo y que la velocidad de la RIU no es la apropiada, se busca brindarle a los alumnos una forma más de poderse conectar al sistema de red inalámbrico sin tener que depender de los equipos que se encuentran en las salas de cómputo de la Facultad. Por otro lado, el análisis nos lleva a otro punto, esto es, si logramos darle a los alumnos la posibilidad de poder conectarse desde sus propios dispositivos, se eliminarían los tiempos de espera que usualmente le toma a un alumno para poder ser asignado a una computadora en tiempos donde la afluencia es mayor y por ende el número de computadoras disminuye, y con ello, al mismo tiempo reducir tiempos de traslado, al darles la posibilidad de elegir el lugar de su preferencia para realizar sus actividades y así aprovechar el tiempo de manera más eficaz. Es por ello que se busca desarrollar este sistema de una manera simple pero eficaz, donde desde la instalación, la configuración y hasta el mantenimiento del sistema sea lo más sencillo posible.

Para el diseño e implementación del sistema se utilizará un Sistema Operativo Debian 10 Buster Arquitectura x86-64, un Access Point marca TP-Link, modelo TL-WR840N, una base de datos MariaDB y un servidor freeRadius junto con daloRadius, este último servirá de apoyo para la administración de freeRadius, ya que, al ser un entorno gráfico, será mucho más sencilla su administración. Todas las herramientas mencionadas tienen las características de ser de uso gratuito (aunque cuentan también con algunas características de pago) para lograr que con esto se puedan disminuir costos en la implementación del diseño.

### a) Access Point

Actualmente en el mercado, encontramos opciones de distintas marcas y gran variedad de modelos, dependiendo de las necesidades para las cuales se quiera destinar. Algunos aspectos que se deben tener en cuenta a la hora de elegir alguno pueden ser el precio, cantidad de usuarios soportados, alcance, cantidad de puertos, cantidad de antenas entre otras.

En el mercado existen diferentes marcas que ofrecen Access Point para casi todas las necesidades, entre las más destacables se encuentran Cisco, Linksys, Aruba, TP-Link y Ubiquiti. Más adelante se hará una comparación entre estos, mostrando sus características para así poder elegir el que mejor se adapte a las necesidades de nuestro proyecto. A continuación se muestra una tabla comparativa entre un modelo de cada una de las marcas mencionadas.

Tabla 11. Comparativa de algunos Access Point disponibles en el mercado

Marca y Modelo	TP-Link CPE220	Ubiquiti Networks UniFi AC In-Wall	Linksys LAPN300	Cisco WAP150	Aruba Instant On AP11	NetGear WAC104
Cantidad de puertos LAN	2	3	1	1	1	4
Cantidad de antenas	2	1	1	1	1	1
Tasa de transferencia (máx.)	300 Mbit/s	1000 Mbit/s	1000 Mbit/s	1200 Mbit/s	867 Mbit/s	1200 Mbit/s
Ganancia de la antena (máx.)	12 dBi	2 dBi	1 dBi	3 dBi	5.8 dBi	3 dBi
Frecuencia de banda	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz	2.4 GHz
Precio (2020)	\$1,089.00	\$11,499.00	\$1,929.00	\$3,109.00	\$1,669.00	\$2,340.00
Números de usuarios	25	Hasta 250	-	Hasta 64	-	Hasta 128
Canales	5	No definido	11	5	No definido	5

Fuente: Datos tomados de Internet<sup>43</sup>

El Access Point a utilizar será un equipo TP-Link TL-WR840N que cuenta con dos antenas externas para rangos de frecuencia de 2.4 GHz, interfaz intuitiva, funciones de seguridad WEP, WPA, WAP2. Botón WPS, 4 puertos LAN, 5 canales y hasta un máximo de 25 usuarios.

#### b) Implementación del servidor freeRadius

freeRADIUS es un protocolo de cliente / servidor gratuito y de código abierto que proporciona autenticación de red centralizada en los sistemas. Es un servidor RADIUS de alto rendimiento y rico en funciones que se entrega con servidores y clientes, bibliotecas de desarrollo y utilidades relacionadas con RADIUS. Las siglas RADIUS corresponden a Remote Authentication Dial-In User Server. Es un protocolo de autenticación y autorización para aplicaciones de acceso a la red, en nuestro caso red inalámbrica (wifi).

Este proceso de “autenticación, autorización y registro” se lo conoce más como AAA. Cuando se realiza la conexión con un punto de acceso wifi, en lugar de una clave de red típica, se enviará un nombre de usuario y una contraseña. Esta información se transfiere a un servidor RADIUS sobre el protocolo RADIUS. El servidor RADIUS comprueba que la información de usuario es correcta utilizando esquemas de autenticación como PAP, CHAP o EAP. Si el usuario es aceptado, el servidor autorizará el acceso al sistema del dispositivo y le asigna los recursos de red necesarios para establecer la conexión a la misma tales como una dirección IP, puerta de enlace, etcétera.

<sup>43</sup> Información obtenida de la liga:

<https://www.cyberpuerta.mx/Computo-Hardware/Redes/Access-Points/2/>

RADIUS le permite mantener perfiles de usuario y rastrear el uso para la facturación desde el servidor central. Proporciona soporte para varias bases de datos, incluyendo OpenLDAP, MySQL, Redis, Microsoft AD, Apache Cassandra, PostgreSQL, MariaDB, etcétera. El uso de RADIUS permite la autenticación y autorización para que una red se centralice, y minimiza la cantidad de cambios que deben realizarse al agregar o eliminar nuevos usuarios a una red. FreeRADIUS puede autenticar a los usuarios en sistemas como 802.1x (wifi), acceso telefónico, PPPoE, VPN, VoIP y muchos otros.

En conclusión, se hará uso de este servidor debido a las condiciones en las que se desarrollara el proyecto, es decir, realizar el proyecto utilizando software libre para que con esto se minimicen los costos que podría llevar el implementar algún otro servidor de paga, todo esto sin dejar de lado la parte de la seguridad y el soporte que se incluye en este servidor sin importar que no sea de paga.



**Figura 34. Servidor FreeRADIUS.**  
Fuente: Imagen tomada de Internet<sup>44</sup>

### c) Implementación de la interfaz gráfica daloRadius

daloRadius es una aplicación basada en web para administrar el servidor RADIUS. Es una aplicación avanzada basada en la web para administrar puntos de acceso e implementaciones de ISP. Está escrito en PHP y JavaScript y es compatible con MySQL, MariaDB, PostgreSQL, SQLite, MsSQL como back-end de la base de datos. Viene con un amplio conjunto de características como, listas de control de acceso, informes gráficos, motor de facturación, informes gráficos y contabilidad, permite crear, eliminar y editar usuarios, HotSpot, NAS, y mucho más.

Esto permite una interacción mucho más amigable con la persona que estará a cargo de la administración en dado caso que no sea alguien experimentado en el uso de terminales.

---

<sup>44</sup> Imagen tomada de:  
<https://www.redeszona.net/2018/06/25/servidor-radius-herramienta-freeradius/>



**Figura 35. Herramientas de daloRadius.**

Fuente: Imagen tomada de Internet<sup>45</sup>

#### d) Base de datos MariaDB

Al ser MariaDB una bifurcación de MySQL, es normal que existan similitudes entre los dos, tales como:

- Los protocolos, las API y las estructuras del cliente son idénticos.
- Los archivos de definición de tablas y datos son completamente compatibles.
- Los desarrolladores de MariaDB incluso hacen una fusión mensual del código de su software y MySQL.

Entonces dado a que MariaDB proviene de MySQL, motivo por el cual son tan similares, hay que especificar por qué MariaDB viene ganando cada vez más popularidad. Cabe aclarar que hoy en día MySQL sigue vigente y que habrá que analizar el motivo por el cual se debería usar uno u otro. En este caso el software de gestión de bases de datos MariaDB viene con algunas capacidades adicionales, a continuación, se mencionan 3 de las razones más sobresalientes por encima de MySQL.

##### 1. Medidas de seguridad más estrictas.

La actualización más reciente de MySQL incluyó algunas características nuevas centradas en la seguridad. Sin embargo, MariaDB cuenta con medidas de seguridad adicionales, que incluyen:

- Seguridad interna y verificación de contraseña
- Autenticación PAM y LDAP
- Roles del usuario
- Cifrado de la base de datos

Otro punto a tomar en cuenta, es su control de acceso basado en roles (RBAC). MySQL no ha podido implementar características similares, lo que lo hace algo más vulnerable.

<sup>45</sup> Imagen tomada de:

<https://co.pinterest.com/pin/524810162820573197/>

<https://www.styleofglobal.com/portfolio-item/freeradius-billing-managment-system/>

2. Rendimiento más rápido y más eficiente.

Aunque tanto MySQL como MariaDB son eficientes, este último ha logrado liderar el rendimiento. Esto se debe en parte a su gran selección de motores de bases de datos alternativos. En otras palabras, al optimizar las consultas, MariaDB tiene en cuenta las estadísticas de tabla independientes del motor. Esto incluye variables del sistema, memorias intermedias, cachés, subprocessos, mejoras de rendimiento de tipo de base de datos y tecnología de cifrado.

3. Mejor asistencia al usuario.

Actualmente, al tener una licencia de MySQL cuenta con soporte las 24 horas. Por otro lado, con MariaDB es similar, pero viene con un tiempo de reacción máximo de 30 minutos. Además, el software MariaDB es utilizado por algunas de las compañías más grandes del mundo, incluidas Nokia, Intel y Samsung.

Entonces como conclusión, nuestro servidor FreeRadius en versiones anteriores trabajaba con MySQL, pero actualmente trabaja por defecto con la base de datos MariaDB al realizar la descarga e instalación del mismo. Y aunque se podría seguir trabajando con MySQL (si así se quisiera), por las cuestiones mencionadas, sobre todo la parte de seguridad que es ligeramente mejor que MySQL, se usará MariaDB para una mayor protección de la base de datos donde se almacenarán los usuarios que utilizarán la red.

### **e) Sistema Operativo**

Para la parte de software se hará uso de un sistema operativo libre, esto con la finalidad de reducir costos, tanto por la parte de su adquisición, como por la parte del mantenimiento. Hoy en día los principales competidores en la parte de Sistemas Operativos son Linux y Windows, pero al ser Windows una licencia de tipo privativa, y en la cual el mantenimiento suele ser muchas veces elevado y/o de paga, únicamente nos enfocaremos en versiones procedentes de Linux.

Se buscará optar por aquel sistema operativo que nos brinde seguridad, eficiencia y pueda ser capaz de manejar nuestro servidor, además de contar con un buen soporte, como lo son los sistemas operativos basados en Linux, ya que al ser 'libre', todo el que lo desee puede aportar cambios y/o mejoras al código, haciendo que siempre se mantenga lo más actualizado posible y con ello, al haber mayor gente trabajando detrás de este sistema, las fallas pueden verse minimizadas o arregladas lo más rápido posible. Se buscará aquel sistema que logre trabajar con los menores recursos posibles pero sin llegar a ser ineficaz en su rendimiento.

Además deberá contar con características que sean amigables para la persona que será responsable de llevar a cabo su manejo, es decir, que cuente con interfaces gráficas fáciles de entender y manejar, en caso de que la persona no sea una experta en el manejo de sistemas operativos manejados por consola.

### **Motivos para utilizar Debian y no otra distribución de Linux o Windows**

Debian no tiene una versión server específica como Ubuntu (Ubuntu Server), pero sus principales fortalezas son una excelente optimización de recursos y una gran seguridad, dos aspectos muy importantes para el servidor. Cada administrador puede adaptar Debian a sus necesidades, partiendo de una distribución base y adaptándola a las necesidades de cada uno.

De hecho, en comparación con alguna otra distribución, las características para realizar la instalación de Debian en cuestión de recursos del sistema se pueden visualizar en la siguiente tabla:

*Tabla 12. Características de instalación básica en distribuciones Linux*

	<b>Procesador</b>	<b>Memoria RAM</b>	<b>Disco Duro</b>
<b>CentOS 8</b>	2 GHz	128 Mb – 2 Gb	1.2 Gb – 20 Gb
<b>Red Hat Linux</b>	2.4 GHz	2 Gb – 8 Gb	5 Gb
<b>SUSE</b>	500 MHz - 2.4 GHz	512 Mb – 1 Gb	3 Gb – 5 Gb
<b>Ubuntu 18.04</b>	700 MHz	1 Gb	10 Gb
<b>Debian 9</b>	Pentium 4 a 1 GHz	128 Mb – 1 Gb	2 Gb – 10 Gb
<b>Fedora</b>	400 MHz	768 Mb – 1 Gb	10 Gb

Fuente: Información tomada de Internet <sup>46</sup>

---

<sup>46</sup> Información tomada de:

<https://www.debian.org/releases/jessie/i386/ch03s04.html.es>

<https://docs.fedoraproject.org/es->

[ES/Fedora/16/html/Installation\\_Quick\\_Start\\_Guide/Requirements.html](https://docs.fedoraproject.org/es-ES/Fedora/16/html/Installation_Quick_Start_Guide/Requirements.html)

<https://ubunlog.com/guia-de-instalacion-de-ubuntu-18-04-lts-bionic-beaver/>

<https://documentation.suse.com/es-es/sles/12-SP4/single-html/SLES-installquick/index.html>

<https://www.sololinux.es/instalar-centos-8-paso-a-paso-con-imagenes/>

Entonces como conclusión, podemos destacar a Debian sobre los demás sistemas operativos por lo siguiente:

1. Se puede instalar en más arquitecturas, tanto de 32 como de 64 bits. Por lo cual puede ser ejecutada en más equipos que Ubuntu (por ejemplo), el cual se centra más en equipos domésticos o servidores x86.
2. Al momento de realizar la instalación, se cuenta con opciones más personalizadas.
3. El entorno gráfico por defecto es GNOME, sin embargo, más adelante podremos elegir entre una variedad de entornos gráficos.
4. Podemos mantener paquetes antiguos en versiones más antiguas o bien, mantener el sistema lo más actualizado posible.
5. Suele ser más seguro, ya que tiene paquetes GPG (y PGP). Además, la terminal SSH permite crear conexiones seguras a otras máquinas que tengan SSH instalado.



**Figura 36. Sistema operativo Debian**  
Fuente: Imagen tomada de Internet<sup>47</sup>

---

<sup>47</sup> Imagen tomada de:  
<https://es.wikipedia.org/wiki/Debian>

## 4.2 Diseño del sistema

El diseño del sistema lo conforman básicamente tres elementos importantes que son, un servidor freeradius, el cual fue creado en 1999 y elegido por ser usado diariamente por cerca de 100 millones de personas para acceder a Internet<sup>48</sup>, su configuración puede ser tan sencilla o tan complicada, dependiendo qué nivel de seguridad y control se requiera aprovechar de este servidor, este servidor será el encargado de realizar la autenticación de los usuarios mediante su usuario y contraseña que serán almacenadas en nuestra base de datos MariaDB, siendo este nuestro segundo elemento, el cual es capaz de gestionar bases de datos relacionales y grandes cantidades de datos, por otro lado podemos utilizar este gestor de bases de datos de manera gratuita, y aunque existen versiones más sofisticadas que incluyen algún tipo de pago, para nuestro objetivo, que solo será el almacenar información básica de los usuarios, sin el uso excesivo de muchas tablas, este tipo de base de datos viene a ser la ideal para el diseño de este sistema.

Entonces, nuestra base de datos almacenará la información de los usuarios, el servidor será el encargado de autenticarlos y el Access Point (Véase Figura 37), nuestro tercer elemento, será quien les brindará el acceso a la red inalámbrica. Siendo que este será un diseño inicial, el cual no contendrá mucha información en su base de datos, ni el Access Point soportará muchos usuarios a la vez, solo será para verificar el correcto funcionamiento de todos los elementos actuando en conjunto, por ende, el Access Point elegido será de la marca TP-Link, modelo WR840N, con capacidad de velocidad inalámbrica de 300Mbps<sup>49</sup> y que cumple con el estándar IEE 802.11b / g / n, pero que solo soporta un máximo de 8 a 10 dispositivos funcionando de manera correcta. Sin embargo, para la realización de las pruebas de manera local, es suficiente para lograr el objetivo deseado.

Aunque los elementos mencionados son los que realizarán todo el trabajo, existe un elemento necesario para que todo esto funcione, y esto es un equipo en el cual se instalará nuestro servidor y nuestra base de datos. Para lo cual utilizaremos un equipo de Escritorio (Véase Figura 37) con procesador Intel Core i7-9700 9th Gen, con 8 GB de memoria RAM y Disco Duro 1TB con monitor P2219H, 21.5" HD. En este equipo instalaremos también un sistema operativo Debian 9 Buster en donde descargaremos e instalaremos el servidor y la base de datos.

A continuación, a manera de resumen se muestra una figura con los elementos necesarios para la construcción de nuestro sistema.

---

<sup>48</sup> Fuente:  
<https://freeradius.org/about/>

<sup>49</sup> Fuente:  
<https://www.tp-link.com/mx/home-networking/wifi-router/tl-wr840n/#overview>





### Equipo Servidor

Desktop Intel Core i7-9700 9<sup>th</sup> Gen, up to 4.7GHz  
8 GB RAM / Disco Duro 1TB 7200 RPM.  
Monitor Dell P2219H, 21.5" HD.  
Sistema Operativo Debian 9 Buster con FreeRadius.  
Gestor de base de datos MariaDB



### Access Point

- Sistema Operativo Linux
- Modelo WR840N
- Seguridad WEP, WPA, WPA2
- 4 puertos LAN
- Eficiencia > 80%

**Figura 37. Elementos a utilizar en el sistema**

Fuente: Imagen tomada de internet<sup>50</sup>

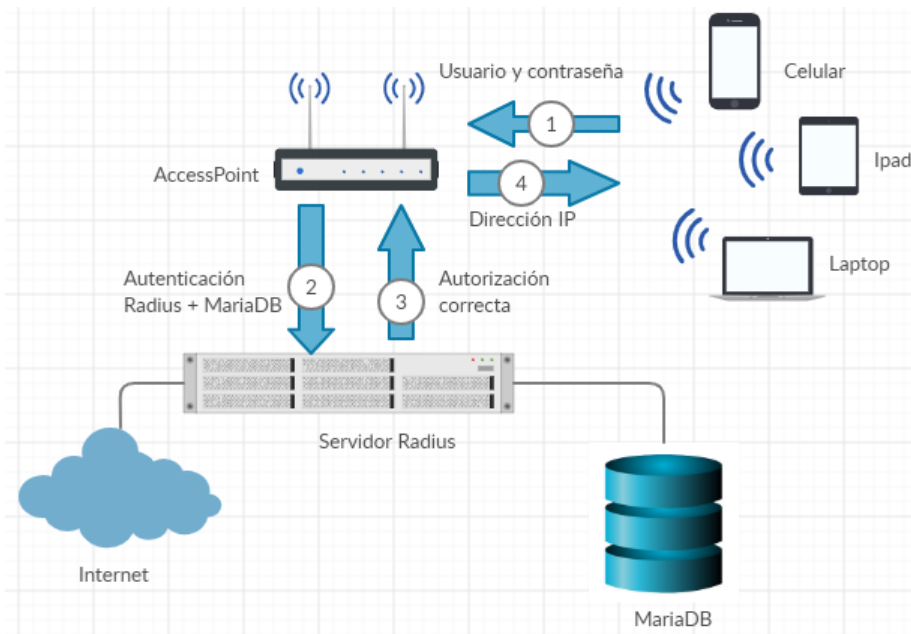
Lo anterior, fue solamente una descripción de los elementos que conforman el diseño de nuestro sistema, sin embargo, falta comprender como funcionarán todos estos elementos en conjunto. Para lo cual, a continuación, se muestra el diseño del entorno de red propuesto, el cual tendría el siguiente flujo de ejecución:

- 1) El usuario intenta conectarse a la red inalámbrica desde su dispositivo ingresando su usuario y contraseña
- 2) El Access Point recibe la petición y este se comunica con el servidor Radius para verificar si el usuario existe o no en la base de datos.
- 3) Si el usuario existe, el servidor Radius notifica al Access Point que ese usuario tiene permitido el acceso y uso de la red Inalámbrica.
- 4) El Access Point asigna una dirección IP al dispositivo que solicitó la conexión, entonces el usuario puede comenzar a utilizar la red.

---

<sup>50</sup> Imagen tomada de:

<https://www.tp-link.com/mx/home-networking/wifi-router/tl-wr840n/#specifications>



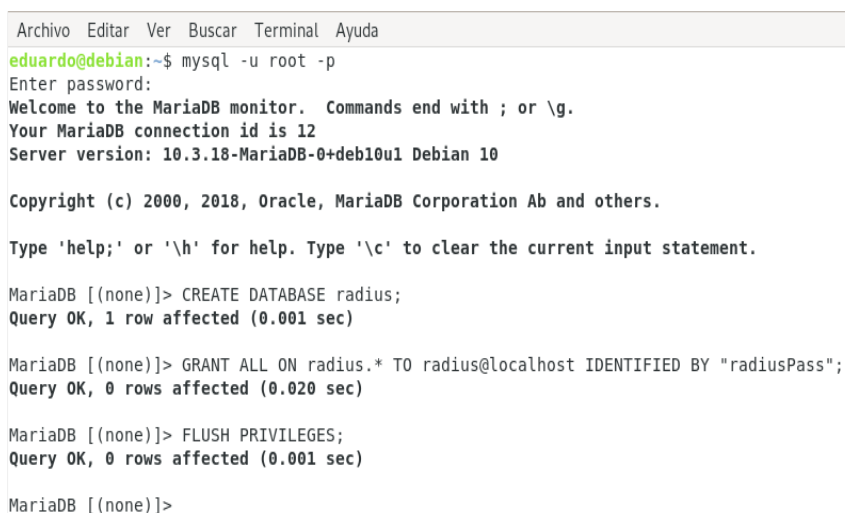
**Figura 38. Diagrama general del funcionamiento de red usando servidor Radius**  
Fuente: Imagen diseñada en internet<sup>51</sup>

<sup>51</sup> Imagen creada con:  
[app.creately.com](http://app.creately.com)

### 4.3 Implementación y configuración del servidor

Primero accedemos a MariaDB con el usuario root (siendo este el usuario con mayores privilegios dentro del sistema operativo) mediante el comando **mysql -u root -p**, donde **-u** indica el usuario (que lo escribimos a continuación) y **-p** indica el password (contraseña), y que en este caso no escribimos por razones de seguridad, pero posteriormente se nos solicitará. Después creamos una base de datos utilizando el comando **CREATE DATABASE radius** (donde radius será el nombre de nuestra base de datos). Posteriormente, crearemos un usuario con el comando **GRANT ALL ON radius.\* TO radius@localhost IDENTIFIED BY "radiusPass";**

El comando anterior nos indica que el usuario **radius** cuya contraseña es **radiusPass** tendrá acceso total a la base de datos **radius** y a todas sus tablas, eso se indica usando el **\***. Finalmente actualizamos la base de datos utilizando el comando **FLUSH PRIVILEGES**.



```
Archivo Editar Ver Buscar Terminal Ayuda
eduardo@debian:~$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.3.18-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE radius;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> GRANT ALL ON radius.* TO radius@localhost IDENTIFIED BY "radiusPass";
Query OK, 0 rows affected (0.020 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]>
```

**Figura 39. Creación de la Base de datos**  
Fuente: Elaboración propia de la Instalación del servidor <sup>52</sup>

Nos aseguramos que podamos acceder a la base datos 'radius' con el usuario creado, utilizando el comando **mysql -u radius -p**, damos **Enter** e introducimos la contraseña. Después mediante el comando **show databases** verificamos que la base de datos **radius** se haya creado exitosamente. Al último para poder utilizar la base de datos creada, tecleamos el comando **use radius**, el cual cambiará la base de datos en la cual trabajaremos.

<sup>52</sup> Implementación y configuración del Servidor Radius

```

eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
eduardo@debian:~$ mysql -u radius -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 16
Server version: 10.3.18-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database           |
+-----+
| information_schema |
| radius              |
+-----+
2 rows in set (0.001 sec)

MariaDB [(none)]> use radius;
Database changed
MariaDB [radius]>

```

**Figura 40. Verificación de la base de datos creada**

Para la instalación de FreeRadius, utilizaremos el siguiente comando:  
***apt -y install freeradius freeradius-mysql freeradius-utils***

```

Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# apt -y install freeradius freeradius-mysql freeradius-utils
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
ca-certificates-java collectd collectd-core default-jre-headless dmeventd
freeradius-common freeradius-config freetds-common intel-cmt-cat java-common
libc-ares2 libcollectdclient1 libconfuse-common libconfuse2 libct4 libdb11
libdevmapper-event1.02.1 libesmtp6 libfreeradius3 libftdi1-2 libganglia1
libgoogle-perftools4 libgss23 libgrpc++1 libgrpc6 libhiredis0.14 libi2c0
liblua5.1-0 liblvm2cmd2.03 libmemcached11 libmicrohttpd12 libmodbus5 libmosquitto1
libopenipmi0 liboping0 libow-3.2-3 libowcapi-3.2-3 libpq5 librabbitmq4 librdkafka1
libriemann-client0 librrd8 librte-acl18.11 librte-bbdev18.11
librte-bitratestats18.11 librte-bpf18.11 librte-cfgfile18.11 librte-cmdline18.11
librte-compressdev18.11 librte-cryptodev18.11 librte-distributor18.11
librte-eal18.11 librte-efd18.11 librte-ethdev18.11 librte-eventdev18.11
librte-flow-classify18.11 librte-gro18.11 librte-gso18.11 librte-hash18.11
librte-ip-frag18.11 librte-jobstats18.11 librte-kni18.11 librte-kvargs18.11
librte-latencystats18.11 librte-lpm18.11 librte-mbuf18.11 librte-member18.11
librte-mempool18.11 librte-meter18.11 librte-metrics18.11 librte-net18.11
librte-pci18.11 librte-pdump18.11 librte-pipeline18.11 librte-port18.11
librte-power18.11 librte-rawdev18.11 librte-reorder18.11 librte-ring18.11
librte-sched18.11 librte-security18.11 librte-table18.11 librte-telemetry18.11
librte-timer18.11 librte-vhost18.11 libtcmalloc-minimal4 libtokyocabinet9
libtokyotyran3 libupsclient4 libvarnishapi2 libvirt0 libxencall1
libxendevicemodel1 libxenevtchn1 libxenforeignmemory1 libxengnttab1 libxenmisc4.11
libxenstore3.0 libxentoolcore1 libxentoollog1 lvm2 make openjdk-11-jre-headless
owfs-common rrdtool thin-provisioning-tools

```

**Figura 41. Descarga e instalación de freeRadius**  
Fuente: Elaboración propia de la Instalación del servidor <sup>53</sup>

Iniciamos el servicio después de la instalación usando el comando:  
**systemctl enable --now freeradius.service**

```
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# systemctl enable --now freeradius.service
Synchronizing state of freeradius.service with SysV service script with /lib/systemd/sy
stemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable freeradius
Created symlink /etc/systemd/system/multi-user.target.wants/freeradius.service -> /lib/s
ystemd/system/freeradius.service.
root@debian:/home/eduardo# █
```

**Figura 42. Descarga e instalación de freeRadius**

Verificamos el estatus de FreeRadius mediante el comando **systemctl status freeradius**, y si el servicio se encuentra en ejecución, como resultado debemos obtener **active (running)**.

```
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# systemctl status freeradius
● freeradius.service - FreeRADIUS multi-protocol policy server
   Loaded: loaded (/lib/systemd/system/freeradius.service; enabled; vendor preset: enab
   Active: active (running) since Sat 2019-12-07 03:17:34 CST; 19min ago
     Docs: man:radiusd(8)
           man:radiusd.conf(5)
           http://wiki.freeradius.org/
           http://networkradius.com/doc/
   Main PID: 12366 (freeradius)
     Tasks: 6 (limit: 1138)
    Memory: 8.5M
    CGroup: /system.slice/freeradius.service
            └─12366 /usr/sbin/freeradius

dic 07 03:17:34 debian freeradius[12273]: [/etc/freeradius/3.0/mods-config/attr_filter/
dic 07 03:17:34 debian freeradius[12273]: rlm_cache (cache_eap): Driver rlm_cache_rbtre
dic 07 03:17:34 debian freeradius[12273]: rlm_detail (auth_log): 'User-Password' suppre
dic 07 03:17:34 debian freeradius[12273]: rlm_mschap (mschap): using internal authentic
dic 07 03:17:34 debian freeradius[12273]: Ignoring "sql" (see raddb/mods-available/REA
dic 07 03:17:34 debian freeradius[12273]: Ignoring "ldap" (see raddb/mods-available/REA
dic 07 03:17:34 debian freeradius[12273]: # Skipping contents of 'if' as it is always
dic 07 03:17:34 debian freeradius[12273]: radiusd: #### Skipping IP addresses and Ports
dic 07 03:17:34 debian freeradius[12273]: Configuration appears to be OK
dic 07 03:17:34 debian systemd[1]: Started FreeRADIUS multi-protocol policy server.
lines 1-23/23 (END)
```

**Figura 43. Verificación de estado de freeRadius**

Fuente: Elaboración propia de la Instalación del servidor <sup>54</sup>

Importamos el esquema de bases de datos de Radius para llenar la base de datos **radius** que creamos, mediante el comando **mysql -u root -p radius < /etc/freeradius/3.0/mods-config/sql/main/mysql/schema.sql**  
Damos **Enter** e introducimos la contraseña para permitir la importación.

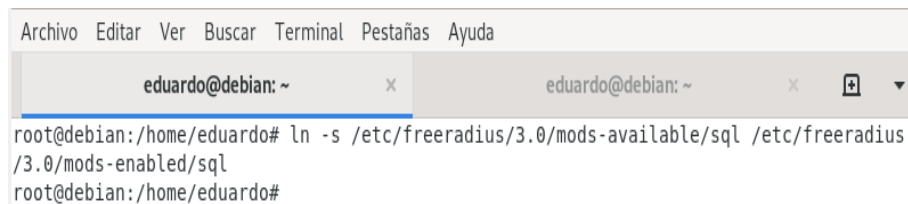
```
eduardo@debian: ~ x
eduardo@debian: ~ x [?] v
root@debian:/home/eduardo# mysql -u root -p radius < /etc/freeradius/3.0/mods-config/sql
main/mysql/schema.sql
Enter password:
root@debian:/home/eduardo# █
```

**Figura 44. Importación de esquema de BD.**

<sup>54</sup> Implementación y configuración del Servidor Radius

Para la configuración de radius primero crearemos un enlace simbólico para el módulo de SQL usando el comando:

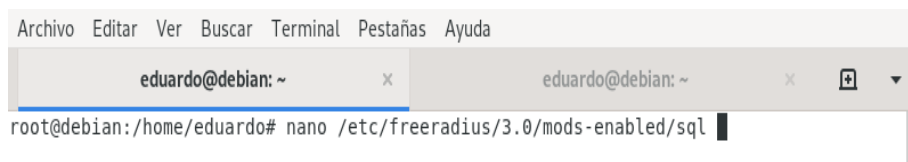
***ln -s /etc/freeradius/3.0/mods-available/sql /etc/freeradius/3.0/mods-enabled/sql***



```
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
eduardo@debian: ~ x
eduardo@debian: ~ x
root@debian:/home/eduardo# ln -s /etc/freeradius/3.0/mods-available/sql /etc/freeradius/3.0/mods-enabled/sql
root@debian:/home/eduardo#
```

**Figura 45. Creación de enlace simbólico.**

Configuramos el módulo de SQL cambiando los parámetros de conexión de la base de datos para ajustarlo a nuestras necesidades utilizando el comando **nano /etc/freeradius/3.0/mods-enabled/sql** el cual nos abrirá un editor de textos para realizar los cambios.



```
Archivo Editar Ver Buscar Terminal Pestañas Ayuda
eduardo@debian: ~ x
eduardo@debian: ~ x
root@debian:/home/eduardo# nano /etc/freeradius/3.0/mods-enabled/sql
```

**Figura 46. Comando para cambios de parámetros de conexión**

Fuente: Elaboración propia de la Instalación del servidor <sup>55</sup>

La sección de SQL debe parecerse a esto:

```
sql {
    # The sub-module to use to execute queries. This should match
    # the database you're attempting to connect to.
    #
    # * rlm_sql_mysql
    # * rlm_sql_mssql
    # * rlm_sql_oracle
    # * rlm_sql_postgresql
    # * rlm_sql_sqlite
    # * rlm_sql_null (log queries to disk)
    #
    driver = "rlm_sql_mysql"

    # If you're using rlm_sql_null, then it should be the type of
    # database the logged queries are going to be executed against.
    dialect = "mysql"

    # Connection info:
    #
    server = "localhost"
    port = 3306
    login = "radius"
    password = "radiusPass"

    # Database table configuration for everything except Oracle
    radius_db = "radius"
```

<sup>55</sup> Implementación y configuración del Servidor Radius

```

# Set to 'yes' to read radius clients from the database ('nas' table)
# Clients will ONLY be read on server startup.
read_clients = yes

# Table to keep radius client info
client_table = "nas"

```

**Figura 47. Configuración final de archivo**

Cambiamos los derechos del grupo utilizando el comando **chgrp -h freerad /etc/freeradius/3.0/mods-available/sql** También cambiamos el propietario utilizando el comando **chown -R freerad:freerad /etc/freeradius/3.0/mods-enabled/sql**

```

eduardo@debian: ~ x
eduardo@debian: ~ x
root@debian:/home/eduardo# chgrp -h freerad /etc/freeradius/3.0/mods-available/sql
root@debian:/home/eduardo# chown -R freerad:freerad /etc/freeradius/3.0/mods-enabled/sql
root@debian:/home/eduardo#

```

**Figura 48. Comandos para asignación de permisos**

Reiniciamos el servicio de Radius mediante el comando **systemctl restart freeradius**

```

root@debian:/home/eduardo# systemctl restart freeradius
root@debian:/home/eduardo#

```

**Figura 49. Comando para reinicio de servidor**  
Fuente: Elaboración propia de la Instalación del servidor <sup>56</sup>

Para verificar que FreeRadius puede autenticar en local, haremos la siguiente prueba. Tecleamos el comando **cd /etc/freeradius/3.0** para dirigirnos a la carpeta **3.0** y editamos el archivo users usando el comando **nano users**

```

eduardo@debian: ~ x
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# cd /etc/freeradius/3.0
root@debian:/etc/freeradius/3.0# nano users

```

**Figura 50. Configuración de archivo users**

Procedemos a descomentar la siguiente línea para realizar pruebas usando al usuario "steve" con contraseña "testing", guardamos los cambios utilizando la combinación de teclas **Ctrl + O** para guardar y **Ctrl + X** para salir.

<sup>56</sup> Implementación y configuración del Servidor Radius

```

eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
GNU nano 3.2 users

# This is a complete entry for "steve". Note that there is no Fall-Through
# entry so that no DEFAULT entry will be used, and the user will NOT
# get any attributes in addition to the ones listed here.
#
steve  Cleartext-Password := "testing"
#      Service-Type = Framed-User,
#      Framed-Protocol = PPP,
#      Framed-IP-Address = 172.16.3.33,
#      Framed-IP-Netmask = 255.255.255.0,
#      Framed-Routing = Broadcast-Listen,
#      Framed-Filter-Id = "std.ppp",
#      Framed-MTU = 1500,
#      Framed-Compression = Van-Jacobson-TCP-IP

#
# The canonical testing user which is in most of the
# examples.
#
#bob  Cleartext-Password := "hello"

```

**Figura 51. Creación de usuario en archivo users**

Reiniciamos una vez más el servidor usando el comando **systemctl restart freeradius**

```

eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# systemctl restart freeradius
root@debian:/home/eduardo#

```

**Figura 52. Reinicio del servidor freeRadius**

Comprobamos que nuestro servidor sea capaz de autenticar el usuario **Steve** usando el comando **radtest steve testing 127.0.0.1 1812 testing 123**, si nada ha fallado, se nos mostrará un mensaje que incluye el texto **Access-Accept**.

```

eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# systemctl restart freeradius
root@debian:/home/eduardo# radtest steve testing 127.0.0.1 1812 testing123
Sent Access-Request Id 211 from 0.0.0.0:59644 to 127.0.0.1:1812 length 75
  User-Name = "steve"
  User-Password = "testing"
  NAS-IP-Address = 127.0.0.1
  NAS-Port = 1812
  Message-Authenticator = 0x00
  Cleartext-Password = "testing"
Received Access-Accept Id 211 from 127.0.0.1:1812 to 127.0.0.1:59644 length 20
root@debian:/home/eduardo#

```

**Figura 53. Prueba de autenticación de usuario en local**  
Fuente: Elaboración propia de la Instalación del servidor <sup>57</sup>

<sup>57</sup> Implementación y configuración del Servidor Radius



## Instalación y configuración de DaloRadius

DaloRadius es utilizado para administrar el servidor Radius desde una interfaz web, siendo esta una configuración opcional.

Primero descargamos el archivo DaloRadius desde Github utilizando el comando **wget** <https://github.com/lirantal/daloradius/archive/master.zip>

```
root@debian:/home/eduardo# wget https://github.com/lirantal/daloradius/archive/master.zip
--2020-09-14 20:16:19-- https://github.com/lirantal/daloradius/archive/master.zip
Resolviendo github.com (github.com)... 140.82.113.4
Conectando con github.com (github.com)[140.82.113.4]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Localización: https://codeload.github.com/lirantal/daloradius/zip/master [siguiendo]
--2020-09-14 20:16:19-- https://codeload.github.com/lirantal/daloradius/zip/master
Resolviendo codeload.github.com (codeload.github.com)... 140.82.114.9
Conectando con codeload.github.com (codeload.github.com)[140.82.114.9]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/zip]
Grabando a: "master.zip"

master.zip           [      <=>      ]  5.20M  489KB/s   en 6.6s

2020-09-14 20:16:26 (803 KB/s) - "master.zip" guardado [5453837]

root@debian:/home/eduardo#
```

**Figura 54. Descarga de archivo de daloRadius**

Desempaquetamos el archivo con el comando **unzip master.zip**

```
root@debian:/home/eduardo# unzip master.zip
Archive:  master.zip
909ec0ac3271f58e36847ef4d176b5c17416a43e
  creating:  daloradius-master/
  extracting: daloradius-master/.gitignore
  inflating: daloradius-master/.htaccess
  inflating: daloradius-master/.htpasswd
  inflating: daloradius-master/ChangeLog
  inflating: daloradius-master/Dockerfile
  inflating: daloradius-master/Dockerfile-freeradius
  inflating: daloradius-master/FAQS
  inflating: daloradius-master/INSTALL
  inflating: daloradius-master/INSTALL.opensuse
  inflating: daloradius-master/INSTALL.quick
  inflating: daloradius-master/INSTALL.win
```

**Figura 55. Comando para desempaquetar archivo**

Fuente: Elaboración propia de la Instalación del servidor <sup>58</sup>

Cambiamos el nombre del archivo de **daloradius-master/** a **daloradius** utilizando el comando **mv daloradius-master/ daloradius** y nos movemos al directorio utilizando el comando **cd daloradius/** para configurarlo.

```
root@debian:/home/eduardo# mv daloradius-master/ daloradius
root@debian:/home/eduardo# cd daloradius/
root@debian:/home/eduardo/daloradius#
```

**Figura 56. Modificación de nombre y acceso a directorio**

<sup>58</sup> Implementación y configuración del Servidor Radius

Importamos las tablas de mysql a daloRadius mediante el comando **mysql -u root -p radius < contrib/db/fr2-mysql-daloradius-and-freeradius.sql**

```
root@debian:/home/eduardo/daloradius# mysql -u root -p radius < contrib/db/fr2-mysql-
daloradius-and-freeradius.sql
Enter password:
root@debian:/home/eduardo/daloradius# mysql -u root -p radius < contrib/db/mysql-dalo
radius.sql
Enter password:
root@debian:/home/eduardo/daloradius#
```

**Figura 57. Importación de tablas a daloRadius**

Movemos el folder **daloradius/** a la ruta **/var/www/html/** con el comando **mv daloradius/ /var/www/html/**

```
root@debian:/home/eduardo/daloradius# cd ..
root@debian:/home/eduardo# mv daloradius/ /var/www/html/
root@debian:/home/eduardo#
```

**Figura 58. Cambio de ruta del directorio daloRadius**

Ahora cambiamos los permisos para el folder con el comando **chown -R www-data:www-data /var/www/html/daloradius/** y establecemos los permisos correctos para el archivo de configuración de daloRadius con el comando **chmod 664 /var/www/html/daloradius/library/daloradius.conf.php**

```
root@debian:/home/eduardo# chown -R www-data:www-data /var/www/html/daloradius/
root@debian:/home/eduardo# chmod 664 /var/www/html/daloradius/library/daloradius.conf.
php
root@debian:/home/eduardo#
```

**Figura 59. Asignación de permisos al directorio**

Se deberá configurar el archivo **daloradius.conf.php** para ajustar la información de la base datos de MariaDB, para ello tecleamos el comando **nano /var/www/html/daloradius/library/daloradius.conf.php**

```
root@debian:/home/eduardo# nano /var/www/html/daloradius/library/daloradius.conf.php
```

**Figura 60. Comando para configurar conexión a base de datos**

Fuente: Elaboración propia de la Instalación del servidor <sup>59</sup>

Escribimos la contraseña, usuario y el nombre de la base de datos para poder establecer la conexión.

```
$configValues ['CONFIG_DB_HOST'] = 'localhost';
$configValues ['CONFIG_DB_PORT'] = '3306';
$configValues ['CONFIG_DB_USER'] = 'radius';
$configValues ['CONFIG_DB_PASS'] = 'radiusPass';
$configValues ['CONFIG_DB_NAME'] = 'radius';
```

**Figura 61. Asignación de parámetros para conexión**

<sup>59</sup> Implementación y configuración del Servidor Radius

Para asegurarnos que todo funcione correctamente, reiniciamos de nueva cuenta el servidor con el comando **systemctl restart freeradius.service apache2**

```
root@debian:/home/eduardo# systemctl restart freeradius.service apache2
root@debian:/home/eduardo#
```

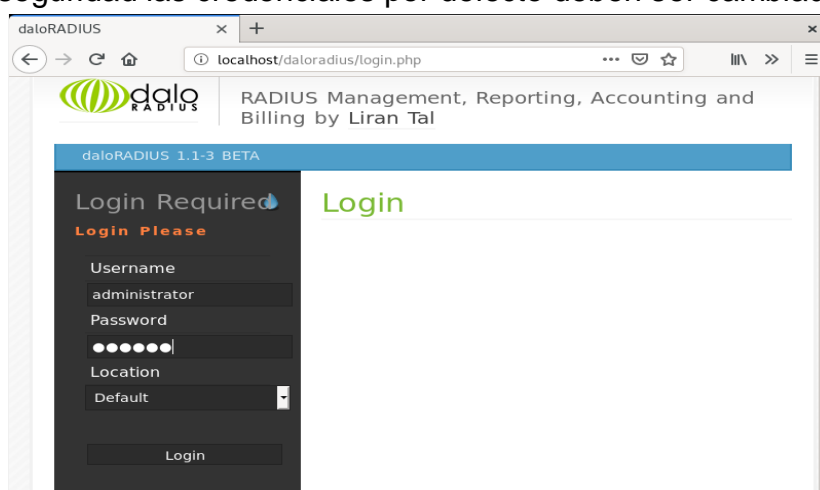
**Figura 62. Reinicio de servicios**

Accedemos a la interfaz Web de daloRadius utilizando la dirección **http://localhost/daloradius** Los detalles de inicio de sesión son los siguientes:

Nombre de usuario: administrator

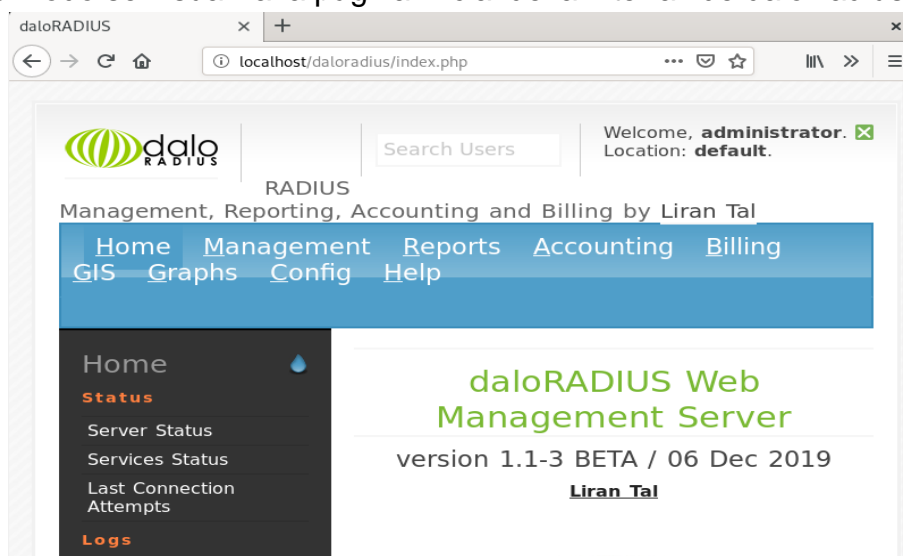
Contraseña: radius

Nota: Por seguridad las credenciales por defecto deben ser cambiadas.



**Figura 63. Interfaz inicial de herramienta daloRadius**

De este modo se visualiza la página inicial de la interfaz de daloRadius



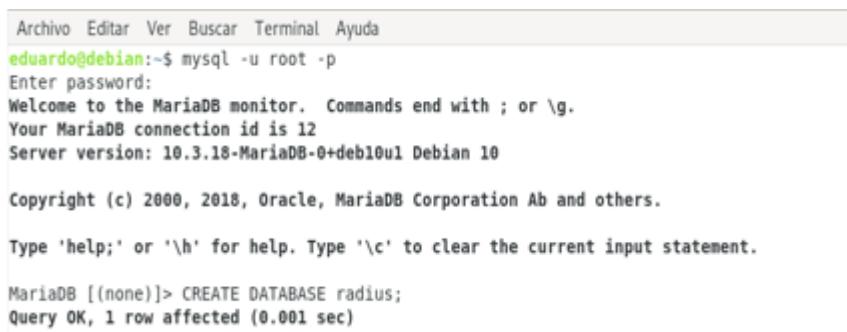
**Figura 64. Página inicial de acceso a daloRadius**  
Fuente: Elaboración propia de la Instalación del servidor <sup>60</sup>

<sup>60</sup> Implementación y configuración del Servidor Radius

#### 4.4 Creación de la base de datos

Primero crearemos una base de datos para hacer uso del sistema, la cual llevará por nombre **radius** con el comando **CREATE DATABASE radius;**

Nota: Estos pasos fueron creados cuando se hizo la instalación del servidor en pasos anteriores.



```
Archivo Editar Ver Buscar Terminal Ayuda
eduardo@debian:~$ mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.3.18-MariaDB-0+deb10u1 Debian 10

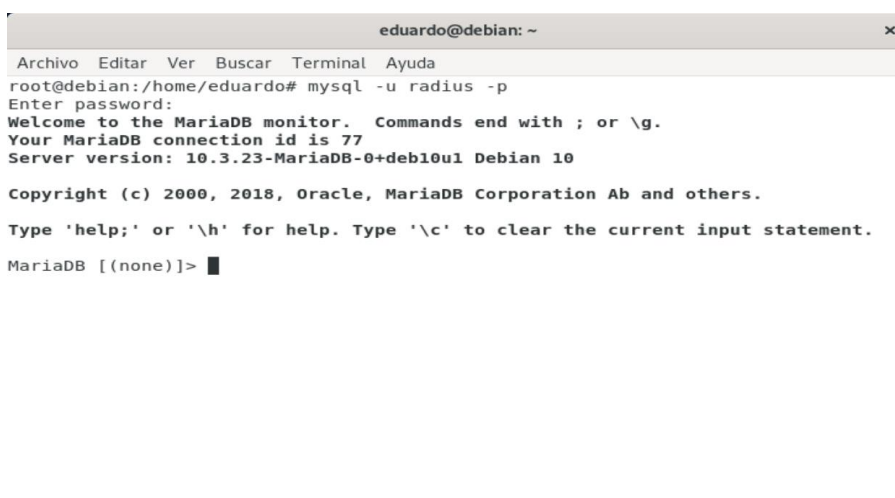
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE radius;
Query OK, 1 row affected (0.001 sec)
```

**Figura 65. Creación de la base de datos**  
Fuente: Elaboración propia<sup>61</sup>

Para acceder al Sistema Gestor de Bases de Datos MariaDB, colocamos el siguiente comando **mysql -u radius -p** accediendo desde el usuario root.



```
eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# mysql -u radius -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 77
Server version: 10.3.23-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> █
```

**Figura 66. Acceso al Sistema Gestor de Bases de Datos**  
Fuente: Elaboración propia<sup>62</sup>

<sup>61</sup> Captura de pantalla de terminal en Sistema Operativo Debian

<sup>62</sup> Captura de pantalla de terminal en Sistema Operativo Debian

Para poder acceder a la base de datos **radius** hacemos uso del siguiente comando **use radius;**

```
MariaDB [(none)]> use radius;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [radius]>
```

**Figura 67. Acceso a la base de datos**  
Fuente: Elaboración propia<sup>63</sup>

Dentro del sistema, utilizaremos distintas tablas

- **radposauth**  
Esta tabla guardará el registro de los usuarios cada vez que sean autenticados por el servidor
- **radcheck**  
Esta tabla almacenará los usuarios y contraseñas de todos los alumnos que se vayan registrando en el sistema.
- **radgroupcheck**  
Esta tabla definirá a que grupo pertenece cada usuario, y con ello saber si el usuario está habilitado o deshabilitado.

Nota: Estas tablas son exportadas directamente cuando se realiza la configuración del servidor freeRadius.

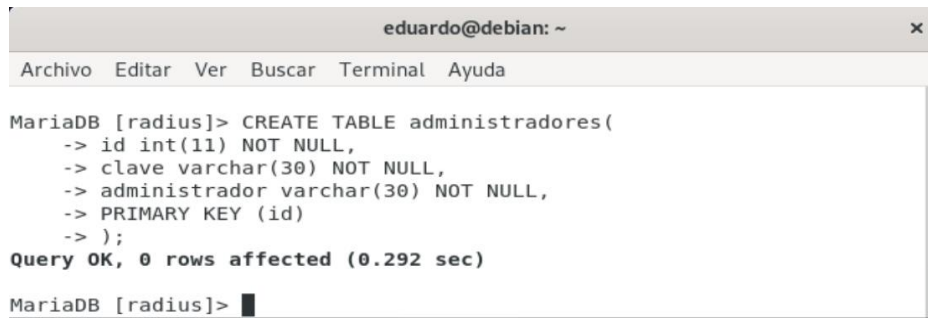
Adicional a estas tablas, crearemos las tablas **administradores** y **alumnos**. La primera para asignar los permisos a las personas que estarán al cargo del sistema y la segunda para guardar los datos de los alumnos que se vayan registrando en el sistema.

La creación de la tabla **administradores** se realizó utilizando el comando:

```
CREATE TABLE adminstradores{
id int(11) NOT NULL,
clave varchar(30) NOT NULL,
administrador varchar(30) NOT NULL,
PRIMARY KEY (id)
};
```

---

<sup>63</sup> Captura de pantalla de terminal en Sistema Operativo Debian



```
eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda

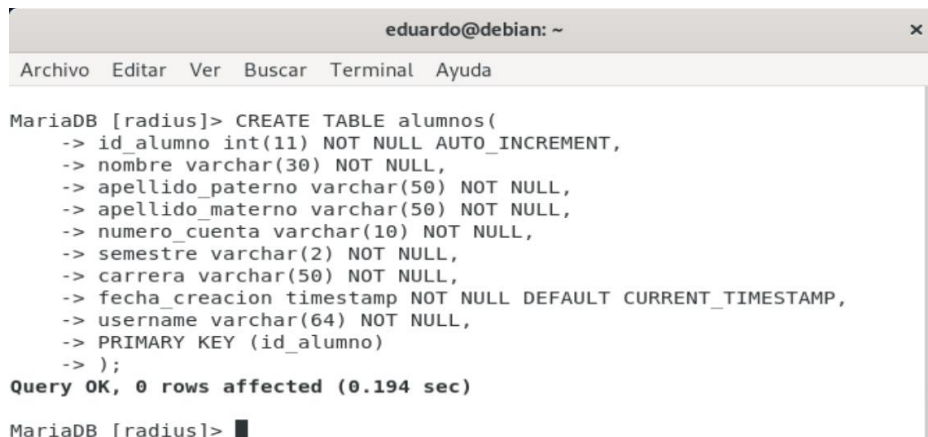
MariaDB [radius]> CREATE TABLE administradores(
-> id int(11) NOT NULL,
-> clave varchar(30) NOT NULL,
-> administrador varchar(30) NOT NULL,
-> PRIMARY KEY (id)
-> );
Query OK, 0 rows affected (0.292 sec)

MariaDB [radius]> █
```

**Figura 68. Creación de tabla administradores**  
Fuente: Elaboración propia<sup>64</sup>

La creación de la tabla **alumnos** se realizó utilizando el comando:

```
CREATE TABLE alumnos(  
id_alumno int(11) NOT NULL AUTO_INCREMENT,  
nombre varchar(30) NOT NULL,  
apellido_paterno varchar(50) NOT NULL,  
apellido_materno varchar(50) NOT NULL,  
numero_cuenta varchar(10) NOT NULL,  
semestre varchar(2) NOT NULL,  
carrera varchar(50) NOT NULL,  
fecha_creacion timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
username varchar(64) NOT NULL,  
PRIMARY KEY (id_alumno)  
);
```



```
eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda

MariaDB [radius]> CREATE TABLE alumnos(
-> id_alumno int(11) NOT NULL AUTO_INCREMENT,
-> nombre varchar(30) NOT NULL,
-> apellido_paterno varchar(50) NOT NULL,
-> apellido_materno varchar(50) NOT NULL,
-> numero_cuenta varchar(10) NOT NULL,
-> semestre varchar(2) NOT NULL,
-> carrera varchar(50) NOT NULL,
-> fecha_creacion timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
-> username varchar(64) NOT NULL,
-> PRIMARY KEY (id_alumno)
-> );
Query OK, 0 rows affected (0.194 sec)

MariaDB [radius]> █
```

**Figura 69. Creación de tabla alumnos**  
Fuente: Elaboración propia<sup>65</sup>

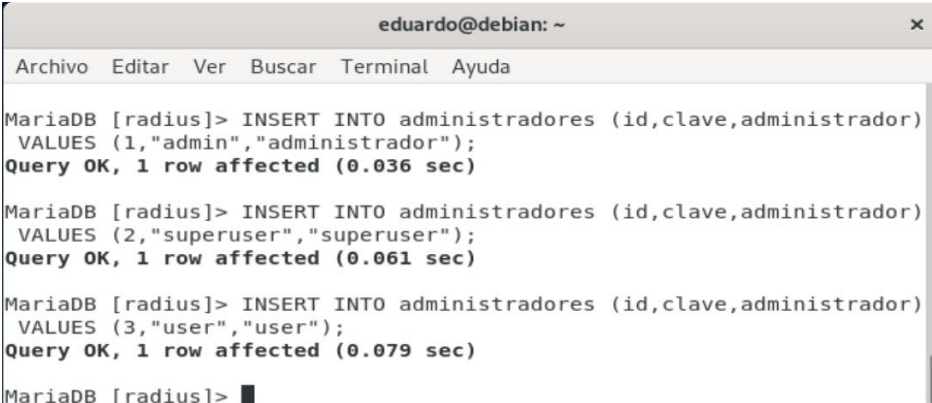
<sup>64</sup> Captura de pantalla de terminal en Sistema Operativo Debian

<sup>65</sup> Captura de pantalla de terminal en Sistema Operativo Debian

## 4.5 Población de la base de datos

A modo de ejemplo, agregaremos algunos datos a las tablas **administradores** desde la terminal usando el comando:

```
INSERT INTO administradores (id,clave,administrador) VALUES  
(1,"admin","administrador");  
INSERT INTO administradores (id,clave,administrador) VALUES  
(2,"superuser","superuser");  
INSERT INTO administradores (id,clave,administrador) VALUES  
(3,"user","user");
```



```
eduardo@debian: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
MariaDB [radius]> INSERT INTO administradores (id,clave,administrador)  
VALUES (1,"admin","administrador");  
Query OK, 1 row affected (0.036 sec)  
MariaDB [radius]> INSERT INTO administradores (id,clave,administrador)  
VALUES (2,"superuser","superuser");  
Query OK, 1 row affected (0.061 sec)  
MariaDB [radius]> INSERT INTO administradores (id,clave,administrador)  
VALUES (3,"user","user");  
Query OK, 1 row affected (0.079 sec)  
MariaDB [radius]> █
```

**Figura 70. Inserción de datos en tabla administradores**  
Fuente: Elaboración propia<sup>66</sup>

A continuación, tecleamos el comando **select \* from administradores** para verificar que la información se agregó correctamente.



```
eduardo@debian: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
MariaDB [radius]>  
MariaDB [radius]> select * from administradores;  
+-----+-----+-----+  
| id | clave      | administrador |  
+-----+-----+-----+  
|  1 | admin     | administrador |  
|  2 | superuser | superuser     |  
|  3 | user      | user          |  
+-----+-----+-----+  
3 rows in set (0.001 sec)  
MariaDB [radius]>
```

**Figura 71. Comprobación de inserción de datos exitosa**  
Fuente: Elaboración propia<sup>67</sup>

Procedemos a agregar algunos datos a las tablas **alumnos** utilizando los comandos:

<sup>66</sup> Captura de pantalla de terminal en Sistema Operativo Debian

<sup>67</sup> Captura de pantalla de terminal en Sistema Operativo Debian

### **INSERT INTO alumnos**

**(nombre,apellido\_paterno,apellido\_materno,numero\_cuenta,semestre,carrera,username) VALUES**  
**(“CARLOS”,“POLO”,“CASTELLANOS”,“310045690”,“5”,“INGENIERIA CIVIL”,“c4r21!”);**

### **INSERT INTO alumnos**

**(nombre,apellido\_paterno,apellido\_materno,numero\_cuenta,semestre,carrera,username) VALUES**  
**(“KAREN”,“DIAZ”,“MEJIA”,“310090172”,“8”,“INGENIERIA EN TELECOMUNICACIONES”,“K\_29N”);**

```
MariaDB [radius]> INSERT INTO alumnos (nombre,apellido_paterno,apellido_materno,numero_cuenta,semestre,carrera,username) VALUES ("CARLOS","POLO","CASTELLANOS", "310045690", "5", "INGENIERIA CIVIL", "c4r21!");
Query OK, 1 row affected (0.037 sec)

MariaDB [radius]> INSERT INTO alumnos (nombre,apellido_paterno,apellido_materno,numero_cuenta,semestre,carrera,username) VALUES ("KAREN","DIAZ","MEJIA", "310090172", "8", "INGENIERIA EN TELECOMUNICACIONES", "K_29N");
Query OK, 1 row affected (0.060 sec)

MariaDB [radius]> █
```

**Figura 72. Inserción de datos en tabla alumnos**

Fuente: Elaboración propia<sup>68</sup>

Mostramos los datos utilizando el comando **SELECT \* FROM alumnos\G** para verificar que se agregaron correctamente.

```
eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
MariaDB [radius]> SELECT * FROM alumnos\G
***** 1. row *****
  id_alumno: 1
    nombre: CARLOS
apellido_paterno: POLO
apellido_materno: CASTELLANOS
  numero_cuenta: 310045690
    semestre: 5
    carrera: INGENIERIA CIVIL
  fecha_creacion: 2021-04-24 01:30:49
    username: c4r21!
***** 2. row *****
  id_alumno: 2
    nombre: KAREN
apellido_paterno: DIAZ
apellido_materno: MEJIA
  numero_cuenta: 310090172
    semestre: 8
    carrera: INGENIERIA EN TELECOMUNICACIONES
  fecha_creacion: 2021-04-24 01:32:37
    username: K_29N
2 rows in set (0.000 sec)

MariaDB [radius]>
```

**Figura 73. Comprobación de inserción de datos exitosa**

Fuente: Elaboración propia<sup>69</sup>

<sup>68</sup> Captura de pantalla de terminal en Sistema Operativo Debian

<sup>69</sup> Captura de pantalla de terminal en Sistema Operativo Debian



# **CAPÍTULO 5**

# **DOCUMENTACIÓN**

# **Y RESULTADOS**

## 5.1 Creación de la Documentación del Sistema.

Para la creación de la documentación de este sistema, se llevó a cabo la realización de varios documentos, los cuales se pueden revisar en el apartado de los apéndices, los cuales van desde la descarga, instalación y configuración en el caso del software, los cuales son la base de datos, y el servidor. Por otro lado, está la configuración del hardware, es decir, el Access Point. Por lo tanto los manuales creados son los siguientes:

### 1. Manual de configuración de Base de datos

Este manual se puede encontrar en el apéndice número 1, en este manual se encuentran en primera instancia los comandos para realizar la instalación mediante la consola de comandos de nuestro sistema operativo, posteriormente se menciona cómo se puede llevar a cabo la configuración, principalmente por temas de seguridad, luego se muestran los comando necesarios para la creación de una base de datos, como acceder a ella y una vez estando dentro, cómo poder crear tablas, y por último también se observan comandos de cómo poder ingresar información en las tablas creadas, lo que usualmente se le llama “poblar la base de datos”.

### 2. Manual de configuración de Access Point

Este manual se puede encontrar en el apéndice número 2, en este manual encontraremos al principio cómo poder acceder a este equipo, luego cómo poder cambiar los parámetros tales como la dirección IP, una contraseña secreta y un puerto, ya que todo esto nos servirá para su interacción con nuestro servidor,

### 3. Manual de configuración del Servidor

Este manual se puede encontrar en el apéndice número 3, en este manual vamos a encontrar los comandos necesarios para poder descargar e instalar este servidor mediante la consola de comandos de nuestro sistema operativo Debian. Posteriormente encontraremos los comandos necesarios para la configuración del servidor, entre los cuales están, como exportar tablas necesarias para el funcionamiento, como asignar los permisos necesarios y como realizar el “registro” de nuestro Access Point dentro del servidor para la correcta interacción entre ambos.

## **5.2 Manual de uso del sistema.**

Debido a la extensión de este apartado, este manual se puede encontrar en el apéndice número 5, en este manual vamos a encontrar cómo utilizar el sistema mediante el uso de una interfaz gráfica creada con los lenguajes php y html principalmente, esta interfaz será la encargada de ayudarnos a operar el sistema para realizar el alta, modificación, deshabilitación y eliminación de los usuarios, sin tener que acceder directamente a la terminal del sistema operativo para realizar estas acciones, de esta manera, será mucho más simple para la persona encargada del sistema llevar el monitoreo y control de los usuarios sin tener que ser una persona experta en sistemas. Además, se muestran ejemplos de cómo poder realizar cada una de estas acciones y los posibles resultados que podemos obtener.

## **5.3 Resultados del Sistema**

Hasta el momento al ser un proyecto que no se encuentra aún implementado en la Facultad, y que no está siendo utilizada por los alumnos, no se puede afirmar que se logró reducir los costos en cuanto a su implementación al no tener un punto de comparación, por ejemplo, comparar el costo real de mantenimiento de las salas de cómputo con el mantenimiento de este sistema o la velocidad de transmisión y la capacidad soportada de usuarios de esta red comparada con la RIU. Sin embargo, se espera que al mantener toda esta filosofía de implementar este sistema utilizando en su mayoría software libre, el único costo mayor que se pudiera dar sería en la adquisición de un buen Access Point y del equipo que alojará al servidor, sin embargo, más adelante se planea realizar un estudio para verificar el correcto funcionamiento y la correcta aceptación por parte de los alumnos de la Facultad. Por otro lado, con base en las pruebas realizadas de manera local, pudimos observar el correcto funcionamiento del sistema, tanto desde el alta (Véase Figura 74, 75 y 76), modificación (Véase Figura 77 y 78), deshabilitación (Véase Figura 79 y 80) y eliminación de usuarios (Véase Figura 81 y 82), hasta la autenticación de los mismos mediante el Access Point (Véase Figura 83 y 84).

Con base en las reglas definidas al principio de este proyecto, de igual manera se creó una página que servirá de interfaz gráfica y que se comunicará con nuestra base de datos, para el registro de usuarios, facilitando así el manejo del sistema para usuarios que no lograrían utilizarlo si no cuentan con conocimientos amplios en sistemas operativos o manejo de sistemas mediante comandos de terminal.

Se llevó a cabo la creación de la documentación necesaria para facilitar la descarga, instalación y configuración de los componentes del sistema, así como para el uso del sistema a la hora de estar en operación. Por último podemos concluir que se espera haber cumplido los objetivos, desde el uso de software

libre y de bajo costo para así reducir costos, hasta la lógica entre el sistema y la base de datos, la cual nos ayudará a mantener un mejor control desde el momento en que un usuario se da de alta y durante todo el momento en que este utilice la red de Internet, que como se mencionó, al menos de manera local todo fue exitoso sin embargo necesitamos de otros factores para valorarlo.

A continuación se describen brevemente los resultados de las pruebas realizadas de manera local.

Para el alta de usuario será necesario navegar hasta la página **agregar.php** para posteriormente rellenar el formulario con los datos solicitados como se puede visualizar en las figuras 74 y 75.

Internet UNAM - Mozilla Firefox

Internet UNAM

localhost/tesis/agregar.php

### Agregar usuarios

Nombre: DANIEL

Apellido Paterno: GÓMEZ

Apellido Materno: DELGADO

Número de cuenta: 302891290

Semestre: 6

**Figura 74. Ingreso de datos para alta de usuario**

Internet UNAM - Mozilla Firefox

localhost/tesis/agregar.php

Número de cuenta: 302891290

Semestre: 6

Carrera: Ingeniería en Telecomunicaciones

Usuario: danielG21

Contraseña: .....

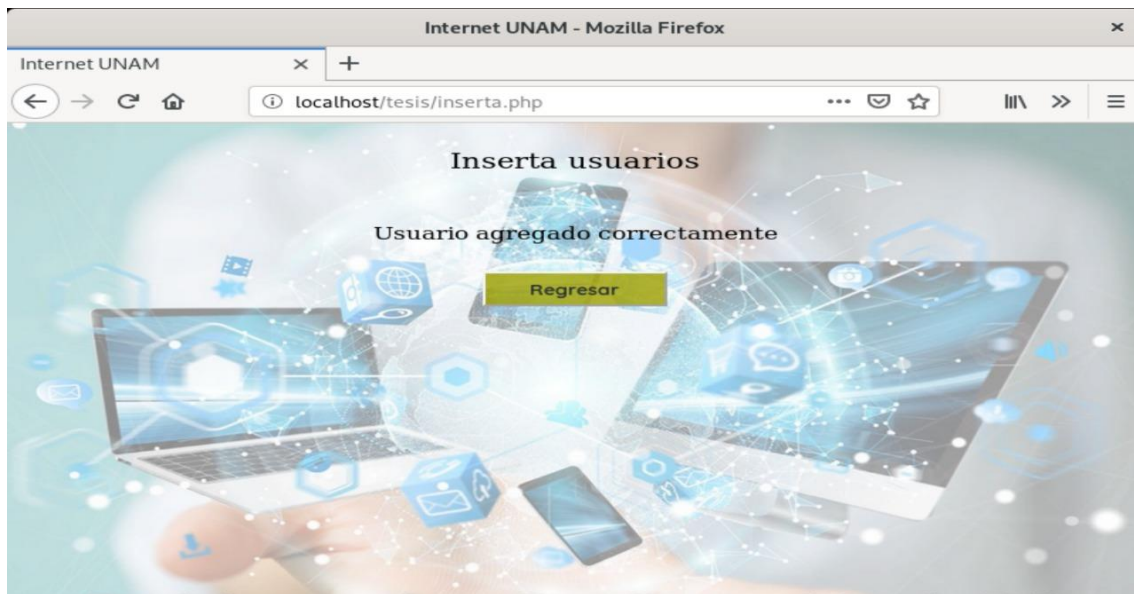
Agregar Usuario

Regresar

Copyright© 2021 - Sistema creado por Eduardo de la Cruz Reyes - Todos los derechos reservados

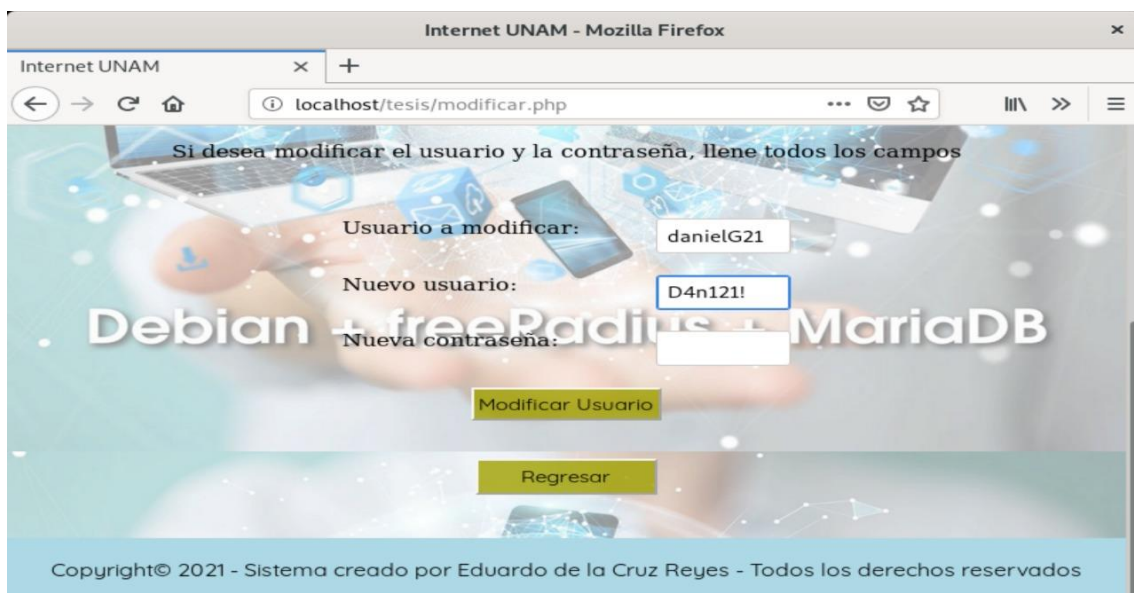
**Figura 75. Ingreso de datos para alta de usuario**

Una vez completado el formulario al dar clic en el botón **Agregar Usuario**, si no existe ningún error, se mostrará un mensaje de confirmación que menciona **Usuario agregado correctamente** como lo podemos apreciar en la Figura 76.



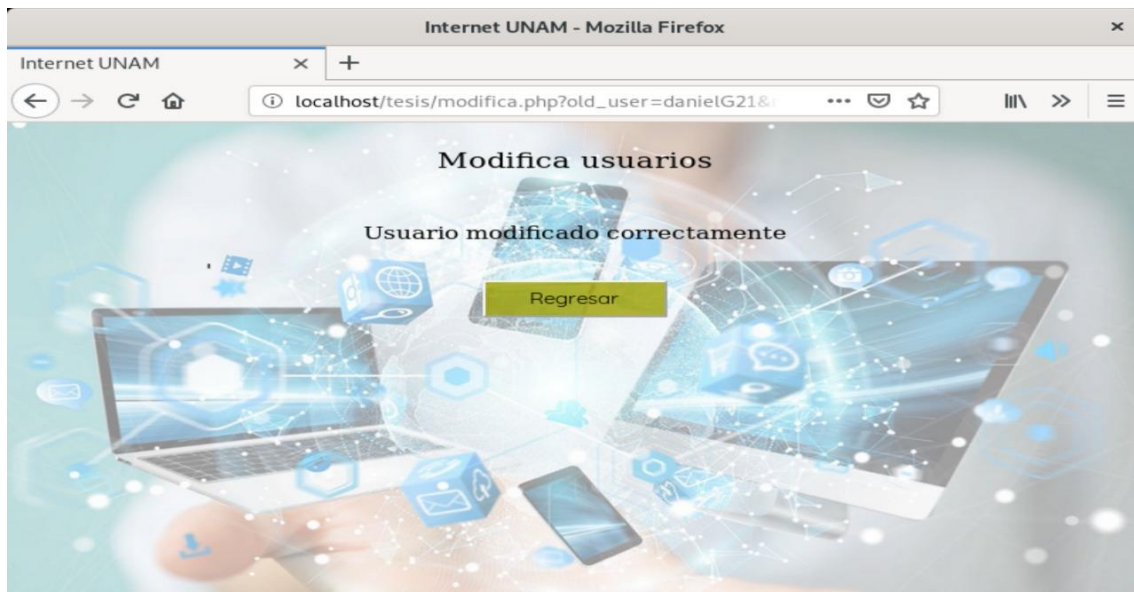
**Figura 76. Alta de usuario exitosa.**

Para la modificación de usuarios será necesario navegar hasta la página **modificar.php** para posteriormente rellenar el formulario con los datos solicitados como se puede visualizar en la figura 77.



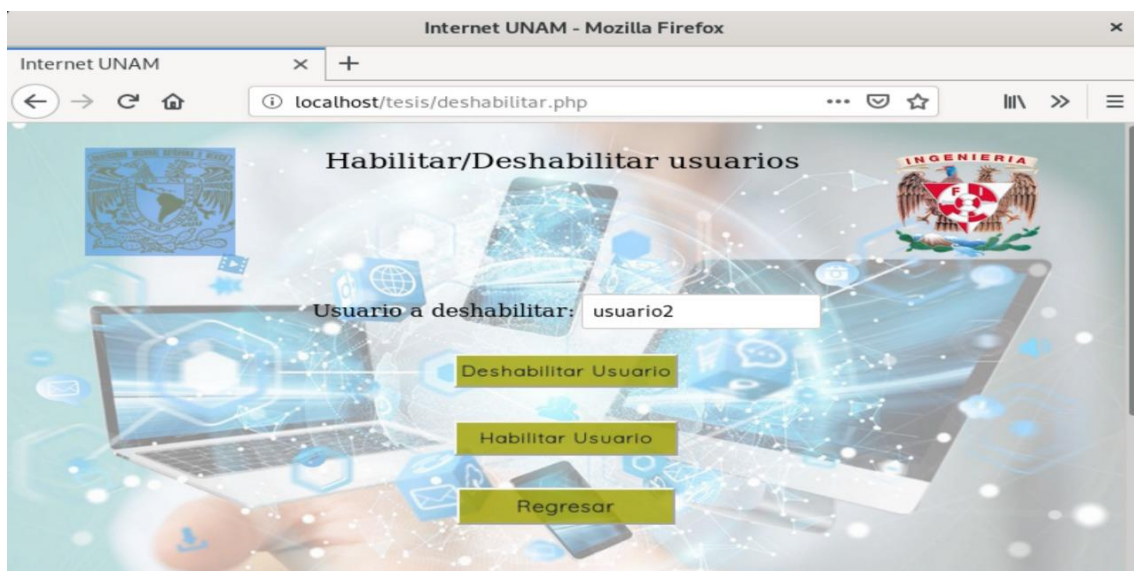
**Figura 77. Ingreso de datos para modificación de usuarios.**

Al dar clic en el botón **Modificar Usuario**, si todo ha salido bien se nos mostrará otra página con el mensaje **Usuario modificado correctamente** como lo podemos apreciar en la Figura 78.



**Figura 78. Modificación de usuario exitoso.**

Para la deshabilitación de usuarios será necesario navegar hasta la página **deshabilitar.php** para posteriormente rellenar el formulario con los datos solicitados como se puede visualizar en la figura 79.



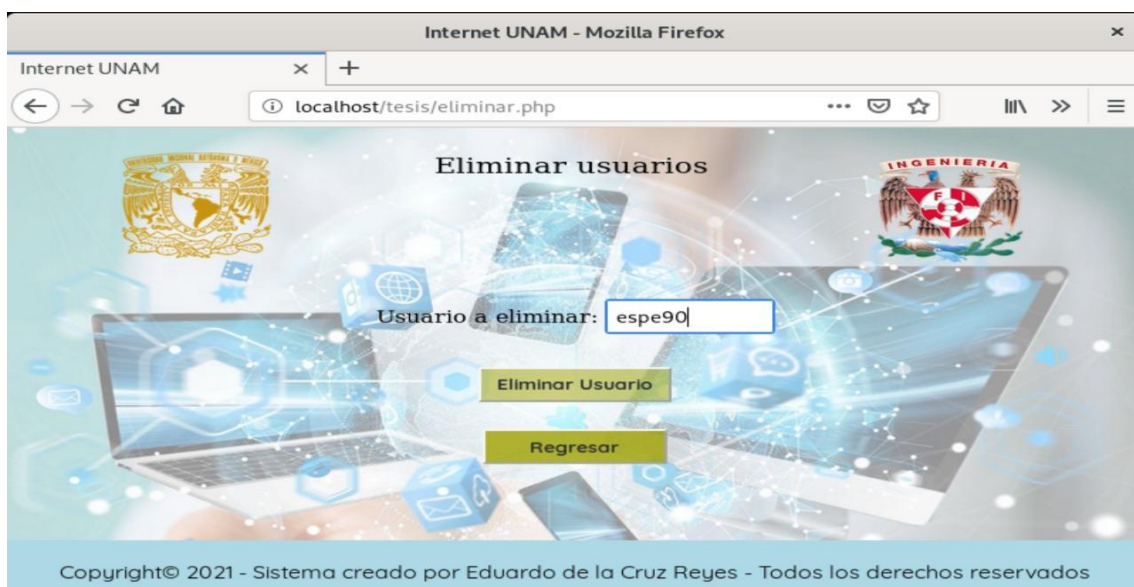
**Figura 79. Ingreso de usuario para deshabilitación.**

Al dar clic en el botón **Deshabilitar Usuario**, si todo ha salido bien se nos mostrará otra página con el mensaje **Usuario deshabilitado correctamente** como lo podemos apreciar en la Figura 80.



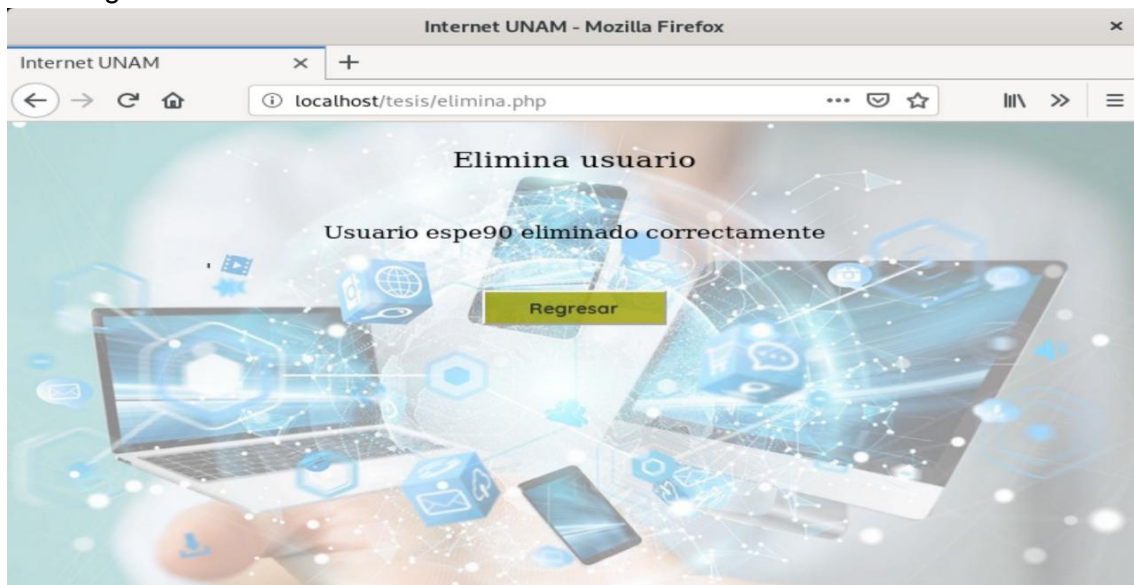
**Figura 80. Deshabilitación de usuario exitoso.**

Para la eliminación de usuarios será necesario navegar hasta la página **eliminar.php** para posteriormente rellenar el formulario con los datos solicitados como se puede visualizar en la figura 81.



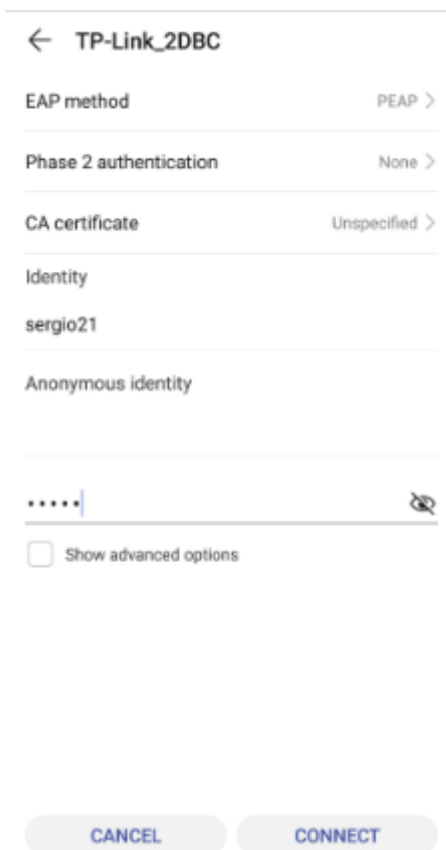
**Figura 81. Ingreso de usuario para eliminación.**

Al dar clic en el botón **Eliminar Usuario**, si todo ha salido bien se nos mostrará otra página con el mensaje **Usuario eliminado correctamente** como lo podemos apreciar en la Figura 82.



**Figura 82. Eliminación de usuario exitosa.**

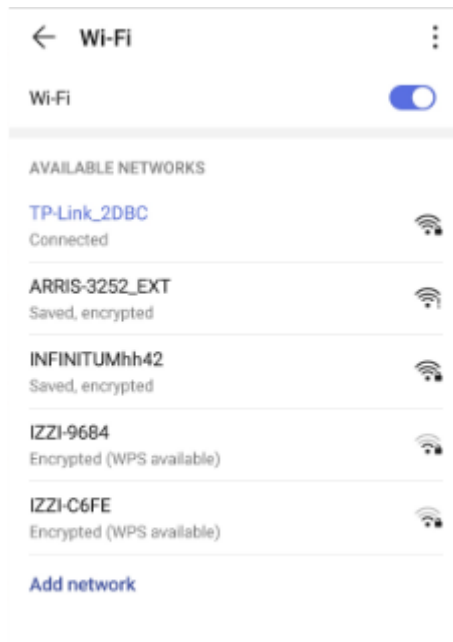
Para la autenticación de usuarios mediante el Access Point, en este caso se utilizó un dispositivo celular con sistema operativo Android. Ingresamos el usuario y la contraseña de algún usuario dado de alta como se observa en la figura 83 y daremos clic en el botón conectar.



**Figura 83. Ingreso de datos para conexión a la red mediante Access Point.**



Si todo ha salido bien, nos habremos conectado al Access Point (que en este caso lleva por nombre TP-Link\_2DBC) y tendremos acceso para poder navegar por Internet.



**Figura 84. Conexión exitosa de usuario al Access Point**

# CONCLUSIONES

Como resultado de la realización de este proyecto de Tesis puedo concluir varios aspectos, utilicé gran parte de los conceptos aprendidos durante mi estancia en la facultad, tales como, fundamentos en sistemas operativos, bases de datos, redes, y administración de proyectos de software. Tales conceptos fueron de gran utilidad para enfocarme en su mayoría, en el aprendizaje e implementación de algo que era desconocido para mí, un servidor Radius. Durante mi trayectoria en la Facultad de Ingeniería, al cursar materias enfocadas al tema de redes y seguridad, conocía conceptos tales como la autenticación, autorización y confidencialidad, cual es el funcionamiento de un router, un switch o un Access Point, y maneras de mantener protegida una red mediante WAP, WEP, WPKS. Finalmente todos estos conceptos fueron implementados en manera conjunta para la realización del servidor, un servidor del cual había escuchado hablar pero que jamás había implementado de manera personal.

El desarrollo de esta tesis fue en momentos laborioso y en momentos complicados, fue un proceso que conllevó a leer mucha documentación, el cómo instalar y configurar el servidor, primero de manera individual y luego en conjunto con el Access Point, conceptos que eran completamente desconocidos para mí. Debo destacar que en lo personal fue un gran reto para mí, debido a que aunque no empecé desde cero con los conceptos mencionados, sí tuve que comenzar desde cero con el aprendizaje y la instalación del servidor. Fue un proceso que duró muchos meses, meses en los cuales hubo momentos de frustración, momentos en los cuales tuve que repetir una y otra vez los pasos. Si bien la curva de aprendizaje fue lenta, al final entendí el sistema y cómo funcionaba mientras me iba involucrando más, mientras iban pasando los meses.

Y como todo en esta vida, existen momentos en los cuales uno piensa en desistir por uno u otro motivo cuando las cosas se ponen complicadas, cuando uno siente que ya no puede, habiendo llegado a ese punto, puedo decir que recordé toda mi estancia en la Facultad y me propuse a mí mismo, no darme por vencido hasta haber culminado el trabajo. Entonces puedo decir que me siento muy orgulloso de haber salido adelante frente a todas las complicaciones que tuve en el desarrollo de esta tesis, me demostré que puedo ser una persona capaz de aprender de manera autónoma, y que como la palabra lo dice, un Ingeniero debe ser capaz de usar su ingenio para la resolución del problema que se le presente, así que yo tomé todos los recursos a mi alcance, desde experiencia propia hasta recursos de libros e Internet para poder sobresalir. Definitivamente, si tal vez, alguna vez llegue a dudar de mis capacidades para aprender algo, con esto no me queda duda que soy un ser humano e Ingeniero capaz de resolver problemáticas, y seguir aprendiendo cosas nuevas.

Por otro lado, la creación de este proyecto de tesis trae consigo, el beneficio de brindar ese apoyo extra a los estudiantes dentro de la Facultad de Ingeniería, ya que este sistema en conjunto con las salas de cómputo y la RIU brindarán a los

alumnos otra opción de conectividad para realizar sus actividades, dándole a los alumnos la posibilidad de elegir la que mejor se adapte a sus necesidades, sin contar que por el lado del sistema creado debido a que, al ser un servicio que únicamente estará limitado a alumnos, el número de usuarios dentro de la red se reduce drásticamente, esto porque personas ajenas a la Facultad, es decir, que no sean alumnos de la Facultad, no podrán ingresar con otras credenciales, caso contrario con la RIU, que sin importar en qué lugar esté, con mis credenciales puedo acceder. Asimismo, al tener un control más estricto del número de sesiones diarias que inicia cada usuario, evitamos que se presten usuarios y contraseñas a personas no dadas de alta en el sistema, ya que esto conllevaría a un bloqueo temporal de la cuenta, para lo cual, el alumno deberá asistir a solicitar el desbloqueo del mismo, junto con la advertencia de que el realizar esa práctica nuevamente, conllevaría a la expulsión definitiva del sistema.

Por lo tanto, como beneficio principal para la comunidad, la creación de este sistema es que los alumnos contarán con su propia red a nivel Facultad, no necesitarán acudir tampoco a la DGTIC, ya que el proceso de alta de usuarios también será directamente desde la Facultad, agilizando el proceso y la desidia de los alumnos de recorrer esa distancia para ir a tramitar sus accesos.

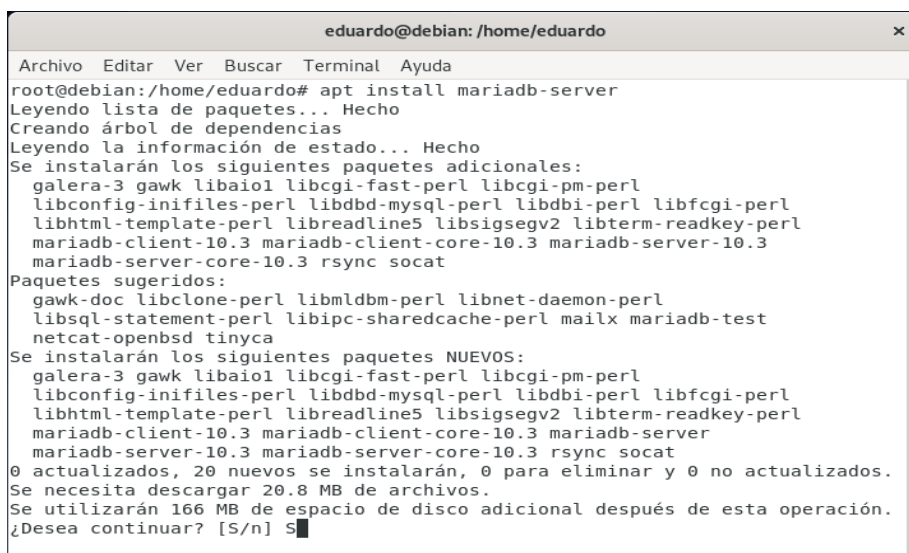
# APÉNDICES

# **APÉNDICE 1. INSTALACIÓN DE MariaDB**

MariaDB es una bifurcación del famoso MySQL. Como resultado, MySQL es por defecto MariaDB cuando la instalación se realiza. Es decir, incluso si utilizas el paquete de instalación de MySQL, mysql-server instalará MariaDB. Por lo tanto, es una buena idea instalar MariaDB usando su paquete de instalación, mariadb-server usando el siguiente comando:

***apt install mariadb-server***

A continuación se nos indicará el tamaño que ocupará la descarga, para lo cual debemos colocar la letra **S** cuando aparezca el mensaje **¿Desea continuar? [S/n]**, para indicar que sí aceptamos y presionar **Enter**, para empezar con la descarga e instalación.



```
eduardo@debian: /home/eduardo
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# apt install mariadb-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
galera-3 gawk libaio1 libcgi-fast-perl libcgi-pm-perl
libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libfcgi-perl
libhtml-template-perl libreadline5 libsigsegv2 libterm-readkey-perl
mariadb-client-10.3 mariadb-client-core-10.3 mariadb-server-10.3
mariadb-server-core-10.3 rsync socat
Paquetes sugeridos:
gawk-doc libclone-perl libmldbm-perl libnet-daemon-perl
libsql-statement-perl libipc-sharedcache-perl mailx mariadb-test
netcat-openbsd tinyca
Se instalarán los siguientes paquetes NUEVOS:
galera-3 gawk libaio1 libcgi-fast-perl libcgi-pm-perl
libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libfcgi-perl
libhtml-template-perl libreadline5 libsigsegv2 libterm-readkey-perl
mariadb-client-10.3 mariadb-client-core-10.3 mariadb-server
mariadb-server-10.3 mariadb-server-core-10.3 rsync socat
0 actualizados, 20 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 20.8 MB de archivos.
Se utilizarán 166 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
```

**Figura 85. Descarga e instalación de la base de datos MariaDB**  
Fuente: Elaboración propia de la Instalación del servidor <sup>70</sup>

Así como con Apache, MariaDB se inicia y se habilita para ejecutarse en el arranque del sistema después de que la instalación se realiza. Por defecto, MariaDB se envía con un script de seguridad simple que puede ser ejecutado para eliminar usuarios de base de datos anónimos, deshabilitar inicio de sesión remoto, eliminar bases de datos de pruebas en un esfuerzo para ejecutar la seguridad de la base datos preliminar.

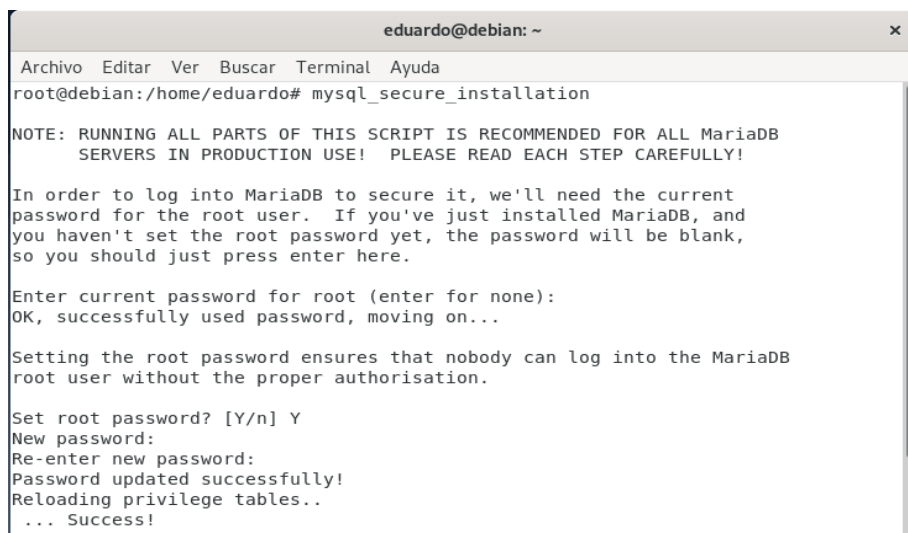
Por lo tanto, se debe ejecutar el siguiente script:

***mysql\_secure\_installation***

Cuando el script se ejecuta, aparece el enunciado **Enter current password for root (enter for none)**, el cual nos indica que debemos ingresar la contraseña para el usuario root o dar **Enter** si no existe alguna. Por defecto, al momento de la instalación, no existe ninguna, así que damos **Enter**. A continuación aparece el mensaje **Set root password? [Y/n]**, el cual nos pregunta si deseamos

<sup>70</sup> Implementación y configuración del Servidor Radius

establecer la contraseña para el usuario root, colocamos la letra **Y** para indicar que si deseamos establecer la contraseña y en **New password** colocamos la contraseña, posteriormente en **Re-enter new password** ingresamos nuevamente la contraseña, si ambas coinciden, obtenemos el mensaje **Password updated successfully**.



```
eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@debian:/home/eduardo# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

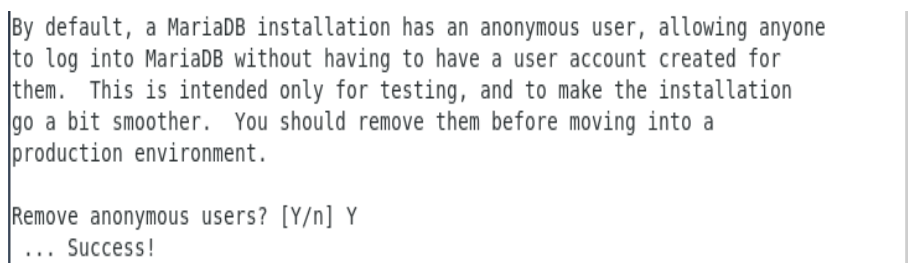
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!
```

**Figura 86. Modificación de datos por defecto de MariaDB**  
Fuente: Elaboración propia de la Instalación del servidor <sup>71</sup>

Por defecto, MariaDB se envía con usuarios anónimos que pueden acceder a la base de datos sin requerir una cuenta de usuario, sin embargo, esto plantea un riesgo de seguridad. Dentro de la configuración nos aparece el mensaje **Remove anonymous users? [Y/n]**, esto, para preguntarnos si deseamos eliminar los usuarios anónimos, colocamos la letra **Y** para confirmar que si deseamos eliminarlos y damos **Enter**, si todo sale bien obtenemos el mensaje **Success!**



```
By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
... Success!
```

**Figura 87. Eliminación de usuarios anónimos.**

El siguiente mensaje en aparecer será **Disallow root login remotely? [Y/n]**, esto nos pregunta si deseamos deshabilitar el inicio de sesión de manera remota, de tal manera que el usuario root pueda acceder solamente a la base de datos de manera local. Colocamos la letra **Y** para confirmar y damos **Enter**, si todo sale bien obtenemos el mensaje **Success!**

<sup>71</sup> Implementación y configuración del Servidor Radius



```
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
... Success!
```

**Figura 88. Desactivación de acceso remoto**

MariaDB incluye una base de datos de prueba, se nos preguntará si deseamos eliminarla con el mensaje **Remove test database and access to it? [Y/n]**. Colocamos la letra **Y** para confirmar y damos **Enter**, si todo sale bien obtenemos el mensaje **Success!**

```
By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!
```

**Figura 89. Eliminación de base de datos de prueba.**

Fuente: Elaboración propia de la Instalación del servidor <sup>72</sup>

Por último, se nos pregunta si deseamos recargar las tablas de privilegios ahora para guardar los cambios hechos, mediante el mensaje **Reload privilege tables now? [Y/n]**. Colocamos la letra **Y** para confirmar y damos **Enter**, si todo sale bien obtenemos el mensaje **Success!**

```
Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] Y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
root@debian:/home/eduardo#
```

**Figura 90. Configuración completa de la base de datos**

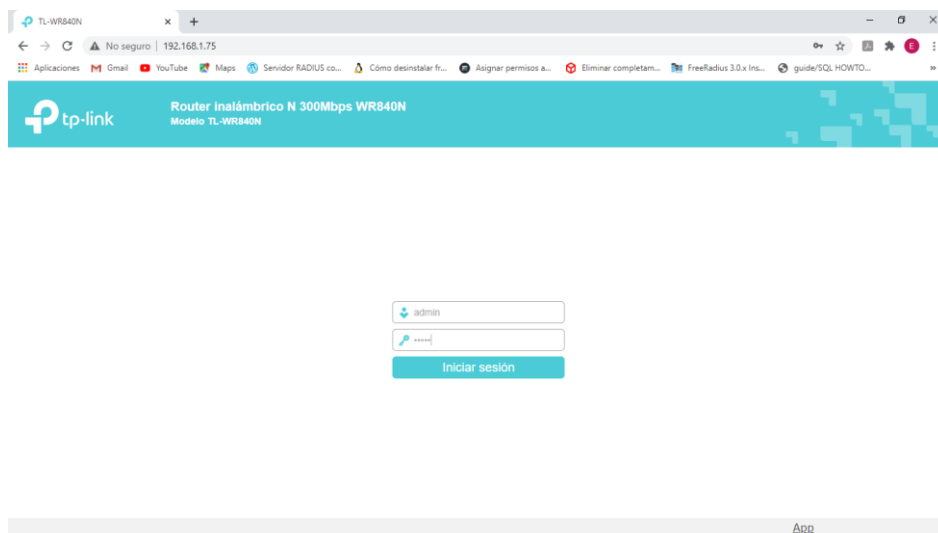
Con esto hemos configurado MariaDB correctamente en Debian.

---

<sup>72</sup> Implementación y configuración del Servidor Radius

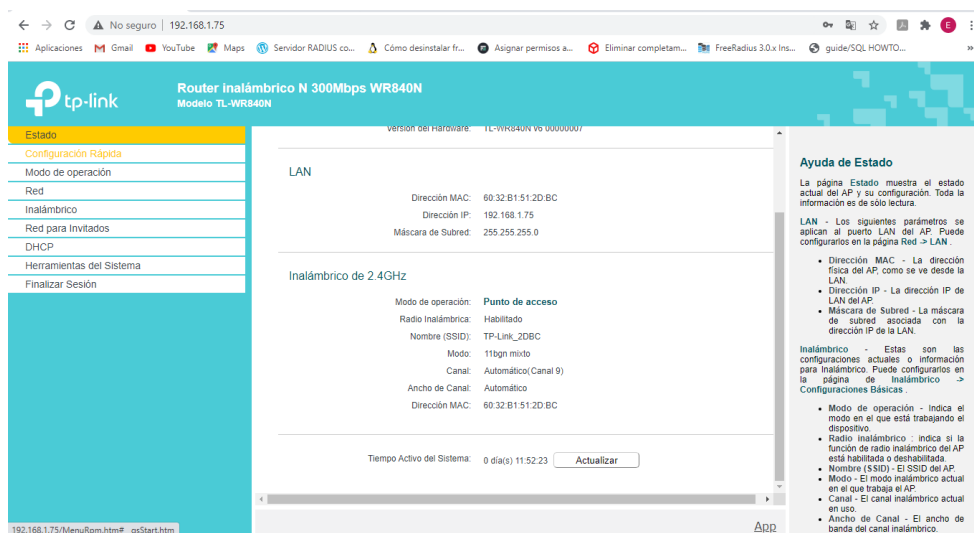
# **APÉNDICE 2. CONFIGURACIÓN DEL ACCESS POINT**

Como primer paso, nos conectaremos a nuestro Access Point ingresando mediante la IP que indique el fabricante, (usualmente esto lo podemos encontrar en alguna parte del equipo).



**Figura 91. Inicio de sesión del Access Point**  
Fuente: Elaboración propia de la Instalación del servidor <sup>73</sup>

A continuación, dentro de la configuración colocamos la IP estática 192.168.1.75, que será la que agregaremos a nuestro servidor para que ambos puedan comunicarse



**Figura 92. Configuración de la IP estática para usarse en el Access Point**

En el apartado **Inalámbrico** seleccionamos la opción **WPA/WPA2 - Empresarial**, en donde agregaremos los datos de nuestro servidor, como lo son la IP, en nuestro servidor colocamos la IP estática 192.168.1.100, y la contraseña del Servidor RADIUS para que al conectarnos a nuestro Access Point, esté se

<sup>73</sup> Implementación y configuración del Servidor Radius

comunique con el servidor mediante el puerto 1812 y realice la autenticación de los usuarios. Damos clic en el botón guardar y de esta manera el Access Point se reiniciará, con esto ya lo tenemos configurado para que nuestro servidor lo reconozca.

Router inalámbrico N 300Mbps WR840N  
Modelo TL-WR840N

Estado

Configuración Rápida

Modo de operación

Red

**Inalámbrico**

- Configuraciones Básicas
- WPS
- Seguridad Inalámbrica
- Filtrado MAC Inalámbrico
- Inalámbrico Avanzado
- Estadísticas Inalámbricas
- Monitor de rendimiento

Red para Invitados

DHCP

Herramientas del Sistema

Finalizar Sesión

WPA/WPA2 - Empresarial

Versión: WPA2

Encriptación: AES

IP del Servidor RADIUS: 192.168.1.100

Puerto del Servidor RADIUS: 1812 (1-5535, 0 representa el puerto predeterminado 1812)

Contraseña del Servidor RADIUS: claveSecreta

Período de Actualización Clave del Grupo: 0

WEP

Tipo de Autenticación: Sistema Abiert

Formato de la Clave de WEP: Hexadecimal

Clave Seleccionada:	Clave de WEP	Tipo de Clave
Clave 1: <input type="radio"/>	<input type="text"/>	Deshabilitado
Clave 2: <input type="radio"/>	<input type="text"/>	Deshabilitado
Clave 3: <input type="radio"/>	<input type="text"/>	Deshabilitado
Clave 4: <input type="radio"/>	<input type="text"/>	Deshabilitado

Guardar

**Ayuda de Seguridad Inalámbrica**

Puede seleccionar una de las siguientes opciones de seguridad:

- Desactivar seguridad inalámbrica - la función de seguridad inalámbrica se puede habilitar o deshabilitar. Si está desactivada, las estaciones inalámbricas podrán conectar el dispositivo sin encriptación. Se recomienda firmemente que elija una de las siguientes opciones para habilitar la seguridad.
- WPA/WPA2 Personal - Seleccionar WPA basándose en la contraseña pre-compartida.
- WPA/WPA2 Empresarial - Seleccionar WPA basándose en el Servidor Radius.
- WEP - Seleccionar Seguridad WEP 802.11.

Cada opción de seguridad tiene sus propias configuraciones como se describe a continuación.

WPA/WPA2 - Personal Versión - Puede seleccionar una de las siguientes versiones.

WEP - Puede seleccionar una de las siguientes versiones.

**Figura 93. Configuración de datos del servidor freeRadius en el Access Point**

# **APÉNDICE 3. CONFIGURACIÓN DE SERVIDOR FREERADIUS**

Primero nos dirigimos a la carpeta /etc/network/interfaces y colocamos la IP estática que deseemos, la cual utilizaremos en el Access Point para que ambos se comuniquen.

```
root@debian:~# cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

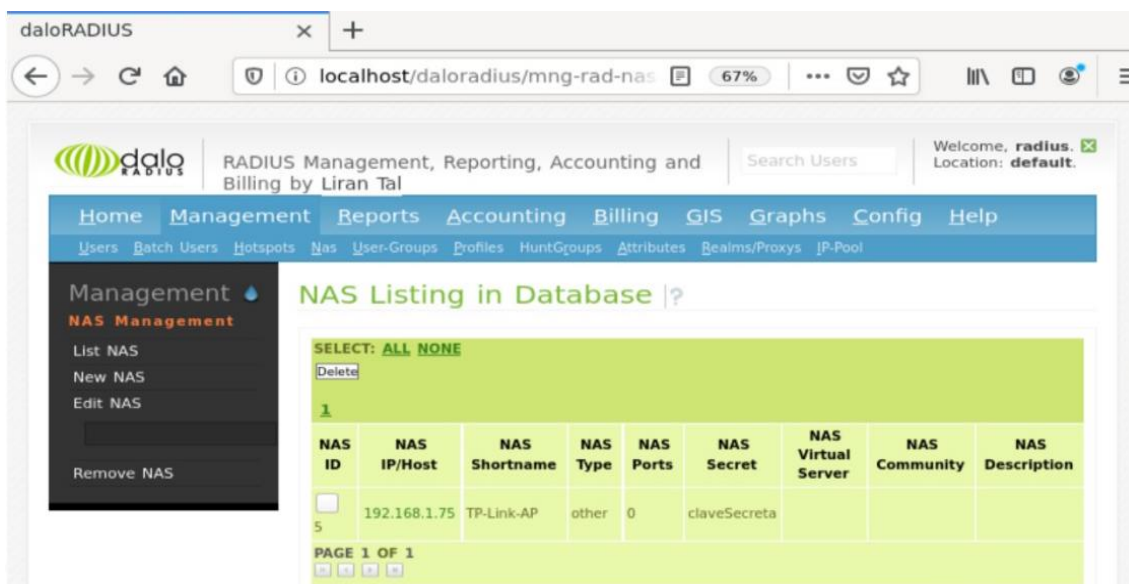
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
#allow-hotplug enp0s3
#iface enp0s3 inet dhcp

#Static IP address
auto enp0s3
iface enp0s3 inet static
    address 192.168.1.100
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
    gateway 192.168.1.1
```

**Figura 94. Asignación de IP estática en el servidor freeRadius**

Posteriormente nos dirigimos a la dirección donde instalamos daloRadius e iniciamos sesión. Navegamos por la pestaña de **Management** y en la sección **NAS** agregamos la IP que colocamos a nuestro Access Point, así como un nombre y una clave secreta que será la misma que colocamos en el Access Point para que ambos se reconozcan.

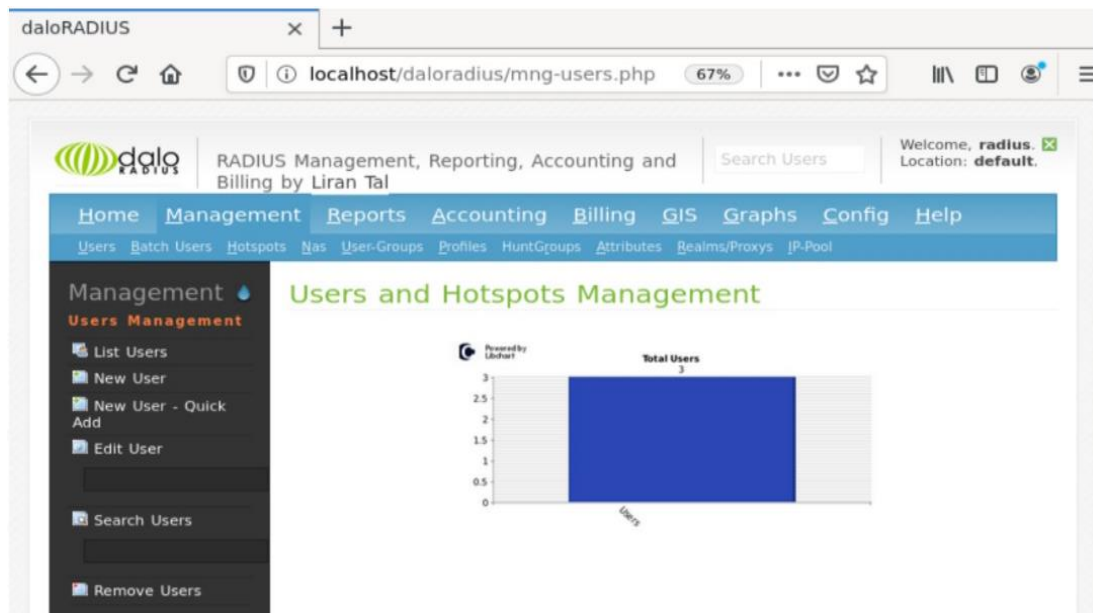


The screenshot shows the daloRADIUS web interface. The browser address bar displays 'localhost/daloradius/mng-rad-nas'. The page title is 'NAS Listing in Database'. A table with 9 columns is shown, containing one row of data for a NAS device.

NAS ID	NAS IP/Host	NAS Shortname	NAS Type	NAS Ports	NAS Secret	NAS Virtual Server	NAS Community	NAS Description
5	192.168.1.75	TP-Link-AP	other	0	claveSecreta			

**Figura 95. Llenado de tabla NAS con datos del Access Point**

A continuación dentro de la misma pestaña de **Management** ahora nos dirigimos a la sección **Users** y damos clic en **New User**



**Figura 96. Gráfica de la sección de usuarios**

Una vez dentro, agregamos un usuario y su contraseña, así como un grupo al que podría o no pertenecer (en este caso se elige uno) y damos clic en **apply**.

The screenshot shows the 'New User' form in the daloRADIUS web interface. The browser address bar indicates the URL is localhost/daloradius/mng-new.php. The page title is 'New User'. The form is titled 'New User' and has tabs for 'Account Info', 'User Info', 'Billing Info', and 'Attributes'. The 'Account Info' tab is selected. The form contains the following fields:

- Username: radius
- Password: radius
- Password Type: Cleartext-Password
- Group: usuarios

An 'Apply' button is located at the bottom of the form.

**Figura 97. Creación de un nuevo usuario**

Nos dirigimos a la sección **List Users** y comprobamos que el usuario se agregó correctamente. Estos serán los usuarios que autenticará freeRadius al conectarnos al Access Point.

The screenshot shows the daloRADIUS web interface. The main content area is titled "Users Listing" and features a table of users. The table has the following data:

ID	Name	Username	Password	Groups
<input type="checkbox"/> 2		user1	user1	
<input type="checkbox"/> 3		sebas	sebas	
<input type="checkbox"/> 12		usuario	usuario	usuarios
<input type="checkbox"/> 13		radius	radius	usuarios

Below the table, it indicates "PAGE 1 OF 1". The interface also includes a sidebar with "Management" options and a top navigation bar with "Home", "Management", "Reports", "Accounting", "Billing", "GIS", "Graphs", "Config", and "Help".

**Figura 98. Comprobación del usuario agregado**



# **APÉNDICE 4. COMANDOS BÁSICOS EN BASES DE DATOS**

Crear una base de datos

```
CREATE DATABASE nombre_bd;
```

Crear o reemplazar una base de datos

```
CREATE OR REPLACE DATABASE nombre_bd;
```

Comprueba que no exista la base de datos, si no existe, la crea

```
CREATE DATABASE IF NOT EXISTS nombre_bd;
```

Elimina una base de datos (sin comprobación)

```
DROP DATABASE nombre_bd;
```

Elimina una base de datos si existe (con comprobación)

```
DROP DATABASE IF EXISTS nombre_bd;
```

Crear una tabla dentro de una base de datos sin valores

```
CREATE TABLE nombre_tabla;
```

Crear una tabla dentro de una base de datos con valores

```
CREATE TABLE nombre_tabla ([campo] [tipo], [campo] [tipo]);
```

Crear o reemplazar una tabla

```
CREATE OR REPLACE TABLE nombre_tabla;
```

Eliminar una tabla (sin comprobación)

```
DROP TABLE nombre_tabla;
```

Eliminar una tabla si existe (con comprobación)

```
DROP TABLE IF EXISTS nombre_tabla;
```

Elimina una tabla (sin comprobación)

```
DROP TABLE nombre_tabla;
```

Elimina una tabla si existe (con comprobación)

```
DROP TABLE IF EXISTS nombre_tabla;
```

## Manipulación de datos

SELECT se utiliza cuando quieres leer (o seleccionar) tus datos

Ejemplos:

- `SELECT * FROM nombre_tabla;` (selecciona todos los campos y valores de la tabla)
- `SELECT campo1, campo2 FROM nombre_tabla;` (selecciona los campos indicados de la tabla)
- `SELECT campo1, campo2 FROM nombre_tabla WHERE campo1=valor;` (selecciona los campos que cumplan con la condición dada por la cláusula WHERE)

INSERT se utiliza cuando quieres añadir (o insertar) nuevos datos.

Ejemplo:

- `INSERT INTO nombre_tabla (campo1, campo2) VALUES (valor1,valor2);`

UPDATE se utiliza cuando quieres cambiar (o actualizar) datos existentes.

Ejemplo:

- `UPDATE nombre_tabla SET campo1=nuevo_valor WHERE campo=valor;` (actualiza el valor del campo1 donde se cumpla la condición de la cláusula WHERE)

DELETE se utiliza cuando quieres eliminar (o borrar) datos existentes.

Ejemplo:

- `DELETE FROM nombre_tabla WHERE campo=valor;` (elimina el valor de la tabla donde se cumpla la condición de la cláusula WHERE)

# **APÉNDICE 5.**

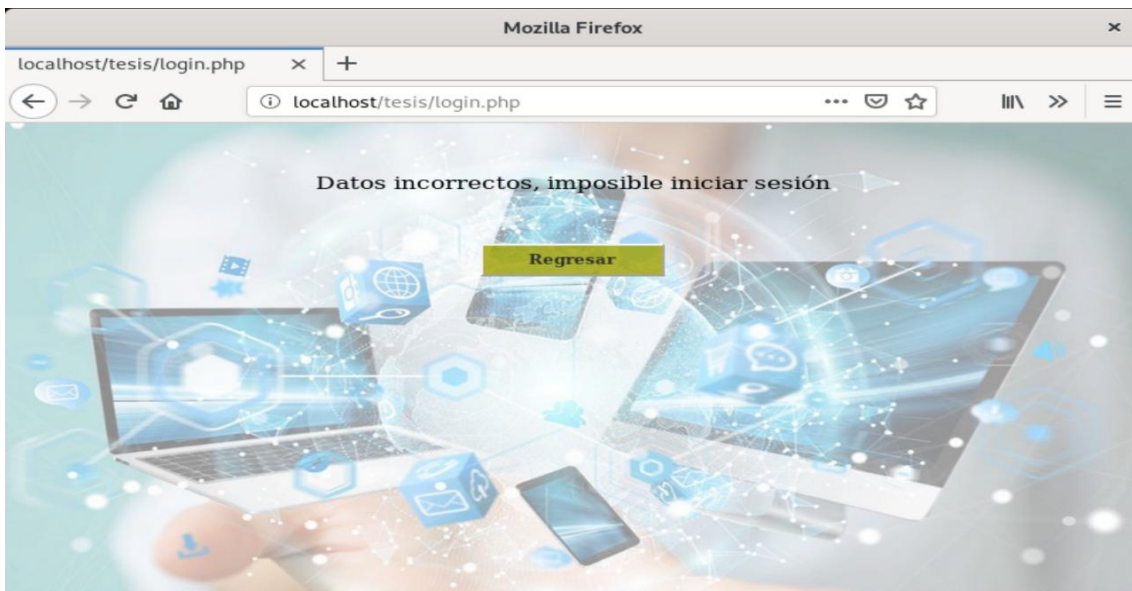
## **USO DEL SISTEMA**

Como primer paso, al tener nuestro servidor de manera local, nos dirigimos a la página **localhost/tesis/sesion.php** y aparecerá la página de inicio donde debemos introducir nuestro usuario y contraseña, posteriormente daremos clic en el botón **Ingresar**.



**Figura 99. Pantalla para inicio de sesión**  
Fuente: Elaboración propia <sup>74</sup>

En caso de introducir algún dato incorrecto o introducir un usuario que no sea permitido, aparecerá la siguiente leyenda **Datos incorrectos, imposible iniciar sesión**.



**Figura 100. Pantalla de inicio de sesión incorrecto.**  
Fuente: Elaboración propia <sup>75</sup>

<sup>74</sup> Captura de pantalla del Sistema

<sup>75</sup> Captura de pantalla del Sistema

Una vez ingresados los datos correctamente, se nos dirigirá a la página **index.php** donde se nos mostrará un menú el cuál podremos elegir la opción que deseamos realizar, estas opciones estarán habilitadas o deshabilitadas según el nivel de usuario con el que ingresemos.



**Figura 101. Menú después de inicio de sesión**  
Fuente: Elaboración propia <sup>76</sup>

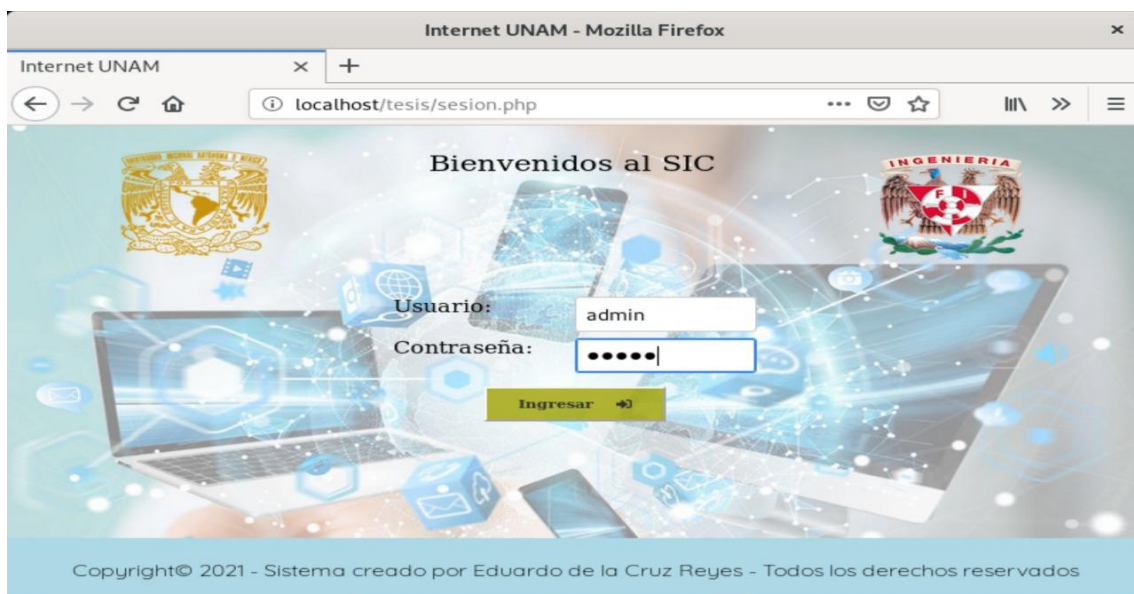


**Figura 102. Menú después de inicio de sesión**  
Fuente: Elaboración propia <sup>77</sup>

<sup>76</sup> Captura de pantalla del Sistema

<sup>77</sup> Captura de pantalla del Sistema

Si ingresamos con el usuario **admin** y su respectiva contraseña, este será el usuario con mayores privilegios. Por lo tanto todas las opciones dentro del menú estarán habilitadas.



**Figura 103. Inicio de sesión con usuario más alto.**  
Fuente: Elaboración propia <sup>78</sup>



**Figura 104. Menú al iniciar sesión con usuario más alto.**  
Fuente: Elaboración propia <sup>79</sup>

<sup>78</sup> Captura de pantalla del Sistema

<sup>79</sup> Captura de pantalla del Sistema

Si ingresamos con el usuario **superuser** y su respectiva contraseña, este será el usuario 'intermedio' en cuanto a privilegios. Por lo tanto las opciones **Deshabilitar Usuario** y **Eliminar Usuario** estarán deshabilitadas.



**Figura 105. Inicio de sesión con usuario intermedio.**  
Fuente: Elaboración propia <sup>80</sup>



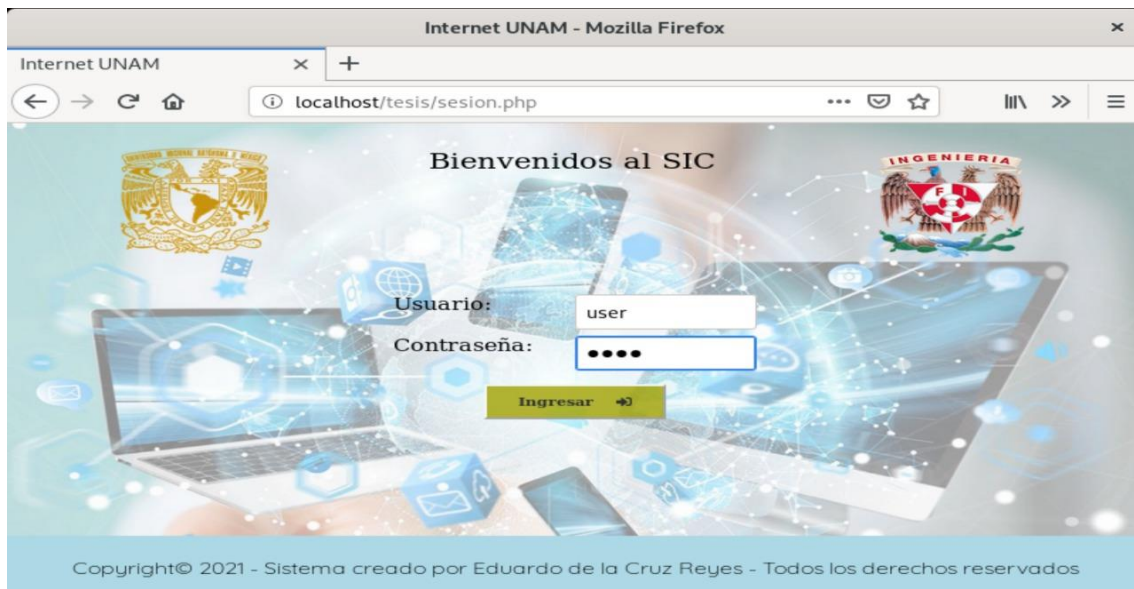
**Figura 106. Menú al iniciar sesión con usuario intermedio.**  
Fuente: Elaboración propia <sup>81</sup>

<sup>80</sup> Captura de pantalla del Sistema

<sup>81</sup> Captura de pantalla del Sistema



Si ingresamos con el usuario **user** y su respectiva contraseña, este será el usuario 'más bajo' en cuanto a privilegios. Por lo tanto todas las opciones estarán deshabilitadas, pudiendo solamente **Listar Usuarios** que se han conectado durante el día.



**Figura 107. Inicio de sesión con usuario más bajo.**  
Fuente: Elaboración propia <sup>82</sup>



**Figura 108. Menú al iniciar sesión con usuario más bajo.**  
Fuente: Elaboración propia <sup>83</sup>

<sup>82</sup> Captura de pantalla del Sistema

<sup>83</sup> Captura de pantalla del Sistema

Al dar clic en la opción **Listar Usuarios** accederemos a la página **usuarios.php** donde veremos una tabla con información tal como **Id, Usuario, Nombre, Número de cuenta, Carrera y Fecha de conexión** de los usuarios conectados durante ese día.

Id	Usuario	Nombre	Numero de cuenta	Carrera	Fecha de conexión
163	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:55:20
164	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:55:20
165	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:55:20
166	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:55:20

**Figura 109. Tabla de usuarios conectados en el día.**

Fuente: Elaboración propia <sup>84</sup>

Más abajo en la página, podremos ver otra tabla con información que nos indicará cuántas veces durante el día se conecta un mismo usuario. Idealmente un mismo usuario podrá registrar hasta 3 conexiones diarias. Al rebasar este número la celda se pintará, denotando que el usuario pudiera estar dando un mal uso de su usuario y contraseña (por ejemplo, compartiendo sus credenciales) con lo cual se procedería a un bloqueo temporal del mismo, teniendo que acudir a realizar una aclaración para poder desbloquear su usuario.

251	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:55:42
252	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:56:21
253	dan131	DANIEL	302891291	Ingeniería en Computación	2021-04-28 00:57:13

Usuario	Cantidad de veces conectados por usuario
aleFI	1
dan131	7
p4t1UNAM	1
sergio20	2

**Figura 110. Tabla con registro de veces conectadas por un usuario.**

Fuente: Elaboración propia <sup>85</sup>

<sup>84</sup> Captura de pantalla del Sistema

<sup>85</sup> Captura de pantalla del Sistema

Al dar clic en la opción **Agregar Usuario**, accederemos a la página **agregar.php** donde deberemos introducir alguna información básica del alumno que será guardada en la base de datos para futuras referencias, así como para que el alumno pueda hacer uso del Internet dentro de la Facultad. De inicio, el botón **Agregar Usuario** se encontrará deshabilitado, una vez ingresados todos los campos, el botón se habilitará y podremos agregarlo.



**Figura 111. Ejemplo de alta de usuario.**  
Fuente: Elaboración propia <sup>86</sup>

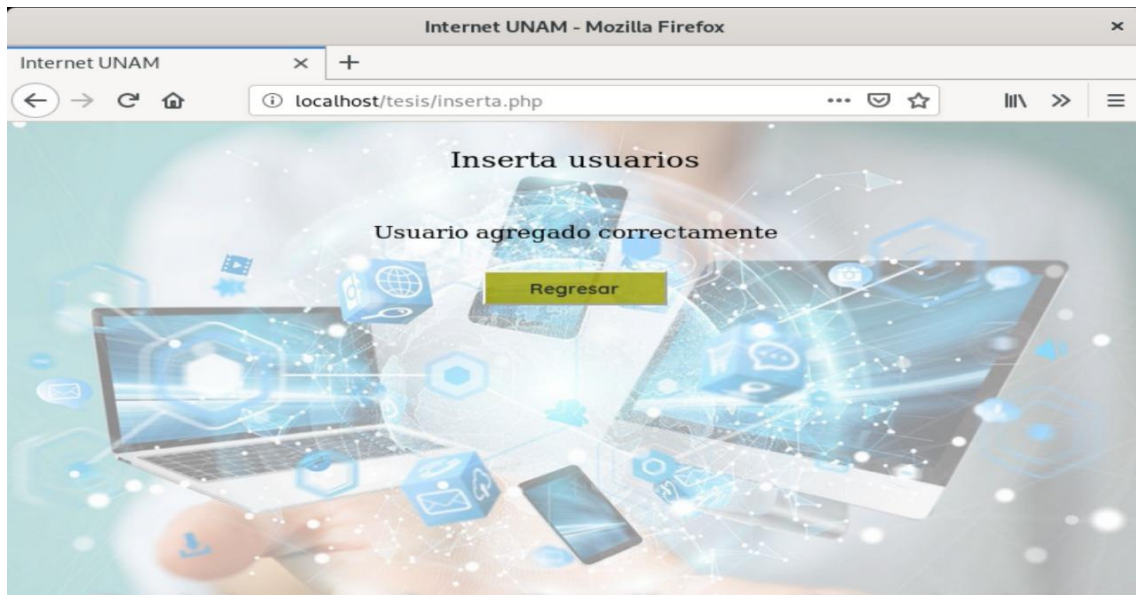


**Figura 112. Ejemplo de alta de usuario.**  
Fuente: Elaboración propia <sup>87</sup>

<sup>86</sup> Captura de pantalla del Sistema

<sup>87</sup> Captura de pantalla del Sistema

Si todos los datos son capturados correctamente por el sistema, nos mostrará otra página con el siguiente mensaje **Usuario agregado correctamente**, con ello el usuario ya podrá hacer uso del Internet con su usuario y contraseña.



**Figura 113. Alta de usuario exitoso.**  
Fuente: Elaboración propia <sup>88</sup>

Una forma rápida de verificar que el usuario ha sido agregado, es dar clic en el botón **Modificar Usuario**, ahí veremos la tabla con los alumnos en la base de datos y observaremos que el usuario agregado aparece hasta el final de la tabla.

The screenshot shows a Mozilla Firefox browser window titled 'Internet UNAM'. The address bar displays 'localhost/tesis/menu.php'. Below the browser window is a table with 8 columns and 8 rows of data. The footer of the page contains the text: 'Copyright© 2021 - Sistema creado por Eduardo de la Cruz Reyes - Todos los derechos reservados'.

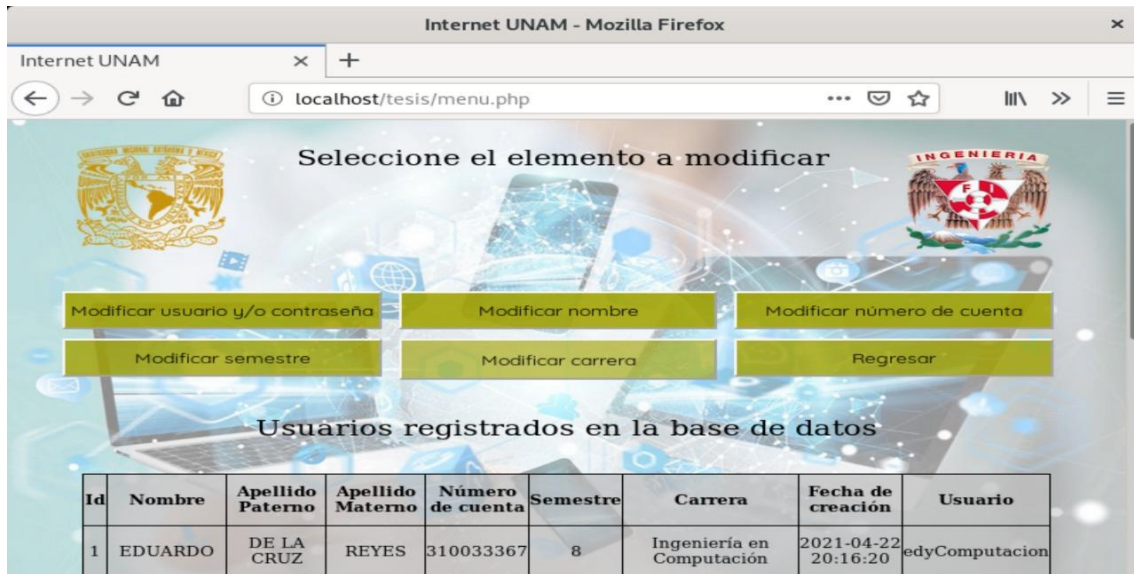
2	SEBASTIAN DE LA CRUZ RESENDIZ	310033369	2	Ingeniería en Computación	2021-03-22 17:35:30	sebas
3	ALEXANDRA RESENDIZ NAZARIO	310033389	9	Ingeniería Civil	2021-04-09 18:46:35	aleFI
4	FRANCISCO DE LA CRUZ REYES	310033380	10	Ingeniería Civil	2021-04-09 18:47:04	paco40
5	CRUZ SERGIO AGUILAR DIAZ	084350262	4	Ingeniería en computacion	2021-04-09 18:40:07	sergio20
9	PATRICIA DE LA CRUZ REYES	310033331	7	Ingeniería en Sistemas Biomédicos	2021-04-09 19:56:28	p4t1UNAM
10	TERESA ORTEGA REYES	310033312	4	Ingeniería Ambiental	2021-04-09 18:23:26	tesa
14	ESPERANZA REYES GOMEZ	310033398	9	Ingeniería Civil	2021-03-29 16:25:27	espe90
16	DANIEL GÓMEZ DELGADO	302891290	6	Ingeniería en Telecomunicaciones	2021-04-23 13:43:20	danielG21

**Figura 114. Verificación de alta de usuario.**  
Fuente: Elaboración propia <sup>89</sup>

<sup>88</sup> Captura de pantalla del Sistema

<sup>89</sup> Captura de pantalla del Sistema

Al dar clic en la opción **Modificar Usuario**, accederemos a la página **menu.php**, dentro de esta observaremos un submenú en el cuál podremos elegir el elemento a modificar de nuestro usuario. Antes de elegir alguna opción, podremos ver los usuarios (alumnos) en la base de datos para saber que elemento deseamos modificar.



**Figura 115. Menú para la opción Modificar Usuario.**  
Fuente: Elaboración propia <sup>90</sup>

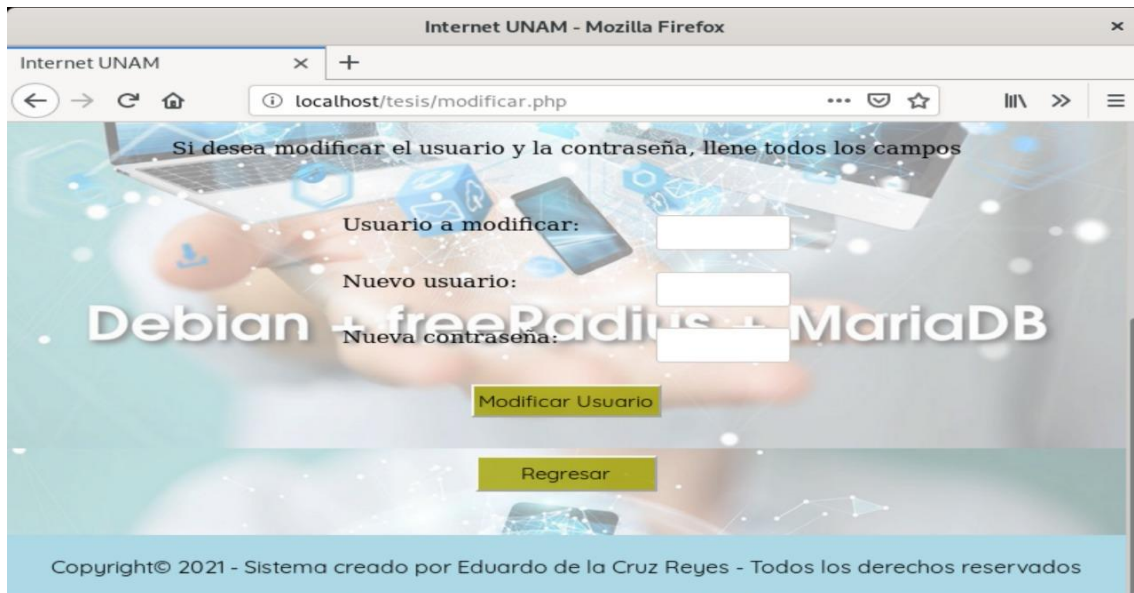
Al dar clic en la opción **Modificar usuario y/o contraseña**, nos dirigiremos a la página **modificar.php** en la cual tendremos instrucciones de cómo utilizar el formulario que ahí aparece. Podremos modificar solamente el usuarios, solamente la contraseña o ambos.



**Figura 116. Instrucciones para modificación de usuario y/o contraseña.**  
Fuente: Elaboración propia <sup>91</sup>

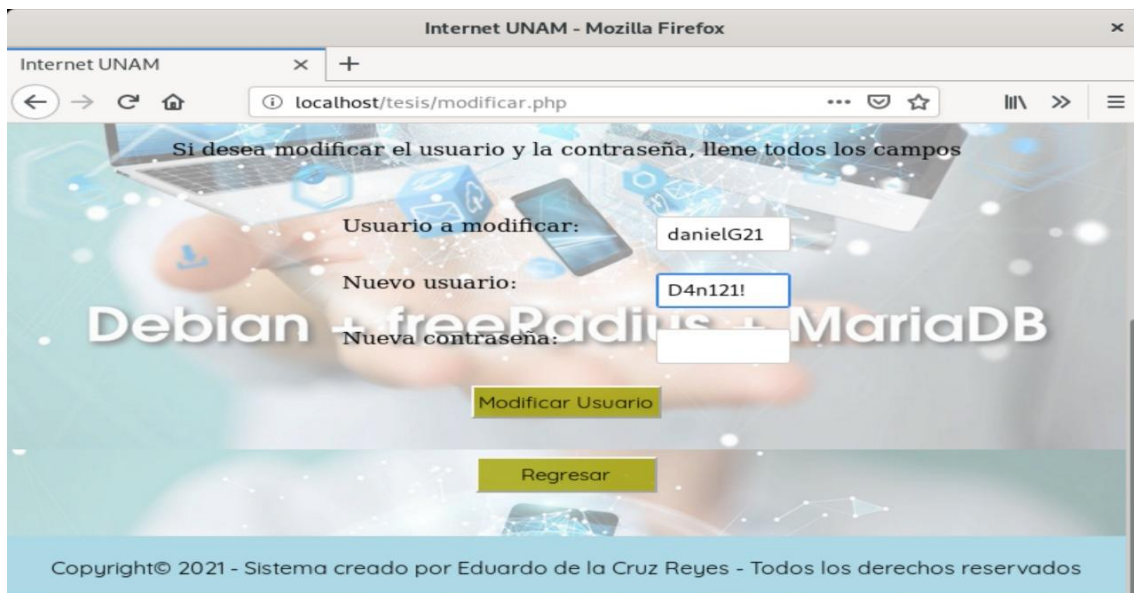
<sup>90</sup> Captura de pantalla del Sistema

<sup>91</sup> Captura de pantalla del Sistema



**Figura 117. Campos para modificación de usuario y/o contraseña.**  
Fuente: Elaboración propia <sup>92</sup>

A manera de ejemplo, procederemos a modificar al usuario **danielG21** por el usuario **D4n121!** Al terminar, daremos clic en el botón **Modificar usuario**.



**Figura 118. Ejemplo de modificación de usuario.**  
Fuente: Elaboración propia <sup>93</sup>

<sup>92</sup> Captura de pantalla del Sistema

<sup>93</sup> Captura de pantalla del Sistema

Si todo sale bien, se nos mostrará una página con el mensaje **Usuario modificado correctamente**, con ello, el usuario ya fue modificado.



**Figura 119. Modificación de usuario exitoso.**  
Fuente: Elaboración propia <sup>94</sup>

Para corroborar, observamos que el usuario con **Id 16** tenía el usuario **danielG21** antes de ser modificado.

2	SEBASTIAN DE LA CRUZ RESENDIZ	310033369	2	Ingeniería en Computación	2021-03-22 17:35:30	sebas
3	ALEXANDRA RESENDIZ NAZARIO	310033389	9	Ingeniería Civil	2021-04-09 18:46:35	aleFI
4	FRANCISCO DE LA CRUZ REYES	310033380	10	Ingeniería Civil	2021-04-09 18:47:04	paco40
5	CRUZ SERGIO AGUILAR DIAZ	084350262	4	Ingeniería en computacion	2021-04-09 18:40:07	sergio20
9	PATRICIA DE LA CRUZ REYES	310033331	7	Ingeniería en Sistemas Biomédicos	2021-04-09 19:56:28	p4t1UNAM
10	TERESA ORTEGA REYES	310033312	4	Ingeniería Ambiental	2021-04-09 18:23:26	tesa
14	ESPERANZA REYES GOMEZ	310033398	9	Ingeniería Civil	2021-03-29 16:25:27	espe90
16	DANIEL GÓMEZ DELGADO	302891290	6	Ingeniería en Telecomunicaciones	2021-04-23 13:43:20	danielG21

Copyright© 2021 - Sistema creado por Eduardo de la Cruz Reyes - Todos los derechos reservados

**Figura 120. Verificación de usuario antes de ser modificado.**  
Fuente: Elaboración propia <sup>95</sup>

<sup>94</sup> Captura de pantalla del Sistema

<sup>95</sup> Captura de pantalla del Sistema

Después de ser modificado, verificamos que el usuario con **Id 16** ya se encuentra con el nuevo usuario **D4n121!**

ID	Nombre	Apellido	ID	Edad	Grado	Fecha de Registro	Usuario
2	SEBASTIAN	DE LA CRUZ	RESENDIZ310033369	2	Ingeniería en Computación	2021-03-22 17:35:30	sebas
3	ALEXANDRA	RESENDIZ	NAZARIO 310033389	9	Ingeniería Civil	2021-04-09 18:46:35	aleFI
4	FRANCISCO	DE LA CRUZ	REYES 310033380	10	Ingeniería Civil	2021-04-09 18:47:04	paco40
5	CRUZ SERGIO	AGUILAR	DIAZ 084350262	4	Ingeniería en computacion	2021-04-09 18:40:07	sergio20
9	PATRICIA	DE LA CRUZ	REYES 310033331	7	Ingeniería en Sistemas Biomédicos	2021-04-09 19:56:28	p4t1UNAM
10	TERESA	ORTEGA	REYES 310033312	4	Ingeniería Ambiental	2021-04-09 18:23:26	tesa
14	ESPERANZA	REYES	GOMEZ 310033398	9	Ingeniería Civil	2021-03-29 16:25:27	espe90
16	DANIEL	GÓMEZ	DELGADO 302891290	6	Ingeniería en Telecomunicaciones	2021-04-23 13:52:13	D4n121!

Copyright© 2021 - Sistema creado por Eduardo de la Cruz Reyes - Todos los derechos reservados

**Figura 121. Verificación de usuario después de ser modificado.**  
Fuente: Elaboración propia <sup>96</sup>

Para la demostración del cambio de contraseña, primero accedemos a nuestra base de datos para verificar que contraseña pertenece a nuestro usuario **D4n121!** y vemos que es **D4n13l21!**

```

eduardo@debian: ~
Archivo Editar Ver Buscar Terminal Ayuda
+-----+-----+-----+-----+-----+-----+
| 6 | zFNaHdtT | Cleartext-Password | := | i3up8Z5D |
| 9 | iqzBZiRF | Cleartext-Password | := | rDw6EPcW |
| 11 | lalo | Cleartext-Password | := | lalo |
| 12 | edy | Cleartext-Password | := | edy |
| 14 | sebas | Cleartext-Password | := | sebas |
| 17 | hRzQ!Z | Cleartext-Password | := | gVHU&C |
| 18 | *tyBtM | Cleartext-Password | := | $uJhBt |
| 19 | u!YEzd | Cleartext-Password | := | UFuQDq |
| 29 | test | Cleartext-Password | := | test |
| 35 | fiUNAM | Cleartext-Password | := | ingeUnam |
| 40 | sergio21 | Cleartext-Password | := | 12345 |
| 52 | edyComputacion | Cleartext-Password | := | myPa$3dy |
| 53 | sebas | Cleartext-Password | := | sebas21 |
| 54 | aleFI | Cleartext-Password | := | 23092018 |
| 55 | paco21 | Cleartext-Password | := | 2468 |
| 56 | sergio20 | Cleartext-Password | := | 123456 |
| 61 | p4t1UNAM | Cleartext-Password | := | 12345 |
| 62 | tesa | Cleartext-Password | := | 179008 |
| 66 | espe90 | Cleartext-Password | := | 123456 |
| 68 | D4n121! | Cleartext-Password | := | D4n13l21! |
+-----+-----+-----+-----+-----+
24 rows in set (0.001 sec)

MariaDB [radius]>
  
```

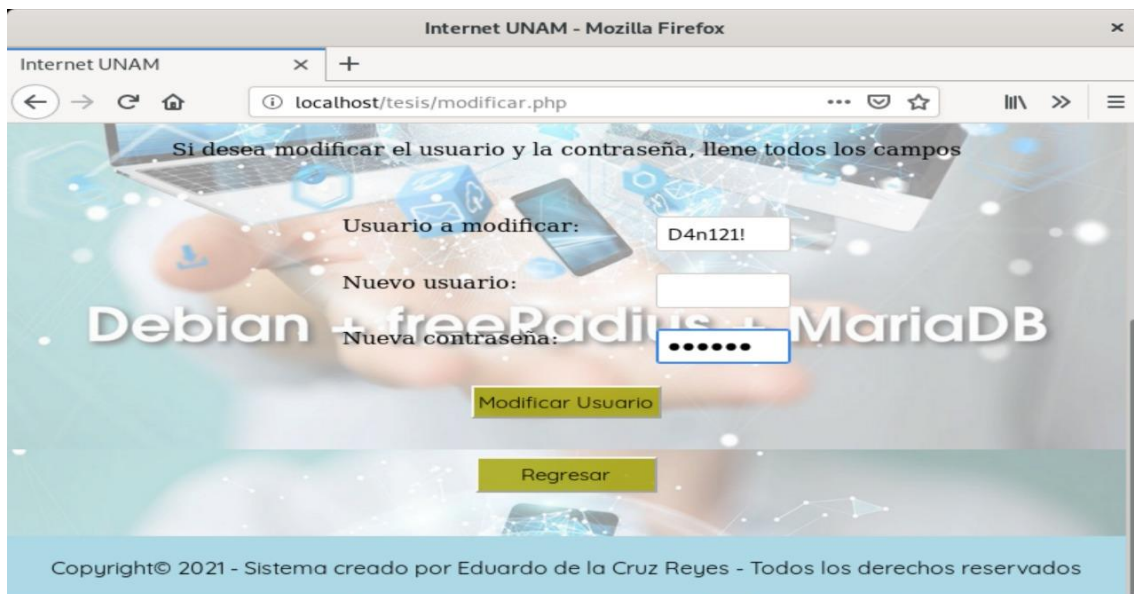
**Figura 122. Verificación de contraseña de usuarios desde terminal.**  
Fuente: Elaboración propia <sup>97</sup>

<sup>96</sup> Captura de pantalla del Sistema

<sup>97</sup> Captura de pantalla del Sistema

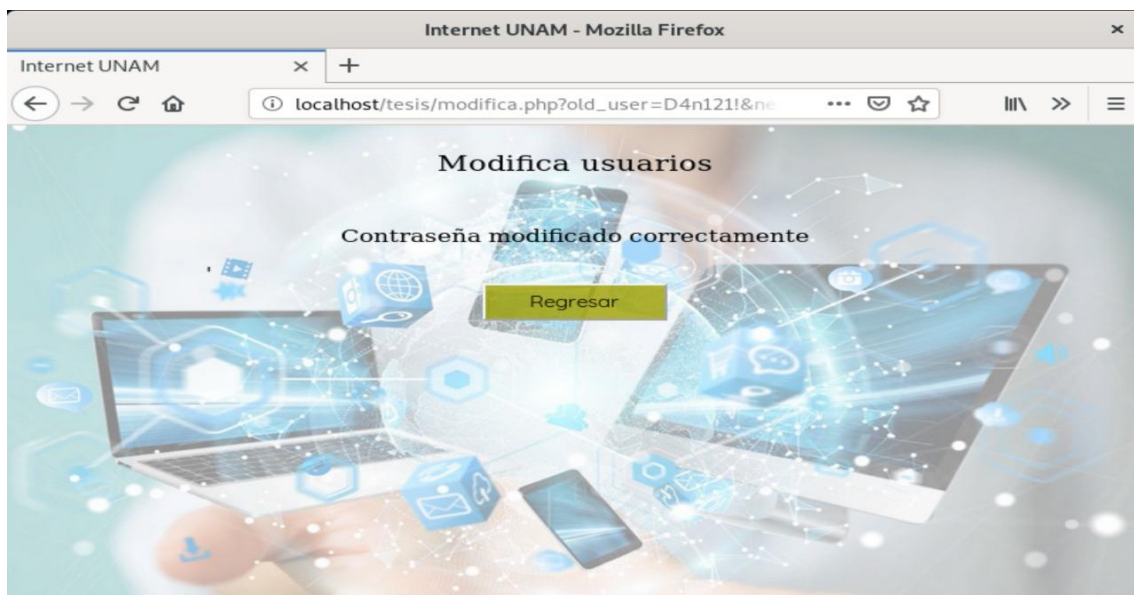


Dentro de nuestro sistema, llenamos los campos **Usuario a modificar** con el usuario y **Nueva contraseña** con la nueva contraseña y damos clic en **Modificar Usuario**.



**Figura 123. Ejemplo de modificación de contraseña.**  
Fuente: Elaboración propia <sup>98</sup>

Si la contraseña se modifica correctamente, se nos dirigirá a otra página con el mensaje **Contraseña modificada correctamente** y habremos concluido la modificación.



**Figura 124. Cambio de contraseña exitosa.**  
Fuente: Elaboración propia <sup>99</sup>

<sup>98</sup> Captura de pantalla del Sistema

<sup>99</sup> Captura de pantalla del Sistema

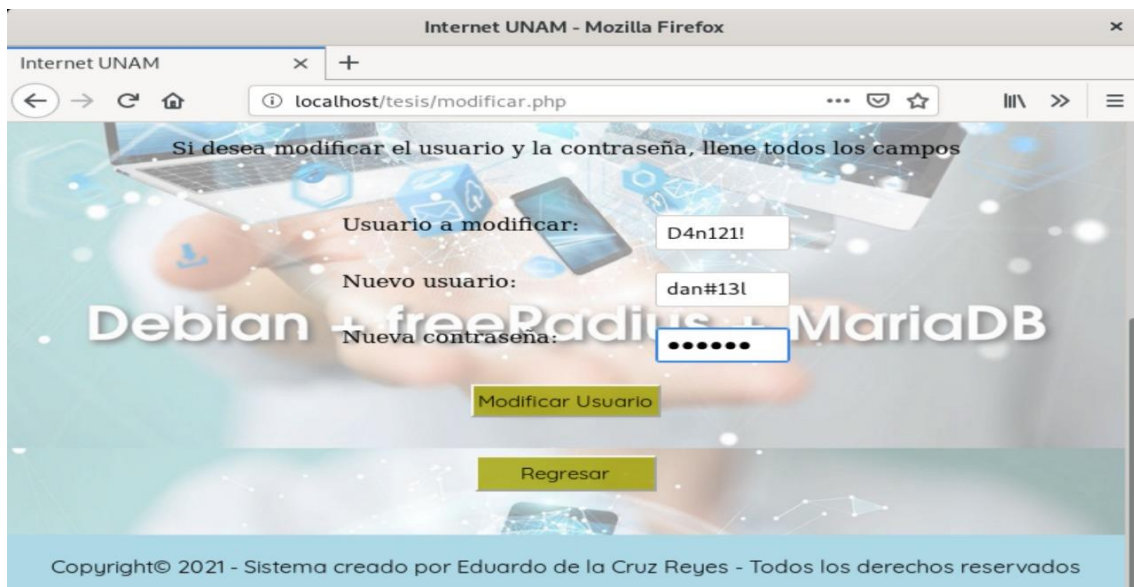
Para verificar el cambio, accedemos nuevamente a la base de datos y vemos que la contraseña del usuario **D4n121!** ha sido modificada exitosamente con la que introducimos en el sistema.

```

eduardo@debian: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
-----
 6 | zFNaHdtT | Cleartext-Password | := | i3up8Z5D |
 9 | iqzBZiRF | Cleartext-Password | := | rDw6EPcW |
11 | lalo      | Cleartext-Password | := | lalo      |
12 | edy      | Cleartext-Password | := | edy      |
14 | sebas    | Cleartext-Password | := | sebas    |
17 | hRzQ!Z  | Cleartext-Password | := | gVHU&C   |
18 | *tyBtM  | Cleartext-Password | := | $uJhBt   |
19 | u!YEzd  | Cleartext-Password | := | UFuQDq   |
29 | test     | Cleartext-Password | := | test     |
35 | fiUNAM   | Cleartext-Password | := | ingeUnam |
40 | sergio21 | Cleartext-Password | := | 12345    |
52 | edyComputacion | Cleartext-Password | := | myPa$3dy |
53 | sebas    | Cleartext-Password | := | sebas21  |
54 | aleFI    | Cleartext-Password | := | 23092018 |
55 | paco21   | Cleartext-Password | := | 2468     |
56 | sergio20 | Cleartext-Password | := | 123456   |
61 | p4t1UNAM | Cleartext-Password | := | 12345    |
62 | tesa     | Cleartext-Password | := | 179008   |
66 | espe90   | Cleartext-Password | := | 123456   |
68 | D4n121!  | Cleartext-Password | := | xFs8E!   |
-----
24 rows in set (0.000 sec)
MariaDB [radius]>
  
```

**Figura 125. Verificación de cambio de contraseña de usuarios desde terminal.**  
Fuente: Elaboración propia <sup>100</sup>

En este paso se procederá a modificar el usuario y la contraseña, llenamos todos los campos y damos clic en el botón **Modificar Usuario** (La nueva contraseña será jS#bK8)

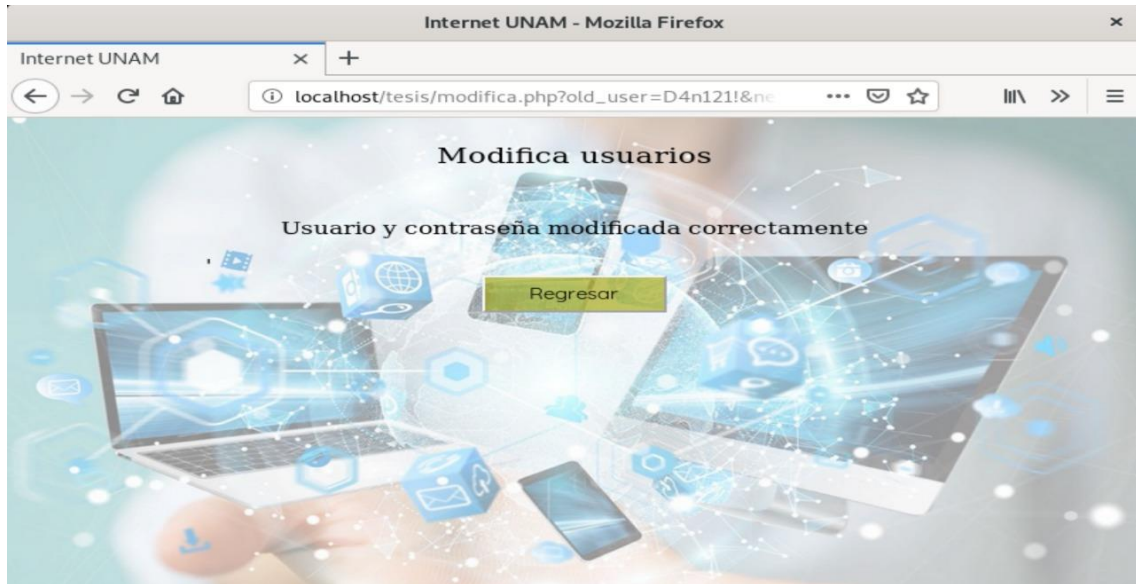


**Figura 126. Ejemplo de cambio de usuario y contraseña.**  
Fuente: Elaboración propia <sup>101</sup>

<sup>100</sup> Captura de pantalla del Sistema

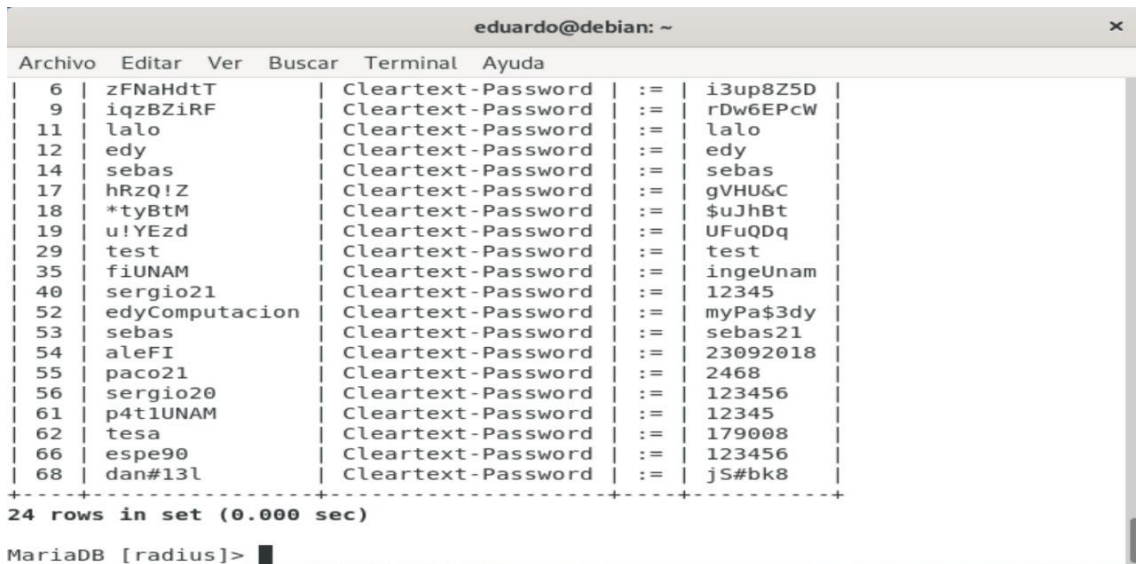
<sup>101</sup> Captura de pantalla del Sistema

Al finalizar, se nos mostrará otra página con el mensaje **Usuario y contraseña modificada correctamente**.



**Figura 127. Cambio de usuario y contraseña exitosa.**  
Fuente: Elaboración propia <sup>102</sup>

Accedemos a la base de datos para verificar que el usuario y la contraseña fueron cambiados correctamente.



**Figura 128. Verificación de cambio de usuario y contraseña desde terminal.**  
Fuente: Elaboración propia <sup>103</sup>

<sup>102</sup> Captura de pantalla del Sistema

<sup>103</sup> Captura de pantalla del Sistema

Al dar clic en el botón **Modificar nombre**, este nos enviara a la página **nombre.php** donde deberemos introducir el número de cuenta del usuario a buscar para que al dar clic en el botón **Buscar Usuario** se nos muestre la tabla con sus datos.



**Figura 129. Búsqueda de usuario por número de cuenta.**  
Fuente: Elaboración propia <sup>104</sup>

Una vez que se nos muestren los datos, podremos elegir entre modificar el nombre, el apellido paterno o el apellido materno por separado o modificar los tres campos al mismo tiempo.

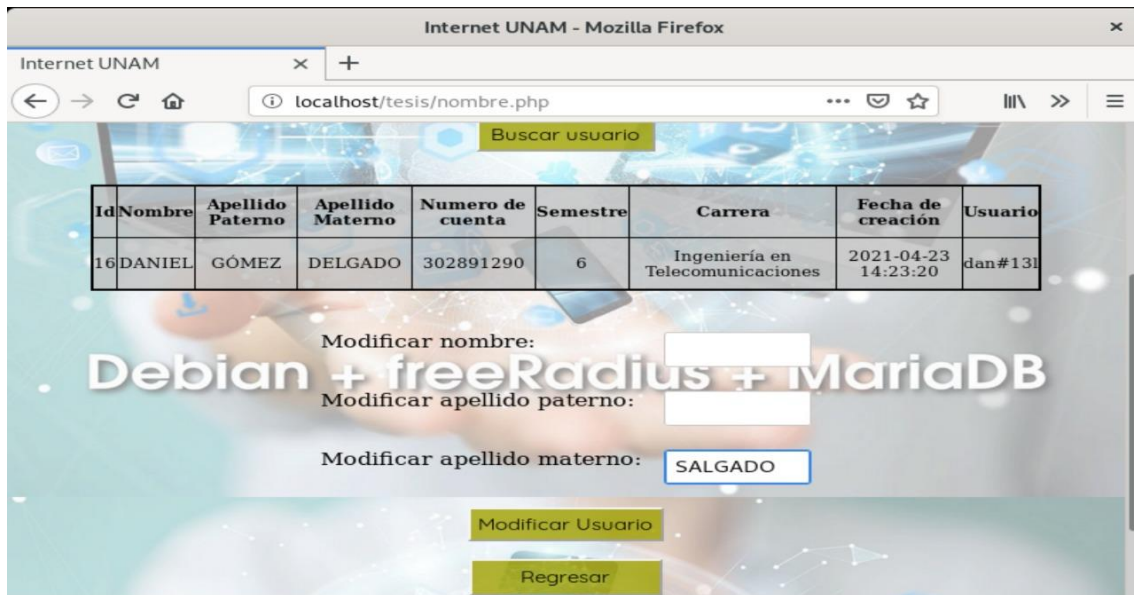


**Figura 130. Muestra de resultados obtenidos.**  
Fuente: Elaboración propia <sup>105</sup>

<sup>104</sup> Captura de pantalla del Sistema

<sup>105</sup> Captura de pantalla del Sistema

En este ejemplo, se modificará solo el apellido materno. Pero si lo deseamos podemos modificar los 3 campos para editar el nombre completo. Al finalizar, daremos clic en el botón **Modificar Usuario**.



**Figura 131. Modificación de apellido materno**  
Fuente: Elaboración propia <sup>106</sup>

Una vez modificado el usuario, nos aparecerá otra página con el mensaje **Apellido materno modificado correctamente.**



**Figura 132. Modificación de apellido materno exitoso.**  
Fuente: Elaboración propia <sup>107</sup>

<sup>106</sup> Captura de pantalla del Sistema

<sup>107</sup> Captura de pantalla del Sistema

Buscamos de nuevo el usuario por su número de cuenta para verificar que se modificó correctamente.



**Figura 133. Confirmación de cambio de apellido materno.**

Fuente: Elaboración propia <sup>108</sup>

Al dar clic en **Modificar número de cuenta**, esta vez buscaremos al usuario por su usuario y daremos clic en **Buscar Usuario** para que al dar clic en el botón **Buscar Usuario** se nos muestre la tabla con sus datos.



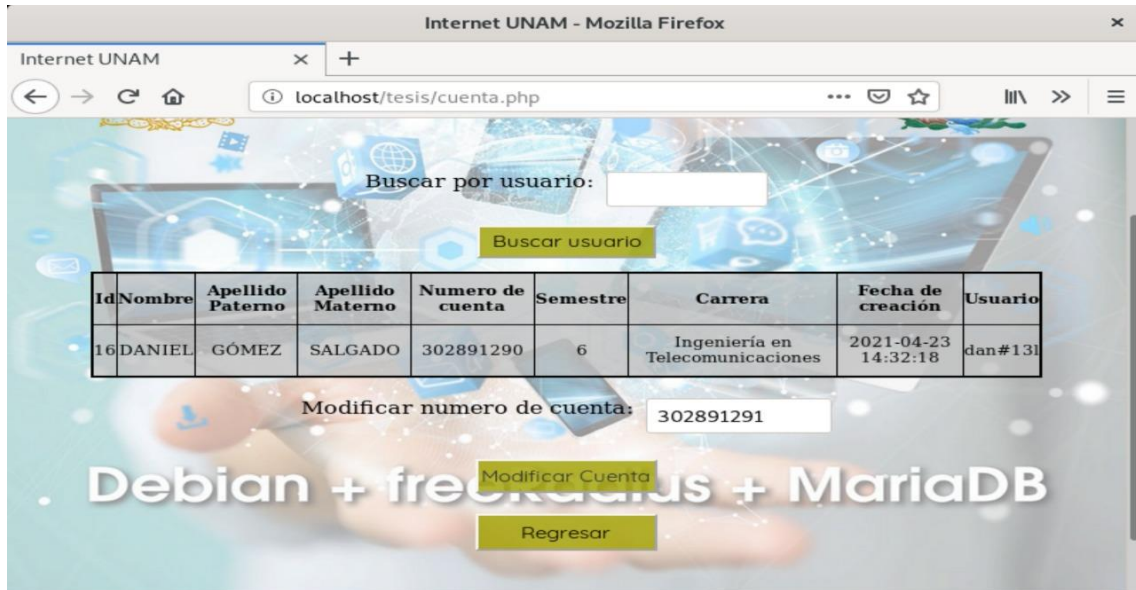
**Figura 134. Búsqueda de usuario por usuario.**

Fuente: Elaboración propia <sup>109</sup>

<sup>108</sup> Captura de pantalla del Sistema

<sup>109</sup> Captura de pantalla del Sistema

Una vez que se nos muestran los datos, llenamos el campo **Modificar número de cuenta** con la nueva cuenta y damos clic en **Modificar cuenta**.



**Figura 135. Muestra de resultados obtenidos.**  
Fuente: Elaboración propia <sup>110</sup>

Si el cambio de número de cuenta fue exitoso, se nos mostrará otra página con el mensaje **Número de cuenta cambiado correctamente**.

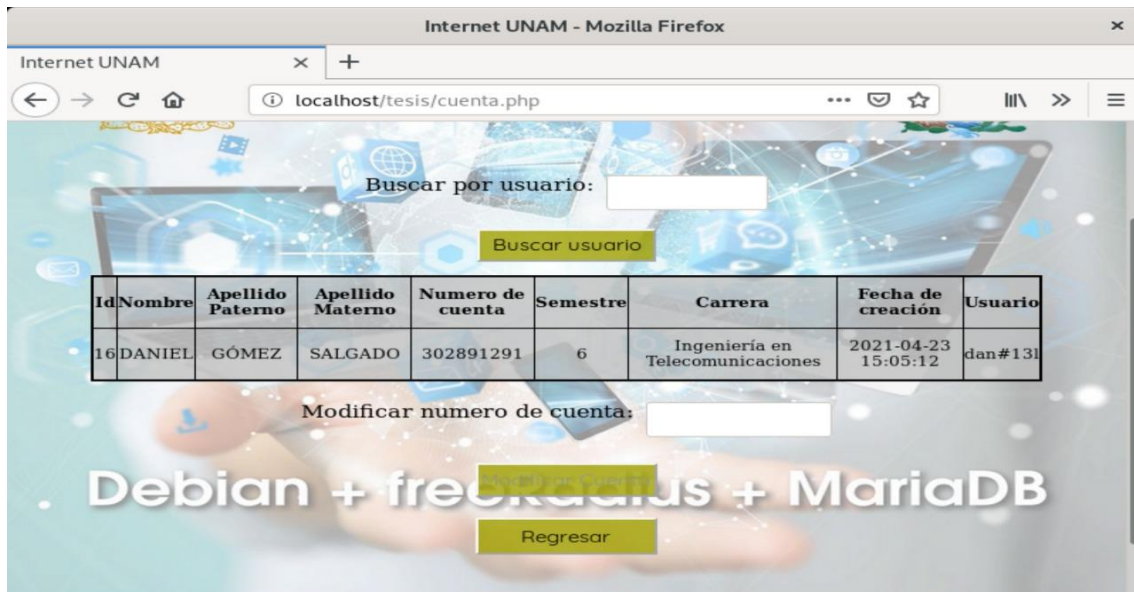


**Figura 136. Modificación de cuenta exitosa.**  
Fuente: Elaboración propia <sup>111</sup>

<sup>110</sup> Captura de pantalla del Sistema

<sup>111</sup> Captura de pantalla del Sistema

A manera de comprobación, colocamos de nueva cuenta el usuario y al aparecer sus datos, veremos que el número de cuenta fue modificado correctamente.



**Figura 137. Confirmación de cambio de cuenta exitoso.**  
Fuente: Elaboración propia <sup>112</sup>

Al dar clic en **Modificar Semestre**, se nos mostrará la página **semestre.php** en la cual debemos introducir el número de cuenta para que al dar clic en el botón **Buscar usuario**, aparezca una tabla con los datos del usuario.



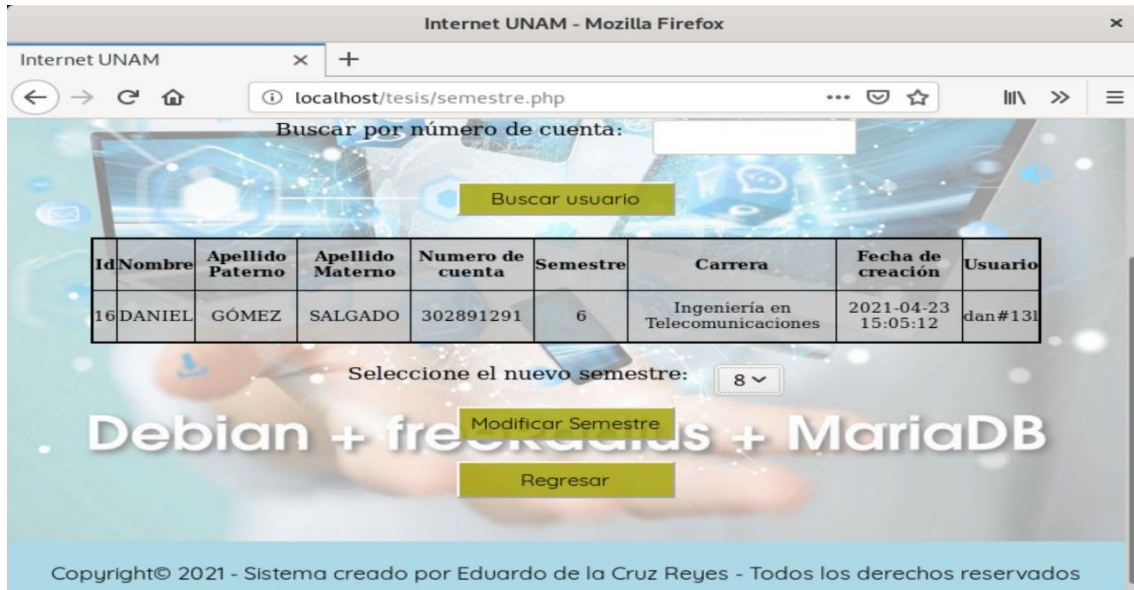
**Figura 138. Búsqueda de usuario por número de cuenta.**  
Fuente: Elaboración propia <sup>113</sup>

<sup>112</sup> Captura de pantalla del Sistema

<sup>113</sup> Captura de pantalla del Sistema

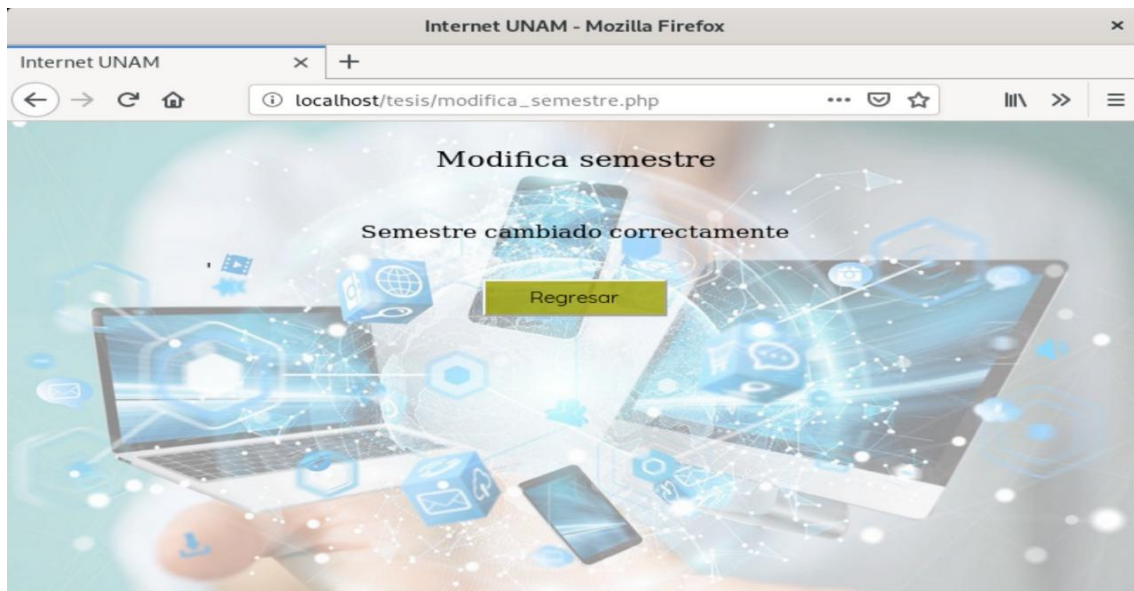


Una vez mostrados los datos, seleccionamos el nuevo semestre de un menú desplegable y damos clic en **Modificar Semestre**.



**Figura 139. Muestra de resultados obtenidos.**  
Fuente: Elaboración propia <sup>114</sup>

Si el cambio de semestre fue exitoso, se nos mostrará otra página con el mensaje **Semestre cambiado correctamente**.



**Figura 140. Modificación de semestre exitoso.**  
Fuente: Elaboración propia <sup>115</sup>

<sup>114</sup> Captura de pantalla del Sistema

<sup>115</sup> Captura de pantalla del Sistema

A manera de comprobación, volvemos a buscar al usuario por su número de cuenta para verificar que el semestre se cambió correctamente.



**Figura 141. Confirmación de cambio de semestre exitoso.**  
Fuente: Elaboración propia <sup>116</sup>

Al dar clic en la opción **Modificar Semestre** nos dirigiremos a la página **carrera.php** y deberemos introducir el número de cuenta del usuario para que al dar clic en el botón **Buscar usuario**, aparezca una tabla con los datos del usuario.

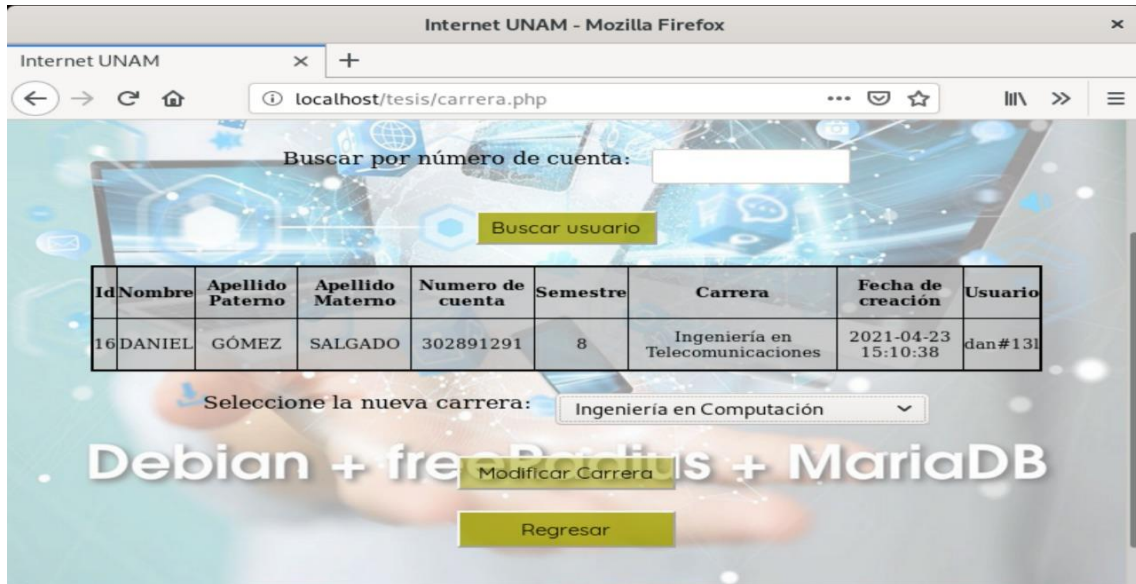


**Figura 142. Búsqueda de usuario por número de cuenta.**  
Fuente: Elaboración propia <sup>117</sup>

<sup>116</sup> Captura de pantalla del Sistema

<sup>117</sup> Captura de pantalla del Sistema

Una vez mostrados los datos, seleccionamos la nueva carrera de un menú desplegable y damos clic en **Modificar Carrera**.



**Figura 143. Muestra de resultados obtenidos.**  
Fuente: Elaboración propia <sup>118</sup>

Si el cambio de carrera fue exitoso, se nos mostrará otra página con el mensaje **Carrera cambiada correctamente**.

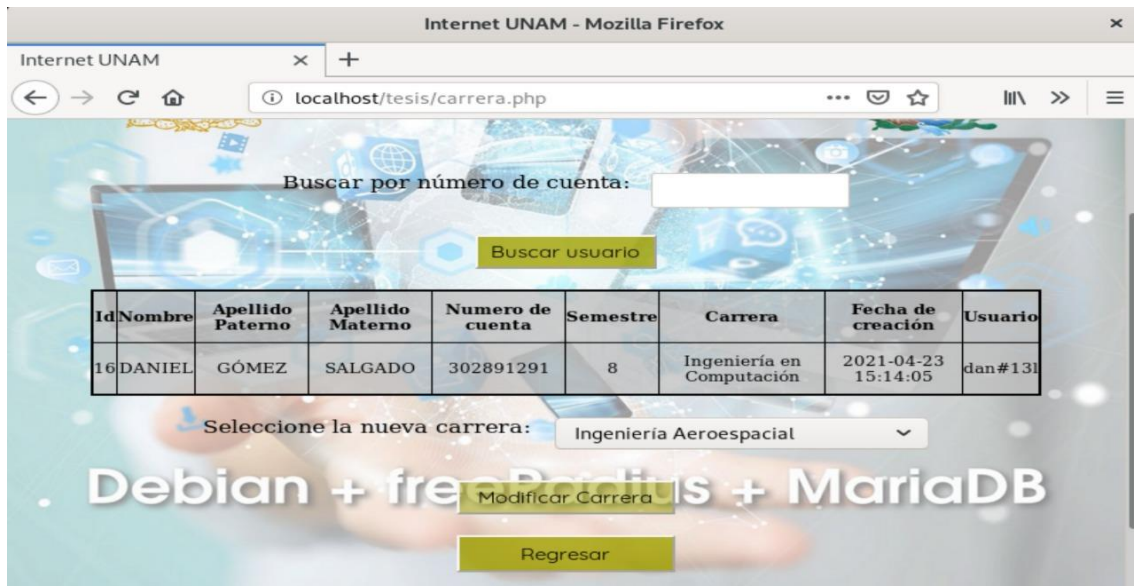


**Figura 144. Modificación de carrera exitosa.**  
Fuente: Elaboración propia <sup>119</sup>

<sup>118</sup> Captura de pantalla del Sistema

<sup>119</sup> Captura de pantalla del Sistema

A manera de comprobación, volvemos a buscar al usuario por su número de cuenta para verificar que la carrera se cambió correctamente.



**Figura 145. Verificación de cambio de carrera exitosa.**  
Fuente: Elaboración propia <sup>120</sup>

Daremos clic en el botón **Deshabilitar Usuario**

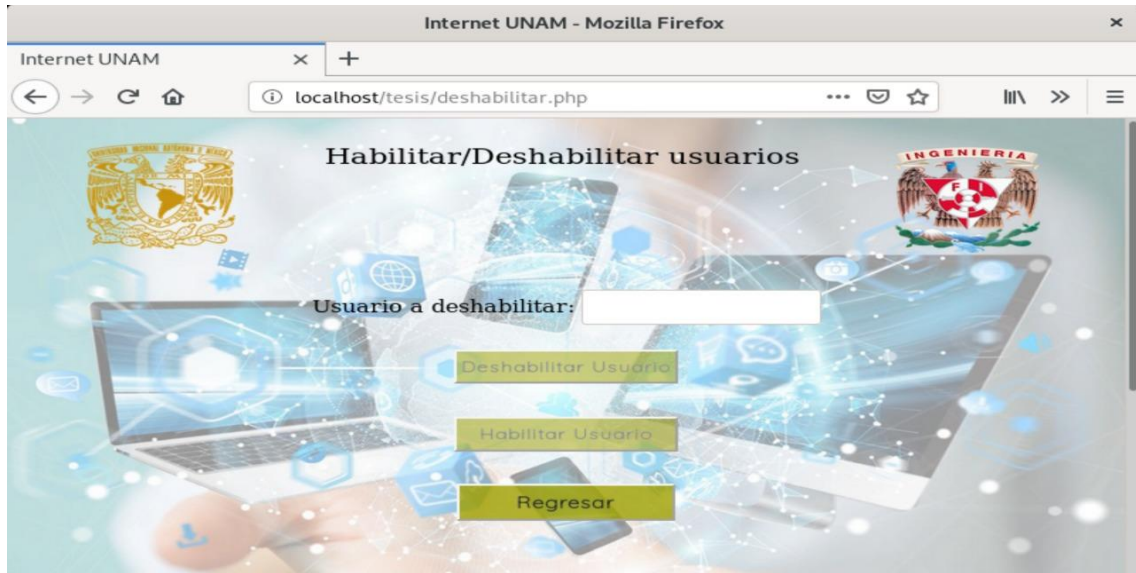


**Figura 146. Menú inicial.**  
Fuente: Elaboración propia <sup>121</sup>

<sup>120</sup> Captura de pantalla del Sistema

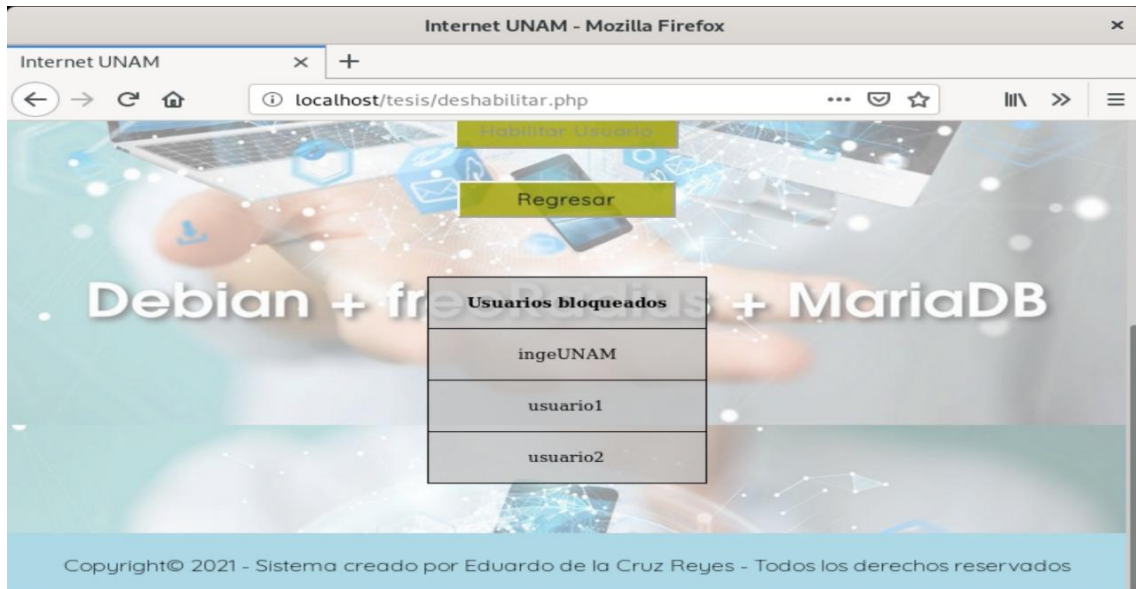
<sup>121</sup> Captura de pantalla del Sistema

Esto nos dirigirá a la página **deshabilitar.php** donde deberemos introducir el usuario a **habilitar** o **deshabilitar**.



**Figura 147. Página para habilitar / deshabilitar usuarios.**  
Fuente: Elaboración propia <sup>122</sup>

Para ayudarnos, más abajo podemos ver una tabla con los usuarios bloqueados hasta el momento.

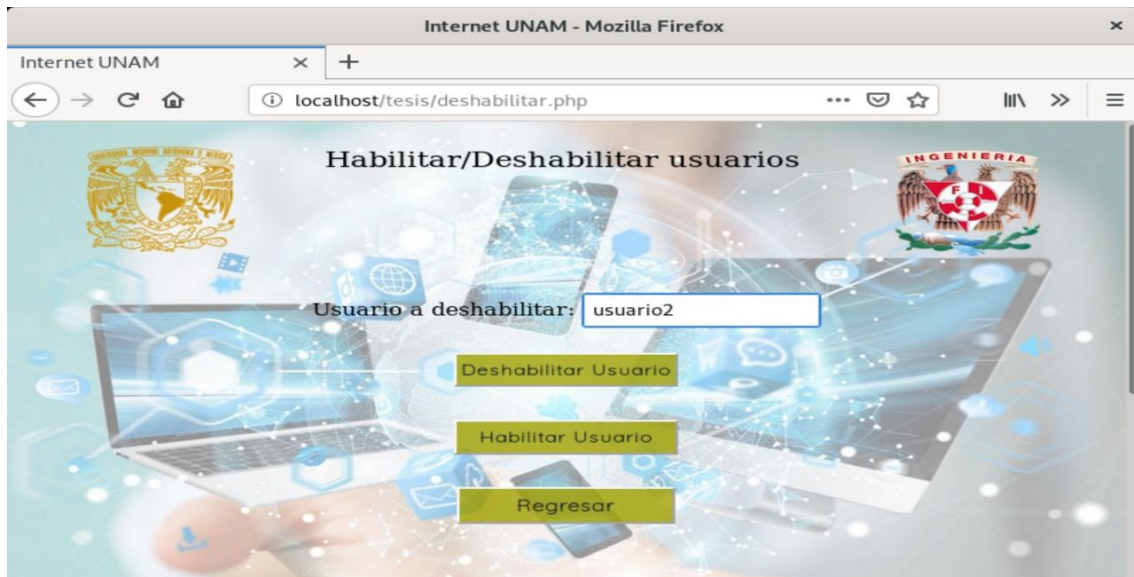


**Figura 148. Tabla con usuarios bloqueados.**  
Fuente: Elaboración propia <sup>123</sup>

<sup>122</sup> Captura de pantalla del Sistema

<sup>123</sup> Captura de pantalla del Sistema

Como ejemplo, deshabilitaremos el usuario **usuario2**. Una vez llenado el campo, deberemos dar clic en el botón **Habilitar Usuario**.



**Figura 149. Ejemplo de habilitación de usuario.**  
Fuente: Elaboración propia <sup>124</sup>

Si por error, se intenta deshabilitar, la siguiente página nos arrojará un mensaje diciendo que el usuario ya está deshabilitado y deberemos volver.



**Figura 150. Error en habilitación de usuario.**  
Fuente: Elaboración propia <sup>125</sup>

<sup>124</sup> Captura de pantalla del Sistema

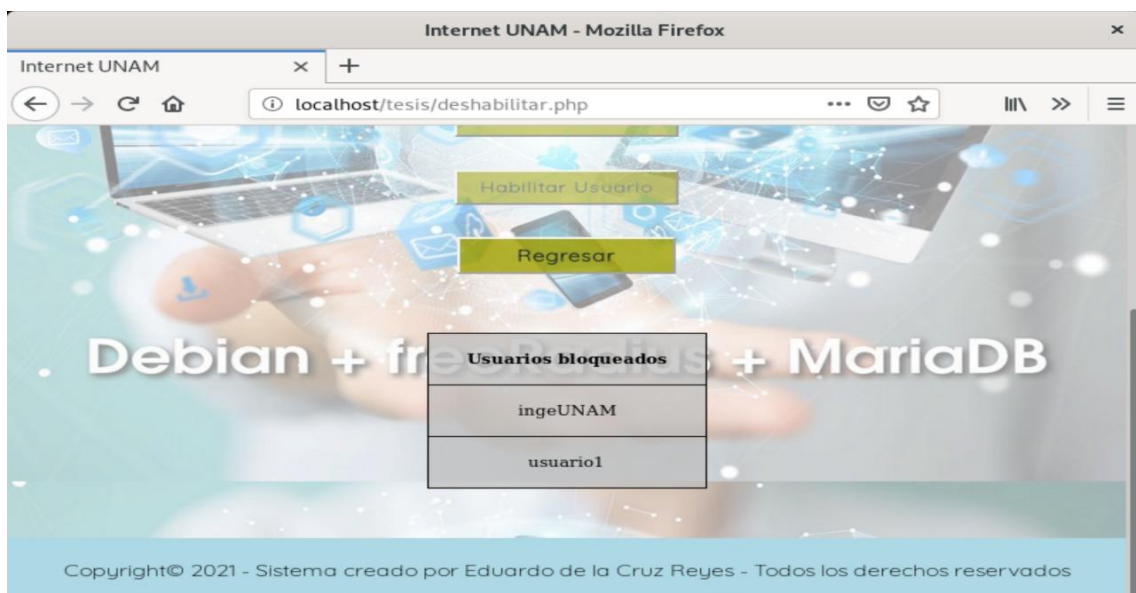
<sup>125</sup> Captura de pantalla del Sistema

Ahora al hacer clic en el botón **Habilitar Usuario**, si este fue exitoso, se nos mostrará el mensaje **Usuario habilitado correctamente**.



**Figura 151. Habilidad de usuario exitoso.**  
Fuente: Elaboración propia <sup>126</sup>

Podemos verificar en la tabla de usuarios bloqueados que ya no existe el usuario.

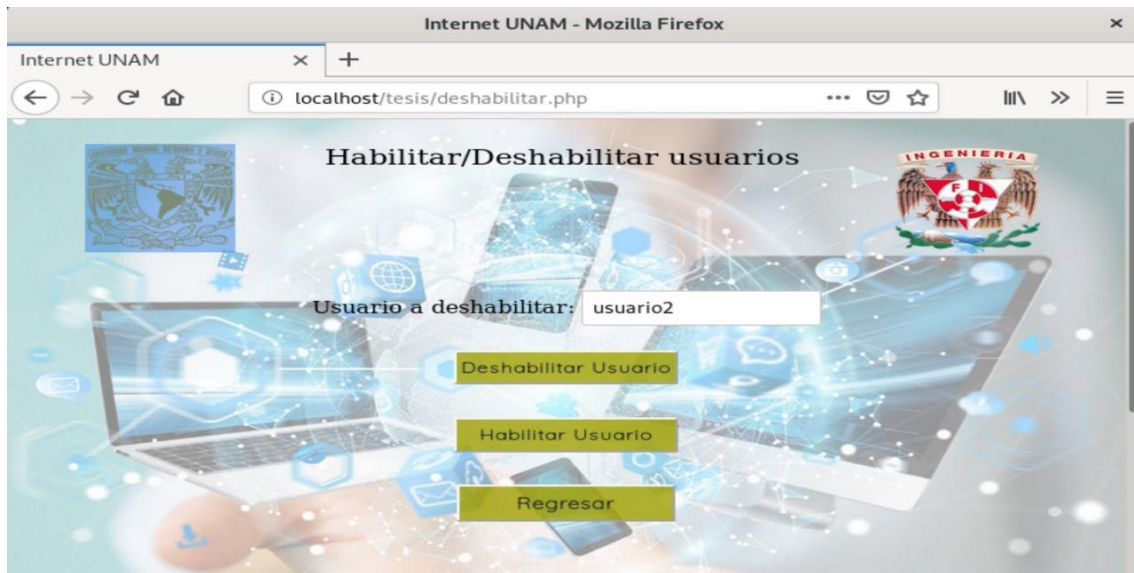


**Figura 152. Verificación de usuario deshabilitado.**  
Fuente: Elaboración propia <sup>127</sup>

<sup>126</sup> Captura de pantalla del Sistema

<sup>127</sup> Captura de pantalla del Sistema

Como ejemplo, volveremos a deshabilitar el usuario **usuario2**, colocándolo en el campo correspondiente.



**Figura 153. Ejemplo para deshabilitar un usuario.**  
Fuente: Elaboración propia <sup>128</sup>

Al dar clic en el botón **Deshabilitar Usuario**, si ha sido exitoso, se nos mostrará el mensaje de **Usuario deshabilitado correctamente**.



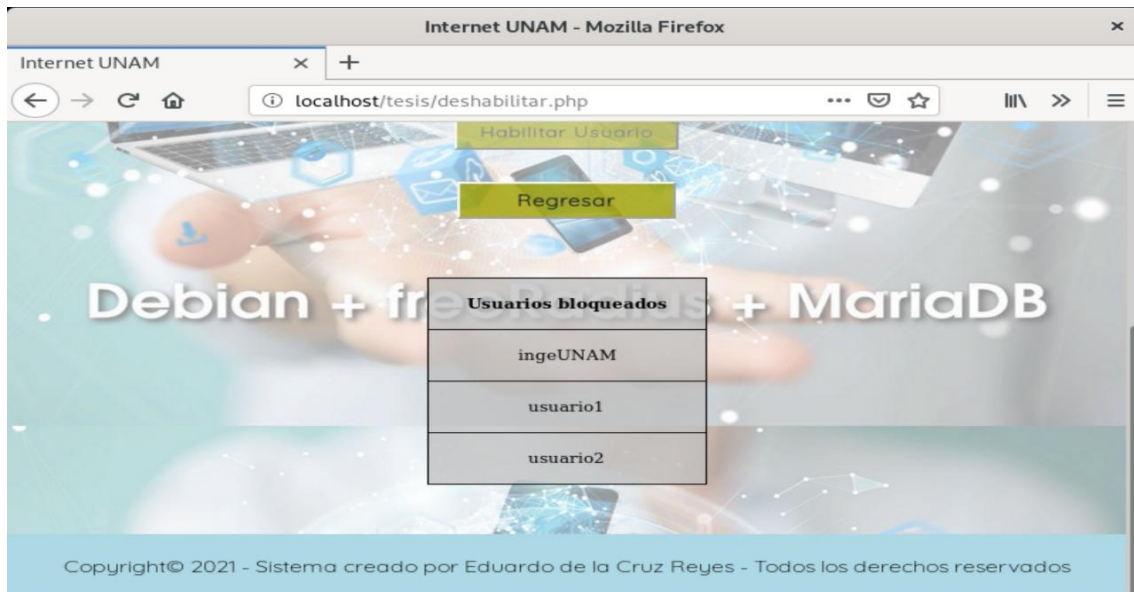
**Figura 154. Deshabilitar un usuario de manera exitosa.**  
Fuente: Elaboración propia <sup>129</sup>

<sup>128</sup> Captura de pantalla del Sistema

<sup>129</sup> Captura de pantalla del Sistema

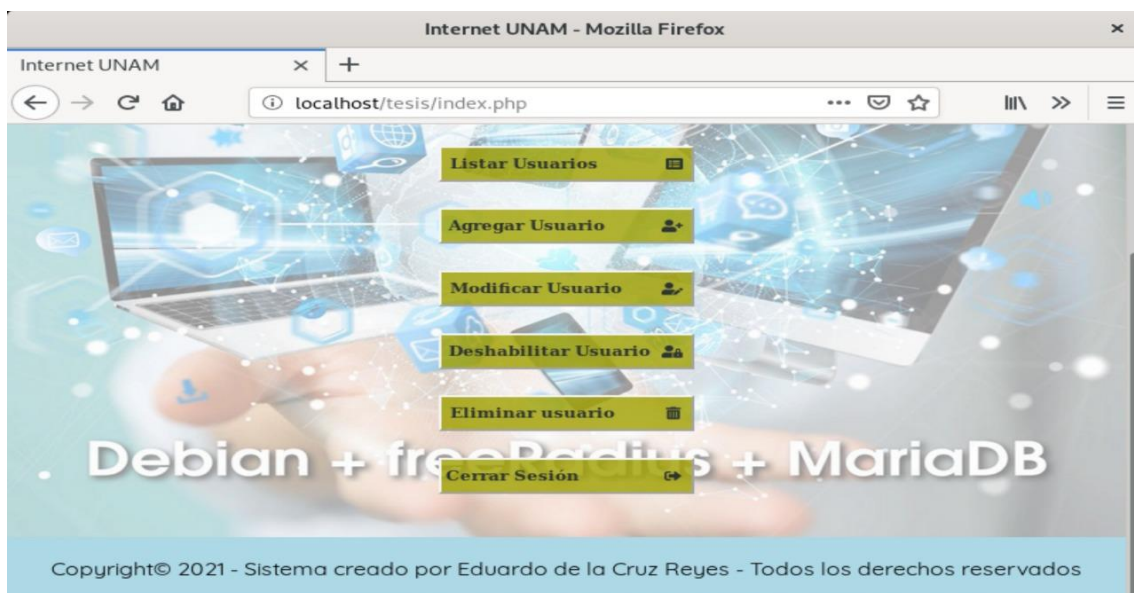


Podemos verificar que el usuario se encuentra de nuevo en la tabla de **usuarios deshabilitados**.



**Figura 155. Verificación para deshabilitar un usuario.**  
Fuente: Elaboración propia <sup>130</sup>

Daremos clic en el botón **Eliminar Usuario**.

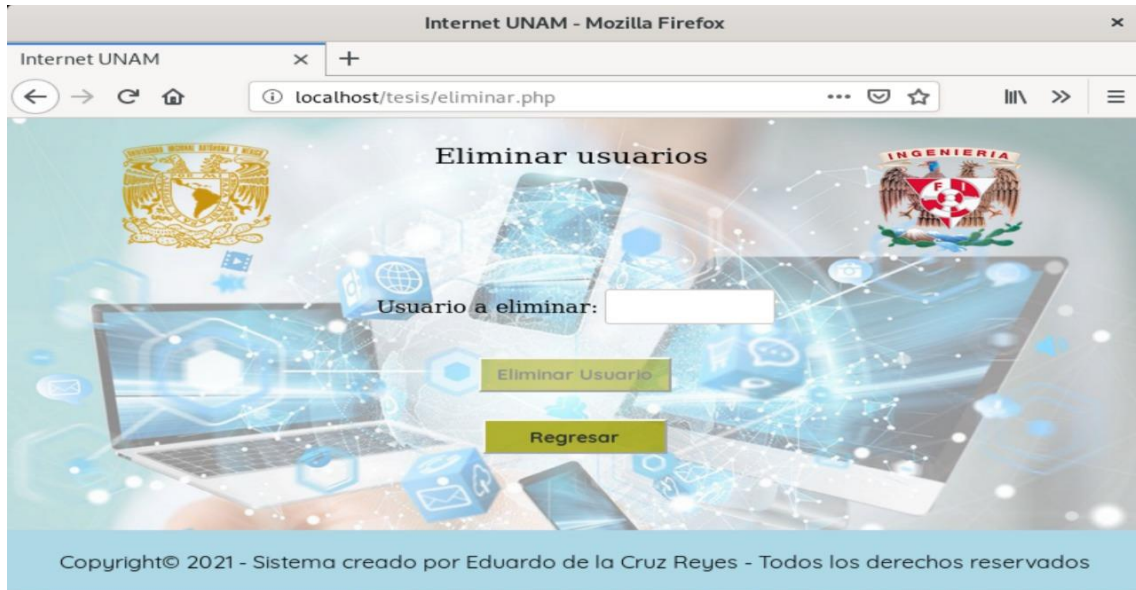


**Figura 156. Menú inicial.**  
Fuente: Elaboración propia <sup>131</sup>

<sup>130</sup> Captura de pantalla del Sistema

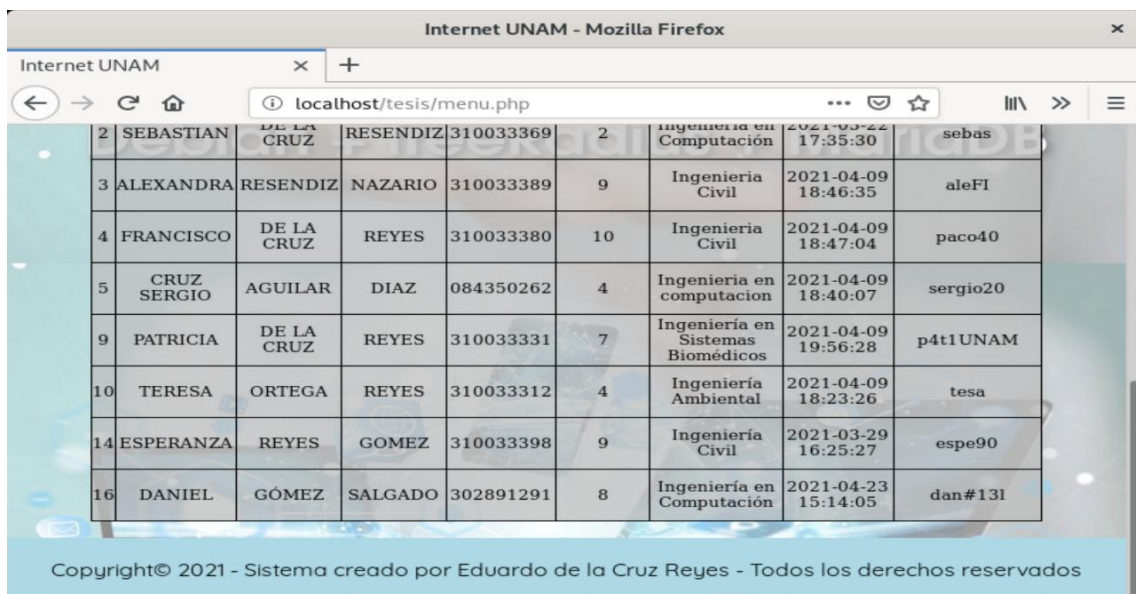
<sup>131</sup> Captura de pantalla del Sistema

Esto nos dirigirá a la página **eliminar.php** donde deberemos introducir el usuario a eliminar.



**Figura 157. Página para eliminación de usuarios.**  
Fuente: Elaboración propia <sup>132</sup>

Escogeremos el usuario con **id 14** y usuario **espe90**.

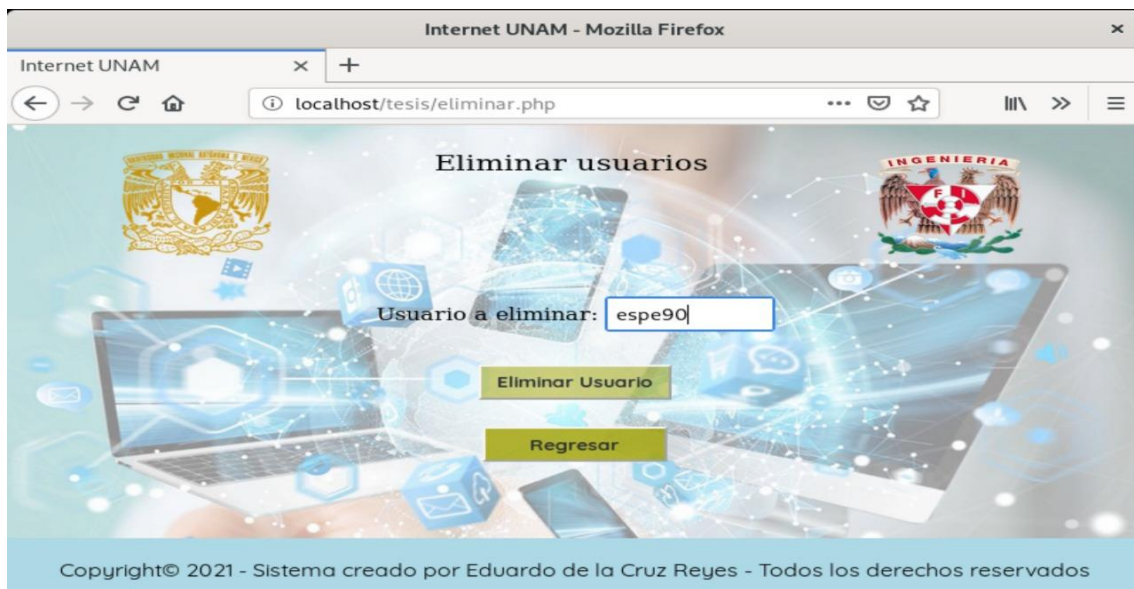


**Figura 158. Selección de usuario a eliminar.**  
Fuente: Elaboración propia <sup>133</sup>

<sup>132</sup> Captura de pantalla del Sistema

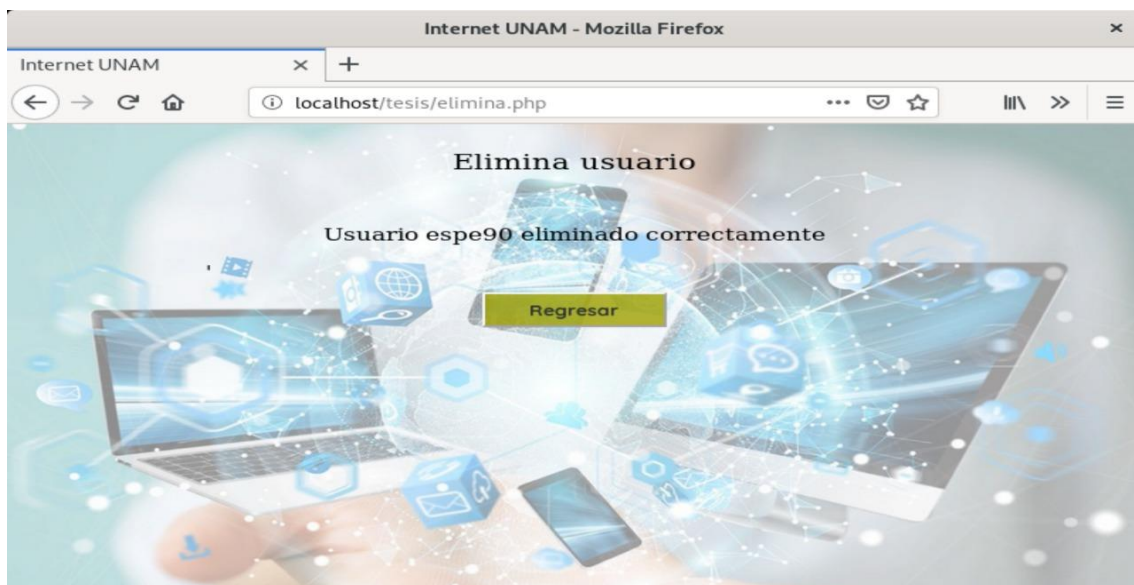
<sup>133</sup> Captura de pantalla del Sistema

Ingresamos el nombre de usuario y damos clic en Eliminar Usuario.



**Figura 159. Llenado de campo con usuario a eliminar.**  
Fuente: Elaboración propia <sup>134</sup>

Nos muestra un mensaje de que se eliminó correctamente.



**Figura 160. Eliminación de usuario exitoso.**  
Fuente: Elaboración propia <sup>135</sup>

<sup>134</sup> Captura de pantalla del Sistema

<sup>135</sup> Captura de pantalla del Sistema

Verificamos nuevamente en la tabla que se encuentra en el botón **Modificar usuario** y verificamos que el **id 14** ya no se encuentra.

1	EDUARDO	DE LA CRUZ	REYES	310033367	8	Ingeniería en Computación	2021-04-22 20:16:20	edyComputacion
2	SEBASTIAN	DE LA CRUZ	RESENDIZ	310033369	2	Ingeniería en Computación	2021-03-22 17:35:30	sebas
3	ALEXANDRA	RESENDIZ	NAZARIO	310033389	9	Ingeniería Civil	2021-04-09 18:46:35	aleFI
4	FRANCISCO	DE LA CRUZ	REYES	310033380	10	Ingeniería Civil	2021-04-09 18:47:04	paco40
5	CRUZ SERGIO	AGUILAR	DIAZ	084350262	4	Ingeniería en computacion	2021-04-09 18:40:07	sergio20
9	PATRICIA	DE LA CRUZ	REYES	310033331	7	Ingeniería en Sistemas Biomédicos	2021-04-09 19:56:28	p4t1UNAM
10	TERESA	ORTEGA	REYES	310033312	4	Ingeniería Ambiental	2021-04-09 18:23:26	tesa
16	DANIEL	GÓMEZ	SALGADO	302891291	8	Ingeniería en Computación	2021-04-23 15:14:05	dan#131

Copyright© 2021 - Sistema creado por Eduardo de la Cruz Reyes - Todos los derechos reservados

**Figura 161. Comprobación de usuario eliminado.**  
Fuente: Elaboración propia <sup>136</sup>

Daremos clic en el botón **Cerrar Sesión**

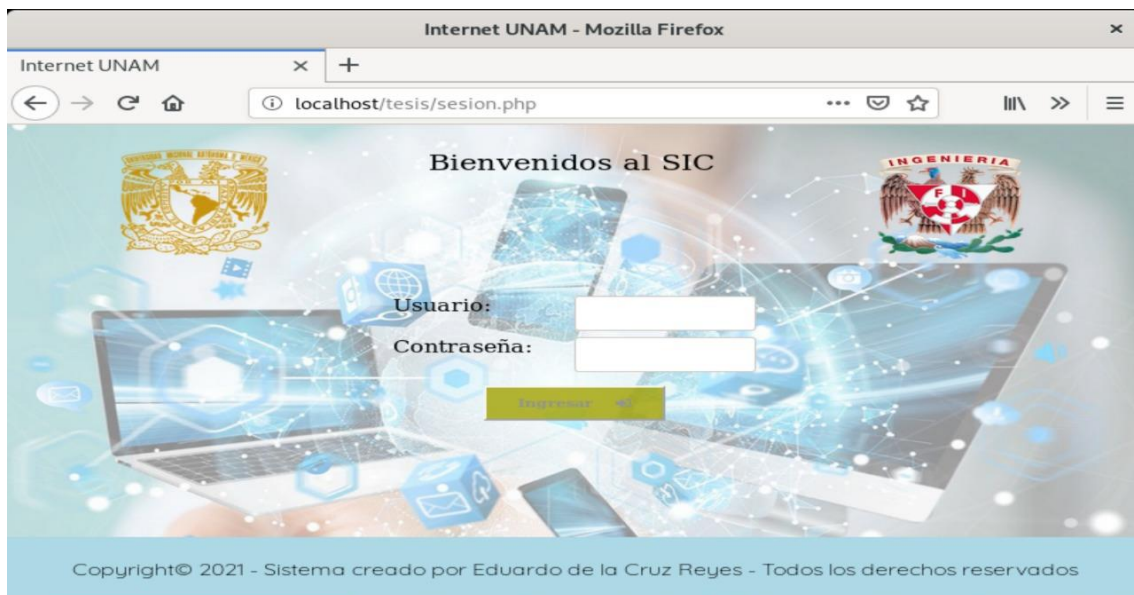


**Figura 162. Menú inicial.**  
Fuente: Elaboración propia <sup>137</sup>

<sup>136</sup> Captura de pantalla del Sistema

<sup>137</sup> Captura de pantalla del Sistema

Esto nos dirigirá de nuevo a la página de inicio de sesión del sistema.



**Figura 163. Página de inicio de sesión.**  
Fuente: Elaboración propia <sup>138</sup>

---

<sup>138</sup> Captura de pantalla del Sistema

# **GLOSARIO**

### **Autenticación**

Se refiere al proceso por el cual se determina si un usuario tiene permiso para acceder a un determinado servicio de red del que quiere hacer uso. El proceso de autenticación se realiza mediante la presentación de una identidad y unas credenciales por parte del usuario que demanda acceso.

### **Autorización**

Se refiere a conceder servicios específicos (entre los que se incluye la “negación de servicio”) a un determinado usuario, basándose para ello en su propia autenticación, los servicios que está solicitando, y el estado actual del sistema.

### **API**

Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones. Las API permiten a los productos y servicios comunicarse con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero.

### **Base de datos relacional**

Colección de elementos de datos organizados en un conjunto de tablas formalmente descritas, se puede acceder a los datos o volver a montarlos de muchas maneras diferentes sin tener que reorganizar las tablas de la base.

### **Campo**

Cada una de las columnas de una tabla. Identifica una familia de datos.

### **CHAP**

Challenge Handshake Authentication Protocol o protocolo de autenticación por desafío mutuo (CHAP) se usa para verificar periódicamente la identidad del par utilizando un protocolo de enlace de tres vías. Esto se realiza en el establecimiento inicial del enlace y se puede repetir periódicamente. El principio distintivo de CHAP se basa en la protección que se brinda al evitar la transmisión de cualquier contraseña a través del enlace, en lugar de confiar en un proceso de desafío y respuesta que solo puede tener éxito si tanto el autenticador como los dispositivos autorizados están reconociendo un valor conocido como secreto.

### **Consulta**

Es una instrucción para hacer peticiones a una BD.

### **Crawler o Araña Web**

Un rastreador web, araña o bot de motor de búsqueda descarga e indexa contenido de todo Internet. El objetivo de un bot de este tipo es aprender de qué tratan (casi) todas las páginas web, para que la información pueda ser recuperada cuando se necesite. Se llaman "arañas web (crawlers)" porque rastrear es el término técnico para acceder automáticamente a un sitio web y obtener datos a través de un programa de software.

### **CSMA/CD**

Protocolo de transmisión que procede del inglés Carrier sense multiple access / Collision detection. La detección de colisiones (CD) es una mejora del protocolo CSMA. Este algoritmo crea un procedimiento que regula la forma en la que debe producirse la comunicación dentro de una red con un medio de transmisión compartido. La mejora introducida regula además cómo proceder en caso de colisiones, es decir, cuando dos o más participantes intentan enviar paquetes de datos simultáneamente a través del medio de transmisión (Bus) y estos interfieren entre sí.

### **Dato**

Un dato es la representación de una variable que puede ser cuantitativa o cualitativa, indican un valor que se le asigna a las cosas. Un dato por sí solo no puede demostrar demasiado, siempre se evalúa el conjunto para poder examinar los resultados. Para examinarlos, primero hay que organizarlos o tabularlos.

### **Daemon**

Tipo especial de proceso informático no interactivo, es decir, que se ejecuta en segundo plano en vez de ser controlado directamente por el usuario. Este tipo de programas continua en el sistema, es decir, que puede ser ejecutado en forma persistente o reiniciado si se intenta matar el proceso dependiendo de configuración del demonio y políticas del sistema. La palabra daemon viene de las siglas en inglés D.A.E.MON (Disk And Execution Monitor).

### **DCL - Data Control Language**

Es un lenguaje de programación para definir estructuras de datos, proporcionado por los sistemas gestores de bases de datos. Estos comandos permiten al Administrador del sistema gestor de base de datos, controlar el acceso a los objetos, es decir, podemos otorgar o denegar permisos a uno o más roles para realizar determinadas tareas. Los comandos para controlar los permisos son los siguientes: GRANT y REVOKE.

### **DDL – Data Definition Language**

Es un lenguaje de programación para definir estructuras de datos, proporcionado por los sistemas gestores de bases de datos. Este lenguaje permite a los programadores de un sistema gestor de base de datos, definir las estructuras



que almacenarán los datos así como los procedimientos o funciones que permitan consultarlos. Para definir las estructuras disponemos de tres sentencias: CREATE, ALTER y DROP.

### **Diccionario de Datos**

Un diccionario de datos es un conjunto de definiciones que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

### **DHCP**

El protocolo de configuración dinámica de host (en inglés: Dynamic Host Configuration Protocol, también conocido por sus siglas de DHCP) es un protocolo de red de tipo cliente/servidor mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo en una red para que puedan comunicarse con otras redes IP. Este servidor posee una lista de direcciones IP dinámicas y las va asignando a los clientes conforme estas van quedando libres, sabiendo en todo momento quién ha estado en posesión de esa IP, cuánto tiempo la ha tenido y a quién se la ha asignado después.

### **DML – Data Manipulation Language**

Es un lenguaje de programación para definir estructuras de datos, proporcionado por los sistemas gestores de bases de datos. Utilizando instrucciones de SQL, permite a los usuarios introducir datos para posteriormente realizar tareas de consultas o modificación de los datos que contienen las Bases de Datos.

Los elementos que se utilizan para manipular los datos, son los siguientes: SELECT, INSERT, UPDATE y DELETE.

### **DNS**

DNS (Domain Name System) es un sistema de nombres de dominio que le permite a los humanos y a las computadoras comunicarse más fácilmente. El DNS es un intermediario que hace coincidir los nombres con los números dentro de una lista específica. Es un sistema de bases de datos distribuidas en la red con la función principal de traducir la solicitud de ciertos nombres de host a números de IP específicos que las computadoras entiendan.

### **EAP**

El Protocolo de Autenticación Extensible (EAP) proporciona el marco para que un dispositivo autentique cuando conecta con la red. EAP protege las redes privadas virtuales (VPN) seguras contra aquellos que realizan ataques mediante diccionario y adivinan contraseñas. Ofrece más ventajas que el Protocolo de autenticación de contraseñas (PAP) y el protocolo de autenticación de reconocimiento de identificación (CHAP).

### **Estándar IEEE 802.11n**

802 es un estándar establecido por la IEEE ("Institute of Electrical and electronics Engineers" o en español "Instituto de Ingenieros en Electricidad y Electrónica") WiFi en el cual según el estándar que corresponda se relaciona una velocidad de transmisión de datos. Para el cual, 802.11n funciona con conexiones de hasta 600mbps, opera en la banda de 2.4GHz y 5GHz,

### **Fork (del sistema)**

Una bifurcación o fork, cuando se aplica en el contexto de un lenguaje de programación o un sistema operativo, hace referencia a la creación de una copia de sí mismo por parte de un programa, que entonces actúa como un "proceso hijo" del proceso originario, ahora llamado "padre".

### **Framework**

Un marco o entorno de trabajo es lo que se conoce como framework. Se trata de una estructura definida para desarrollar y gestionar un software, es decir, un entorno que facilita la programación de una aplicación o software. Ayuda a que el programador separe en la aplicación desarrollada, la gestión de los datos, las operaciones, y la presentación. Pueden ser utilizados para programar con diferentes lenguajes de programación, aunque bien es cierto que en general suelen especializarse en alguno o algunos en concreto, incluyendo funciones especiales para los mismos.

### **Guiones o scripts**

Son un conjunto de instrucciones, que ejecutadas de forma ordenada, realizan operaciones avanzadas o mantenimiento de los datos almacenados en la BD.

### **GPG**

Sistema multiplataforma de gestión de llaves utilizado para el cifrado y firma digital de mensajes y archivos. Es un software libre liberado bajo la licencia GPL e implementa el estándar OpenPGP. Combina criptografía simétrica o de clave privada y criptografía asimétrica o de clave pública.

### **Hardening**

(Palabra en inglés que significa endurecimiento) en seguridad informática es el proceso de asegurar un sistema mediante la reducción de vulnerabilidades en el mismo, esto se logra eliminando software, servicios y usuarios innecesarios en el sistema; así como cerrando puertos que tampoco estén en uso.

### **HotSpot**

Se trata del acceso de cobertura que se ofrece a través de un dispositivo para que haya una comunicación inalámbrica, sin cables, wifi. Una de sus principales peculiaridades es que para tener dicho acceso se pide la identificación del

dispositivo que hace la función de “invitado” o “cliente”. Mediante un usuario y contraseña, se puede activar un hotspot fácilmente.

## **HOST**

El término host o anfitrión se usa en informática para referirse a las computadoras u otros dispositivos (tabletas, móviles, portátiles) conectados a una red que proveen y utilizan servicios de ella. De forma genérica, podemos decir que un anfitrión es todo equipo informático que posee una dirección IP y que se encuentra interconectado con uno o más equipos y que funciona como el punto de inicio y final de las transferencias de datos.

## **HTTP**

Las siglas HTTP, acrónimo de Hypertext Transfer Protocol, es un protocolo de transferencia de hipertexto. En otras palabras, HTTP es un protocolo de comunicación que define la manera en que se comunican un dispositivo cliente y un servidor conectados a través de la web. Es un protocolo que permite acceder a una página web cuando accedemos a su dirección en la barra de búsqueda, y también es el protocolo que le permite descifrar la información de la web que visitas y mostrártela en la pantalla. Sin embargo, no se encuentra protegido de forma segura, por lo tanto el protocolo HTTP no cuentan con un certificado SSL

## **HTTPS**

El protocolo HTTPS es una evolución del protocolo HTTP diseñada para garantizar la seguridad de las comunicaciones en HTTP. Lo que hace este protocolo es combinar el protocolo HTTP con el protocolo SSL/TLS (Secure Socket Layer / Transport Layer Security) para otorgarle al protocolo HTTP una capa de protección que no tiene en su versión básica.

## **Índice**

Es una estructura que almacena los campos clave de una tabla, organizándolos para hacer más fácil encontrar y ordenar los registros.

## **Informe**

Es un listado ordenado de los campos y registros seleccionados en un formato fácil de leer. Por ejemplo, un informe de las facturas impagadas del mes de enero ordenadas por nombre de cliente.

## **ISP**

Un proveedor de servicios de Internet (ISP) es una entidad que ofrece acceso a Internet mediante diferentes métodos de conexión (y velocidades) como: conexión telefónica, módem por cable, DSL, acceso inalámbrico o interconexiones dedicadas de alta velocidad. Un ISP puede ser una empresa de telecomunicaciones o estructuras independientes de menor tamaño que ofrecen

servicios como acceso seguro y VPN, funciones de búsqueda, correo electrónico, almacenamiento de datos remoto y otros tipos de beneficios que pueden ser exclusivos para sus clientes.

## **JSON**

JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

## **Kubernetes**

Plataforma portable y extensible de código abierto para administrar cargas de trabajo y servicios. Facilita la automatización y la configuración declarativa, para ofrecer un entorno de administración centrado en contenedores. Orquesta la infraestructura de cómputo, redes y almacenamiento para que las cargas de trabajo de los usuarios no tengan que hacerlo. Esto ofrece la simplicidad de las Plataformas como Servicio (PaaS) con la flexibilidad de la Infraestructura como Servicio (IaaS) y permite la portabilidad entre proveedores de infraestructura.

## **KVM**

La máquina virtual basada en el kernel (KVM) es una tecnología de virtualización de open source integrada a Linux. KVM puede convertir a Linux en un hipervisor que permite que una máquina de host ejecutar entornos virtuales múltiples y aislados llamados máquinas virtuales (VM) o huéspedes.

## **LDAP**

LDAP son las siglas de Protocolo Ligero de Acceso a Directorio, o en inglés Lightweight Directory Access Protocol. Se trata de un conjunto de protocolos de licencia abierta que son utilizados para acceder a la información que está almacenada de forma centralizada en una red. Este protocolo se utiliza a nivel de aplicación para acceder a los servicios de directorio remoto

## **Metadatos**

Los metadatos son datos acerca de los datos presentes en la base de datos, como por ejemplo, qué tipo de datos se van a almacenar (si son texto o números o fechas), que nombre se le da a cada dato (nombre, apellidos, fecha, precio, edad), cómo están agrupados los datos, cómo se relacionan, etcétera.

## **Mesos (Apache)**

Apache Mesos es un kernel administrador de clúster de código abierto desarrollado en la Universidad de California, Berkeley. Mesos se ejecuta en cada nodo del clúster y provee aplicaciones con API's para el manejo de recursos y planificación de tareas de todo el datacenter.

## **Microsoft SQL Server**

Es un sistema de gestión de base de datos relacional desarrollado como un servidor que da servicio a otras aplicaciones de software que pueden funcionar ya sea en el mismo ordenador o en otro ordenador a través de una red. Los servidores SQL Server suelen presentar como principal característica una alta disponibilidad al permitir un gran tiempo de actividad y una conmutación más rápida.

## **Modelo OSI**

El modelo de interconexión de sistemas abiertos, también llamado OSI (en inglés, open system interconnection) es el modelo de red descriptivo propuesto por la Organización Internacional para la Estandarización (ISO) en el año 1977 y aprobado en el año 1984. Es una normativa formada por siete capas que define las diferentes fases por las que deben pasar los datos para viajar de un dispositivo a otro sobre una red de comunicaciones. Constituye por tanto un marco de referencia para la definición de arquitecturas de interconexión de sistemas de comunicaciones.

## **NAS**

Un Network Access Server es un sistema que proporciona acceso a la red.

En algunos casos también se conoce como Remote Access Server (RAS) o Terminal Server. En general, NAS es un elemento que controla el acceso a un recurso protegido.

## **OpenLDAP**

OpenLDAP es una implementación libre y de código abierto del protocolo Lightweight Directory Access Protocol (LDAP) desarrollada por el proyecto OpenLDAP. OpenLDAP es una implementación del protocolo LDAP; en otras palabras, es una base de datos de propósito especial diseñada para almacenar directorios.

## **PAP**

Password Authentication Protocol o protocolo de autenticación de contraseña es un protocolo de autenticación de intercambio de dos vías que transmite una contraseña en texto sin formato. Se realiza durante el establecimiento inicial del enlace. Una vez completada la fase de enlace de enlace, el interlocutor envía repetidamente el nombre de usuario y la contraseña al autenticador hasta que

se confirma la autenticación o se termina la conexión. Simula eficazmente la operación de inicio de sesión en la que las contraseñas de texto sin formato se utilizan para establecer el acceso a un host remoto. Los datos devueltos autenticados envían el nombre de usuario local y el password al autenticador.

## **PAM**

Por sus iniciales en inglés de Pluggable Authentication Modules o Módulos de Autenticación Conectables son los programas que permiten el acceso del usuario a un sistema utilizan una autenticación para verificar la identidad de cada uno. PAM usa una arquitectura modular, con complementos, que le da al administrador del sistema un buen grado de flexibilidad en la configuración de las políticas de autenticación para el sistema.

## **Product backlog**

El product backlog (o pila de producto) es un listado de todas las tareas que se pretenden hacer durante el desarrollo de un proyecto.

## **Procedimientos**

Son un tipo especial de script que están almacenados en la BD y forman parte de su esquema.

## **PostgreSQL**

También conocido como Postgres, es un sistema de gestión de bases de datos relacionales (RDBMS) libre y de código abierto (Open Source). Una característica interesante de PostgreSQL es el control de concurrencias multiversión; o MVCC por sus siglas en inglés. Este método agrega una imagen del estado de la base de datos a cada transacción, permitiendo el uso de transacciones eventualmente consistentes.

## **PGP**

PGP, o Pretty Good Privacy, utiliza una combinación de métodos de encriptación tales como hashing, compresión de datos, criptografía de clave simétrica y criptografía de clave pública para mantener los datos seguros. Este proceso se puede utilizar para encriptar archivos de texto, correos electrónicos, archivos de datos, directorios y particiones de disco.

## **RBAC**

El control de acceso basado en roles (RBAC) es una función de seguridad para controlar el acceso de usuarios a tareas que normalmente están restringidas al superusuario. Mediante la aplicación de atributos de seguridad a procesos y usuarios, RBAC puede dividir las capacidades de superusuario entre varios

administradores. La gestión de derechos de procesos se implementa a través de privilegios. La gestión de derechos de usuarios se implementa a través de RBAC.

### **Refactoring**

El proceso de refactoring es un proceso sistemático, en el cual vamos a conseguir mejorar nuestro código sin crear nuevas funcionalidades. La idea es partir de código que está sucio y mejorarlo para conseguir que sea un código más limpio, que tenga un diseño más simple, que sea más fácil de arreglar y de modificar.

### **Registro (en una base de datos)**

Corresponde a cada una de las filas de la tabla. También se llaman tuplas.

### **Registro (en otros servicios)**

Suele incluir aspectos como la identidad del usuario, la naturaleza del servicio prestado, y cuándo empezó y terminó el uso de dicho servicio.

### **Remote Authentication Dial-In User Service**

Se refiere a un protocolo de establecimiento de una red que proporciona la administración centralizada de la autenticación, de la autorización, y de las estadísticas (AAA o triple - a) para los usuarios que conectan y utilizan un servicio de red. Un servidor de RADIUS regula el acceso a la red verificando la identidad de los usuarios a través de las credenciales de la clave ingresadas.

### **Shell**

En informática, el shell o intérprete de órdenes o intérprete de comandos es el programa informático que provee una interfaz de usuario para acceder a los servicios del sistema operativo.

### **SMTP**

El protocolo para transferencia simple de correo (en inglés Simple Mail Transfer Protocol o SMTP) es un protocolo de red utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos. El funcionamiento de este protocolo se da en línea, de manera que opera en los servicios de correo electrónico.

### **Sprint**

Sprint es el nombre que va a recibir cada uno de los ciclos o iteraciones que vamos a tener dentro de dentro de un proyecto Scrum. En cada Sprint o cada ciclo de trabajo lo que vamos a conseguir es lo que se denomina un entregable o incremento del producto, que aporte valor al cliente.

## **Sprint Review**

El Sprint Review es una reunión de colaboración donde se busca "feedback" de todos los presentes fundamentalmente para crear transparencia sobre el incremento de producto y permitir la adaptación del "Product Backlog" el "Release Plan" si fuera el caso. Debe contar con usuarios, interesados, clientes y otros participantes además del Product Owner y el Equipo de Desarrollo o que puedan dar "feedback" sobre el producto.

## **Sprint planning**

La reunión de planificación del Sprint sigue un flujo y tiene un límite de tiempo de hasta ocho horas para un Sprint de un mes. En este evento es creado por el trabajo colaborativo de todo el Equipo Scrum (Scrum Master, Product Owner y el Equipo de Desarrollo). Para Sprints más cortos, el evento suele ser más corto. El Scrum Master asegura que el evento se lleve a cabo y que los asistentes comprendan su propósito. Esa es la cosa más importante, porque sino el equipo perderá su motivación y su objetivo.

## **Stakeholders**

Se les designa a todos los que tienen el interés o la porción en el proyecto. Ellos pueden ser los dirigentes directos o, por ejemplo, las personas que financian el proyecto.

## **SQL (Structured Query Language)**

Se trata de un lenguaje estándar de bases de datos, el cual es usado para crear, mantener y recuperar bases de datos relacionales. SQL también te permite insertar, actualizar y eliminar registros de una base de datos, así como crear base de datos, para después agregar tablas dentro de ellas. De igual manera es posible crear vistas y establecer permisos en tablas, procedimientos y vistas.

## **SQLite**

Herramienta de software libre, que permite almacenar información en dispositivos empujados de una forma sencilla, eficaz, potente, rápida y en equipos con pocas capacidades de hardware, como puede ser una PDA o un teléfono celular. SQLite soporta desde las consultas más básicas hasta las más complejas del lenguaje SQL, y lo más importante es que se puede usar tanto en dispositivos móviles como en sistemas de escritorio, sin necesidad de realizar procesos complejos de importación y exportación de datos.

## **Tabla**

Es un conjunto de filas y columnas bajo un mismo nombre que representa el conjunto de valores almacenados para una serie de datos. Por ejemplo, la información de todos los clientes de una BD se almacenará en una tabla llamada CLIENTES.



## **TDD**

El Test-Driven Development o desarrollo dirigido por pruebas, es una práctica de programación que consiste en escribir primero las pruebas (generalmente unitarias), después escribir el código fuente que pase la prueba satisfactoriamente y, por último, refactorizar el código escrito. Con esta práctica se consigue entre otras cosas: un código más robusto, más seguro, más fácil de mantener y una mayor rapidez en el desarrollo.

## **Tipo de Dato**

El tipo de dato indica la naturaleza del campo. Se pueden tener datos numéricos, que son aquellos con los que se pueden realizar cálculos aritméticos (sumas, restas, multiplicaciones...), los datos alfanuméricos, que son los que contienen caracteres alfabéticos y números.

## **UDP**

El Protocolo de datagrama de usuario (UDP) es un protocolo ligero de transporte de datos que funciona sobre IP. Proporciona un mecanismo para detectar datos corruptos en paquetes, pero no intenta resolver otros problemas que surgen con paquetes, como cuando se pierden o llegan fuera de orden. Por eso, a veces UDP es conocido como el protocolo de datos no confiable.

## **Vista**

Se obtienen al guardar una consulta de una o varias tablas. De esta forma se obtiene una tabla virtual, es decir, no está almacenada en los dispositivos de almacenamiento de la computadora, aunque sí se almacena su definición.

## **VLAN**

Una VLAN, acrónimo de virtual LAN o Red de Área Local Virtual, es una tecnología para crear redes lógicas independientes dentro de una misma red física. Una VLAN está formada por dos o más dispositivos, que se comportan como si estuviesen conectados al mismo conmutador, aunque se encuentren físicamente enlazados a diferentes switches de la misma red de área local. Cada VLAN individual recibe su propio dominio de difusión (broadcast), de manera que si un dispositivo envía una difusión dentro de la VLAN, todos los demás participantes de ese segmento (y solo esos) reciben el mensaje. La difusión no se transmite más allá de los límites de la red virtual.

## **VoIP**

VoIP (del inglés voice over IP o voz sobre IP en español) es la transmisión de voz y contenido multimedia a través de redes de datos del Protocolo de Internet (IP). Un sistema de VoIP consiste básicamente en una serie de puntos finales que pueden ser teléfonos VoIP, teléfonos móviles, navegadores habilitados para

VoIP en las computadoras, etc. y también una red IP sobre la que se transportan los datos de los paquetes.

## **VPS**

Del inglés Virtual Private Server o Servidor Privado Virtual, es un servicio de alojamiento web que se obtiene dividiendo un servidor físico en varios servidores virtuales, haciendo que cada uno de ellos cuente con recursos dedicados y esté aislado de los demás. A nivel operativo, un Servidor Privado Virtual funciona igual que otros servicios de hosting web, ofreciendo un espacio conectado a internet de forma permanente al que podemos subir los contenidos de nuestra web para que otras personas puedan acceder a ellos.

## **VMWare (Virtualización)**

Es un sistema de virtualización por software, es decir, un programa que simula un sistema físico con unas características de hardware determinadas. Cuando se ejecuta el programa proporciona un ambiente de ejecución similar a todos los efectos a un computador físico, con CPU, BIOS, tarjeta gráfica, memoria RAM, tarjeta de red, sistema de sonido, conexión USB, disco duro, etcétera.

## **wifi**

Tecnología de red inalámbrica a través de la cual los dispositivos, como computadoras (portátiles y de escritorio), dispositivos móviles (teléfonos inteligentes y accesorios) y otros equipos (impresoras y videocámaras), pueden interactuar con Internet. Permite que estos dispositivos, entre tantos otros, intercambien información entre sí y establezcan, de esta manera, una red.

## **XEN (virtualización)**

XEN es una herramienta de virtualización que se ejecuta por debajo del sistema operativo y actúa como hypervisor del mismo. Es decir, si el sistema operativo actúa como supervisor de los programas que se ejecutan sobre él, XEN supervisa al supervisor y por lo tanto está en una capa inferior. XEN llama también a las máquinas virtuales dominios.

# **BIBLIOGRAFÍA**

Da Costa Carballo. (2002). Linux Versus Windows. Revista General de Información y Documentación, 12, 8

Guerrero Rosas. (2006). Bases de datos para la administración. México. Pearson Prentice Hall.

López Menéndez de Jiménez. (2015). Metodologías Ágiles de Desarrollo de Software Aplicadas a la Gestión de Proyectos Empresariales. REVISTA TECNOLÓGICA, 8, 6.

Meza Badillo. (S.F). Apuntes Sistema de Telecomunicaciones. CDMX: Sin Editorial.

Macías Ríos, Rangel Licea. (2009). Apuntes de la asignatura de Redes de Datos I y Redes de Datos II. Facultad de Ingeniería. UNAM

Mark A., Rick McDonald y Antoon W. Aspectos Básicos de networking. Guía de estudio de CCNA Exploration 2008, Pearson Educación S.A Cisco Press.

Orjuela Duarte, Rojas C., (Junio, 2008). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo. Avances en Sistemas e Informática, 5, 12.

# **MESOGRAFÍA**

## **(Referencias Electrónicas)**

¿Qué es Debian GNU/Linux? (s.f).  
<https://www.debian.org/releases/wheezy/ia64/ch01s03.html.es>

Ingeniería de Software. (s.f).  
<https://ingenieriadesoftwaretdea.weebly.com/diccionario-de-datos.html>

Aston, B. (2021, 15 de Enero). 9 Ejemplos De Metodología De Un Proyecto, Simplificados. dpm. <https://thedigitalprojectmanager.com/es/metodologias-gestion-proyectos-simplificadas/#methodology>

¿Qué es NoSQL? (s.f.). AWS. <https://aws.amazon.com/es/nosql/>

¿Qué es una base de datos relacional? (s.f.). AWS.  
<https://aws.amazon.com/es/relational-database/>

Isac. (s.f.). Debian vs Ubuntu. LinuxAdictos.  
<https://www.linuxadictos.com/debian-vs-ubuntu.html>

¿Por qué escoger software libre? (s.f.). Universidad de la Laguna.  
<https://www.ull.es/servicios/osl/por-que-escoger-software-libre/>

Categorías de software libre y software que no es libre. (s.f.). El sistema operativo GNU. <https://www.gnu.org/philosophy/categories.es.html>

¿Qué es un servidor? (2020, 15 de Septiembre). Digital Guide IONOS.  
<https://www.ionos.mx/digitalguide/servidores/know-how/que-es-un-servidor-un-concepto-dos-definiciones/#:~:text=Los%20servidores%20web%20m%C3%A1s%20populares,a%20trav%C3%A9s%20de%20una%20red>

Solano, Jose. (S.F). El modelo OSI.  
[http://dis.um.es/~lopezquesada/documentos/IES\\_1213/LMSGI/curso/xhtml/xhtml22/index.html](http://dis.um.es/~lopezquesada/documentos/IES_1213/LMSGI/curso/xhtml/xhtml22/index.html)

CEI - V - Sistemas operativos II. (S.F).  
<https://sites.google.com/site/sistemasoperativosv/home/2015-i/que-son-los-daemon>

¿Qué es una araña web (crawler)? | Cómo funcionan las arañas web. (s.f). CLOUDFLARE. <https://www.cloudflare.com/es-la/learning/bots/what-is-a-web-crawler/>

Gustavo, B. (2020, 10 de Diciembre). ¿Qué es DNS y cómo funciona? HOSTINGER TUTORIALES. <https://www.hostinger.mx/tutoriales/que-es-dns/>

Protocolo para transferencia simple de correo. (s.f.). WIKIPEDIA. [https://es.wikipedia.org/wiki/Protocolo\\_para\\_transferencia\\_simple\\_de\\_correo](https://es.wikipedia.org/wiki/Protocolo_para_transferencia_simple_de_correo)

Ventajas y desventajas de las metodologías Agile (ágiles). (2019, 11 de Septiembre). EBF. <https://ebf.com.es/blog/ventajas-y-desventajas-de-las-metodologias-agiles-y-su-aplicacion-en-el-trabajo/>

El modelo en cascada: desarrollo secuencial de software. (2019, 21 de Marzo). Digital Guide IONOS. <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/>

Kanban. (s.f.). Digital Guide IONOS. <https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/que-es-kanban/>

Eulises Ortiz, A. (2020, 16 de Junio). Ventajas de usar MariaDB sobre MySQL. Host Dime Blog. <https://www.hostdime.com.ar/blog/3-ventajas-de-usar-mariadb-sobre-mysql/>

Jimenez, L. (2019, 19 de Abril). Introducción a PPP: Métodos de autenticación PAP y CHAP. HUAWEI. <https://forum.huawei.com/enterprise/es/introducci%C3%B3n-a-ppp-m%C3%A9todos-de-autenticaci%C3%B3n-pap-y-chap/thread/521093-100235#:~:text=La%20autenticaci%C3%B3n%20PAP%20se%20realiza%20durante%20el%20establecimiento%20inicial%20del%20enlace.&text=El%20protocolo%20de%20autenticaci%C3%B3n%20por,de%20enlace%20de%20tres%20v%C3%ADas.>

Servidor FTP. (s.f.). <http://servidorftp.es/>

Germán González, D. (s.f.). Distribuciones Linux para servidores. Algunas opciones. Linux Adictos. <https://www.linuxadictos.com/distribuciones-linux-para-servidores-algunas-opciones.html>

Linux vs. Windows: cuadro comparativo. (2021, 14 de Enero). Digital Guide IONOS. <https://www.ionos.mx/digitalguide/servidores/know-how/linux-vs-windows-el-gran-cuadro-comparativo/>

Marín, R. (2019, 16 de Abril). Los gestores de bases de datos más usados en la actualidad. Revista digital INESEM. <https://revistadigital.inesem.es/informatica-y-tics/los-gestores-de-bases-de-datos-mas-usados/>

CSMA/CD: protocolo de transmisión anticollisiones. (2019, 2019, 7 de Mayo). Digital Guide IONOS. <https://www.ionos.mx/digitalguide/servidores/know-how/csmacd/>

Módulos de autenticación conectables (PAM, por las iniciales en inglés de Pluggable Authentication Modules). (s.f.). Fedora. [https://docs.fedoraproject.org/es-ES/Fedora/13/html/Security\\_Guide/sect-Security\\_Guide-Pluggable\\_Authentication\\_Modules\\_PAM.html](https://docs.fedoraproject.org/es-ES/Fedora/13/html/Security_Guide/sect-Security_Guide-Pluggable_Authentication_Modules_PAM.html)

Stakeholders Scrum / Las partes Interesadas en el Scrum. (s.f.). SCRUM. <https://es.scrum-time.com/infobase/stakeholders-scrum.php>

Requena Mesa, A. (2018, 19 de Diciembre). Qué es un Sprint de Scrum. OpenWebinars. <https://openwebinars.net/blog/que-es-un-sprint-scrum/>

Díaz Alcolea, C. (2019, 05 de Diciembre). Qué es Refactoring y cómo hacerlo paso a paso. <https://openwebinars.net/blog/que-es-refactoring-y-como-hacerlo-paso-paso/>

García Tamarit, Rafael. (2019, 27 de Febrero). ¿Qué es el backlog? MUY AGILE. <https://muyagile.com/que-es-el-backlog/>

Francia Huambachano, J. (2018, 4 de Septiembre). Sprint Review. Scrum.org The Home of Scrum. <https://www.scrum.org/resources/blog/sprint-review-spanish-edition>

Hernandez Sola, G. (2020, 16 de Febrero). ¿Qué es la Sprint Planning y Qué Flujo Sigue para el Equipo? Scrum.org The Home of Scrum. <https://www.scrum.org/resources/blog/que-es-la-sprint-planning-y-que-flujo-sigue-para-el-equipo>