



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE SITIO WEB PARA LA GESTIÓN DE ASIGNACIÓN DE
IPS UTILIZANDO API REST EN LENGUAJE PYTHON

CASTRO ARIZAGA CHRISTIAN GERMAN
INGENIERO DE SISTEMAS

MACHALA
2020



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE SITIO WEB PARA LA GESTIÓN DE
ASIGNACIÓN DE IPS UTILIZANDO API REST EN LENGUAJE
PYTHON

CASTRO ARIZAGA CHRISTIAN GERMAN
INGENIERO DE SISTEMAS

MACHALA
2020



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

DESARROLLO DE SITIO WEB PARA LA GESTIÓN DE ASIGNACIÓN DE IPS
UTILIZANDO API REST EN LENGUAJE PYTHON

CASTRO ARIZAGA CHRISTIAN GERMAN
INGENIERO DE SISTEMAS

MOROCHO ROMAN RODRIGO FERNANDO

MACHALA, 18 DE DICIEMBRE DE 2020

MACHALA
2020

Titulación-20201207

INFORME DE ORIGINALIDAD

1 %

INDICE DE SIMILITUD

1 %

FUENTES DE
INTERNET

0 %

PUBLICACIONES

0 %

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

1

archive.org

Fuente de Internet

<1 %

2

fr.slideshare.net

Fuente de Internet

<1 %

3

media.neliti.com

Fuente de Internet

<1 %

4

ddd.uab.cat

Fuente de Internet

<1 %

5

bestreferat.su

Fuente de Internet

<1 %

6

manualzz.com

Fuente de Internet

<1 %

7

doczz.es

Fuente de Internet

<1 %

8

genesys-ve.com

Fuente de Internet

<1 %

9

www.clubensayos.com

Fuente de Internet

<1 %

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, CASTRO ARIZAGA CHRISTIAN GERMAN, en calidad de autor del siguiente trabajo escrito titulado DESARROLLO DE SITIO WEB PARA LA GESTIÓN DE ASIGNACIÓN DE IPS UTILIZANDO API REST EN LENGUAJE PYTHON, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 18 de diciembre de 2020



Christian Castro

CASTRO ARIZAGA CHRISTIAN GERMAN
0704024298

DEDICATORIA

El presente trabajo va dedicado principalmente a Dios, por darme las fuerzas necesarias para realizar mis estudios y culminarlos.

A mis padres Marcos Castro y Rosa Arízaga por darme la vida y apoyarme incondicionalmente durante toda esta etapa. A mis hermanos que siempre estuvieron presentes para brindarme su ayuda cuando los necesitaba, por respetarme y esperar lo mejor para cumplir mis objetivos propuestos.

Sr. Christian German Castro Arízaga

AGRADECIMIENTO

Agradezco principalmente a Dios por darme su bendición para poder culminar la etapa académica y poder alcanzar la etapa profesional.

A mis padres y hermanos por siempre apoyarme en los estudios y compartirme sus conocimientos y valores para poder ser la persona que soy ahora.

También agradezco a mi tutor el Ing. Rodrigo Fernando Morocho Román por guiarme en el proceso de redacción de este informe.

Sr. Christian German Castro Arízaga

RESUMEN

El internet se ha vuelto un recurso muy importante en la mayoría de hogares, debido a que permite estar conectado con amigos y familiares, mediante los ordenadores o dispositivos móviles, a través de las diferentes redes sociales existentes. Para ello es necesario tener acceso a este servicio, el cual es proporcionado por los proveedores de internet, quienes se encargan de realizar su respectiva configuración.

Los routers cumplen un papel muy importante al momento de querer disponer del servicio de internet, ya que estos dispositivos son los que permiten la conexión dentro de una red informática y su configuración se debe realizar con mucha cautela, una ruta mal configurada puede causar la falla total de la red, para lo cual, los técnicos encargados ingresan al dispositivo, a través de comandos escritos en la terminal, o a su vez, utilizando herramientas de software que permiten observar de manera gráfica todas las configuraciones disponibles en el router que el cliente haya adquirido.

En la actualidad, existen una gran variedad de herramientas de software que permiten a los técnicos que trabajan en las empresas proveedoras de internet configurar las IP y rutas en los routers, para así poder realizar una correcta configuración de la red, la utilización de estos sistemas reduce el tiempo que normalmente ocupan en ajustar las configuraciones adecuadas, mejorando así su rendimiento y ofreciendo una gran ventaja sobre otras instituciones que ofrecen servicios similares.

Las empresas proveedoras de internet optan por buscar un sistema sencillo que sea amigable con los usuarios, que no siempre permiten al usuario realizar las configuraciones requeridas, y es aquí cuando se crea la idea de desarrollar su propia herramienta de software, para lo cual los desarrolladores tienden a usar APIs, que permitan la conexión con los routers y con esto realizar tareas específicas que los técnicos puedan necesitar para cumplir con su trabajo.

La finalidad de esta propuesta tecnológica es desarrollar un software que permita la gestión y asignación de IPs, mediante el uso de API, en el lenguaje de programación Python, de tal manera que se pueda configurar una topología de

red virtualizada en el software GNS3, de modo que exista una correcta conexión entre sus dispositivos. El sistema desarrollado posee funciones básicas, pero importantes al momento de configurar los routers, estas funciones son la asignación de IPs, la creación de rutas y habilitar o deshabilitar las interfaces de los routers. Además de la creación de usuarios con sus respectivos permisos para su configuración.

Para la creación de la aplicación web, se usó Flask, el cual es un framework que funciona del lado del back-end y está escrito en lenguaje Python, permite crear aplicaciones web de manera rápida y sencilla, y gracias a sus funciones que se ejecutan del lado del servidor se puede interactuar con el API RouterOS, que también está escrita en Python. Del lado del front-end se utilizó el framework Bootstrap, el cual permite crear interfaces web con CSS y Javascript, quienes se adaptan al tamaño del dispositivo en el que son visualizadas.

Palabras claves: Redes definidas por Software, GNS3, Mikrotik, API REST, Python.

ABSTRACT

The internet has become a very important resource in most homes, because it allows you to be connected with friends and family, through computers or mobile devices, through the different existing social networks. For this, it is necessary to have access to this service, which is provided by internet providers, who are in charge of carrying out their respective configuration.

Routers play a very important role when you want to have internet service, since these devices are the ones that allow the connection within a computer network and their configuration must be carried out with great caution, a badly configured route can cause failure total of the network, for which the technicians in charge enter the device through commands written in the terminal, or in turn, using software tools that allow to graphically observe all the available configurations in the router that the client has acquired.

At present, there are a great variety of software tools that allow technicians who work in internet provider companies to configure the IPs and routes in the routers, in order to make a correct network configuration, the use of these systems reduces the time they normally spend adjusting the appropriate settings, thus improving their performance and offering a great advantage over other institutions that offer similar services.

Internet provider companies choose to look for a simple system that is user-friendly, which does not always allow the user to make the required configurations, and it is here when the idea of developing their own software tool is created, for which developers tend to make use of APIs, which allow connection with routers and thereby perform specific tasks that technicians may need to do their job.

The purpose of this technological proposal is to develop a software that allows the management and assignment of IPs, through the use of APIs, in the Python programming language, in such a way that a virtualized network topology can be configured in the GNS3 software, of so that there is a correct connection between your devices. The developed system has basic functions, but important when configuring the routers, these functions are the assignment of IPs, the creation of

routes and enable or disable the interfaces of the routers. In addition to the creation of users with their respective permissions.

For the creation of the web application, Flask was used, which is a framework that works on the back-end side and is written in Python language, allows to create web applications quickly and easily, and thanks to its functions that are executed on the server side you can interact with the RouterOS API, which is also written in Python. On the front-end side, the Bootstrap framework was used, which allows creating web interfaces with CSS and Javascript, which adapt to the size of the device on which they are displayed.

Keywords: Software Defined Networks, GNS3, Mikrotik, REST API, Python.

CONTENIDO

DEDICATORIA	¡Error! Marcador no definido.
AGRADECIMIENTO	2
RESUMEN	3
ABSTRACT	5
INTRODUCCIÓN.....	11
1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	12
1.1. ÁMBITO DE APLICACIÓN: DESCRIPCIÓN DEL CONTEXTO Y HECHOS DE INTERES.....	12
1.2. ESTABLECIMIENTO DE REQUERIMIENTOS.....	13
1.3. JUSTIFICACIÓN DEL REQUERIMIENTO A SATISFACER.....	14
2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO	15
2.1. DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO	15
2.1.1. Capa Cliente (presentación).....	15
2.1.2. Capa Servidor.....	16
2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO	17
2.2.1. Metodología de desarrollo de software.....	17
2.2.2. Metodología SCRUM.....	17
2.2.3. Herramientas de desarrollo	18
2.3. OBJETIVOS DEL PROTOTIPO	20
2.3.1. Objetivo General	20
2.3.2. Objetivos Específicos	20
2.4. DISEÑO DEL PROTOTIPO.....	20
2.4.1. Diseño de interfaces.....	21
2.5. EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO	28
2.5.1. Pantalla principal.....	28
2.5.2. Inicio de sesión del router	28
2.5.3. Vista de direcciones IP del router.....	29
2.5.4. Ingreso de dirección IP del router	29
2.5.5. Modificación o eliminación de dirección IP del router.....	30
2.5.6. Vista de interfaces del router.....	30
2.5.7. Habilitar o deshabilitar interfaces del router	31
2.5.8. Vista de rutas del router	31
2.5.9. Ingreso de ruta al router.....	32
2.5.10. Modificación o eliminación de ruta del router.....	32

2.5.11.	Vista de usuarios del router	33
2.5.12.	Ingreso de usuario al router	33
2.5.13.	Modificación o eliminación de usuario del router	34
2.5.14.	Cambio de contraseña de usuario del router	34
3.	CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO	35
3.1.	PLAN DE EVALUACIÓN	35
3.1.1.	Preparación de herramientas	35
3.1.2.	Inicio de sesión	36
3.1.3.	Direccionamiento IP	36
3.1.4.	Administración de interfaces	36
3.1.5.	Creación de rutas	36
3.1.6.	Administración de usuarios	36
3.2.	RESULTADOS DE LA EVALUACIÓN	37
3.2.1.	Resultado de Inicio de sesión	37
3.2.2.	Resultado de Direccionamiento IP	37
3.2.3.	Resultado de Administración de interfaces	37
3.2.4.	Resultado de Creación de rutas	38
3.2.5.	Resultado de Administración de usuarios	38
3.3.	CONCLUSIONES	39
3.4.	RECOMENDACIONES	39

ÍNDICE DE FIGURAS

Figura 1. Arquitectura del prototipo	15
Figura 2: Descripción gráfica de la fundamentación teórica.....	17
Figura 3: Arquitectura de la plataforma web	18
Figura 4: Diseño de interfaces – Pantalla principal	21
Figura 5: Diseño de interfaces - Formulario de inicio de sesión en el router	21
Figura 6: Diseño de interfaces - Vista de direcciones IP del router	22
Figura 7: Diseño de interfaces - Ingreso de dirección IP al router	22
Figura 8: Diseño de interfaces - Modificación o eliminación de dirección IP del router	23
Figura 9: Diseño de interfaces - Vista de interfaces del router.....	23
Figura 10: Diseño de interfaces - Habilitar o deshabilitar interfaces del router.....	24
Figura 11: Diseño de interfaces - Vista de rutas del router	24
Figura 12: Diseño de interfaces - Ingreso de ruta al router	25
Figura 13: Diseño de interfaces - Modificación o eliminación de ruta del router	25
Figura 14: Diseño de interfaces - Vista de usuarios del router	26
Figura 15: Diseño de interfaces - Ingreso de usuario al router	26
Figura 16: Diseño de interfaces – Modificación o eliminación de usuario del router	27
Figura 17: Diseño de interfaces – Cambio de contraseña de usuario del router.....	27
Figura 18: Ejecución del prototipo - Pantalla principal	28
Figura 19: Ejecución del prototipo - Inicio de sesión del router	28
Figura 20: Ejecución del prototipo - Vista de direcciones IP del router.....	29
Figura 21: Ejecución del prototipo - Ingreso de dirección IP al router	29
Figura 22: Ejecución del prototipo - Modificación o eliminación de dirección IP del router.....	30
Figura 23: Ejecución del prototipo - Vista de interfaces del router	30
Figura 24: Ejecución del prototipo - Habilitar o deshabilitar interfaces del router	31
Figura 25: Ejecución del prototipo - Vista de rutas del router.....	31
Figura 26: Ejecución del prototipo - Ingreso de ruta al router	32
Figura 27: Ejecución del prototipo - Modificación o eliminación de ruta del router.....	32
Figura 28: Ejecución del prototipo - Vista de usuarios del router	33
Figura 29: Ejecución del prototipo - Ingreso de usuario al router	33
Figura 30: Ejecución del prototipo - Modificación o eliminación de usuario del router	34
Figura 31: Ejecución del prototipo - Cambio de contraseña de usuario del router.....	34
Figura 32: Topología virtualizada para las pruebas del prototipo tecnológico.....	35

ÍNDICE DE TABLAS

Tabla 1: Resultado de inicio de sesión	37
Tabla 2: Resultado de Direccionamiento IP	37
Tabla 3: Resultado de Administración de interfaces.....	37
Tabla 4: Resultado de Creación de rutas.....	38
Tabla 5: Resultado de Administración de usuarios	38

INTRODUCCIÓN

La gestión de asignación de IPs se ha vuelto una tarea del día a día para las empresas proveedoras de internet, puesto que estas se encargan de mantener a sus clientes conectados en la gran red global de redes conocida como internet. Entre las configuraciones se destacan el direccionamiento IP, el enrutamiento de los dispositivos, y la creación de usuarios para otorgar los permisos adecuados a los encargados de realizarlas.

Algunos de los inconvenientes que presentan las empresas proveedoras de internet, es la capacitación que se le debe dar a los técnicos, esto se debe a que algunas herramientas de software para la configuración de routers son complejas de utilizar, incluso poseen más opciones de las requerida por los técnicos, por lo que algunas empresas optan por desarrollar su propio sistema para la configuración de routers.

Un API REST no es más que un estándar lógico para la creación de servicios profesionales, los cuales utilizan los sistemas para optimizar recursos al utilizar los disponibles por el API y así crear una aplicación web eficiente [1].

Capítulo 1: Este capítulo explica la funcionalidad que se le puede dar al sistema para la gestión de asignación de IP, se describe también la problemática, hechos de interés, requerimientos del prototipo y la justificación, describiendo por qué están importante dar solución a la necesidad planteada.

Capítulo 2: Describe la fundamentación teórica, los objetivos, el diseño del prototipo y la ejecución y ensamblaje del mismo.

Capítulo 3: Se ejecuta un plan para evaluar el prototipo tecnológico, de tal forma que se obtienen resultados, los cuales ayudan a establecer las conclusiones y recomendaciones acerca del prototipo presentado.

1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS

1.1. ÁMBITO DE APLICACIÓN: DESCRIPCIÓN DEL CONTEXTO Y HECHOS DE INTERES

El internet se ha convertido en un recurso valioso en muchos hogares, a tal punto que llega a ser casi tan necesario como cualquier servicio básico, ya que permite la conexión virtual entre todos sus usuarios, los cuales pueden intercambiar información sin la necesidad de estar cerca uno del otro [2].

Los proveedores de internet cumplen un papel muy importante a la hora de ofrecer sus servicios, pues ellos se encargan de la distribución, configuración y mantenimiento de la red para sus clientes, generándoles satisfacción por el buen servicio que brindan [3].

En la actualidad, muchas aplicaciones web utilizan API REST para crear sus servicios, ejemplos de estos son el inicio de sesión o consulta de información. Se utilizan operaciones como GET, POST, PUT y DELETE, para consulta, creación, edición y eliminación de los recursos de un sitio web [4]. Las aplicaciones SDN (redes definidas por software), pueden obtener información de red de los dispositivos, lo cual permite actualizar el estado de la red en caso de ser requerido [5].

La presente propuesta tecnológica tiene como propósito desarrollar un sitio web para la asignación de IP mediante el uso de un API REST, haciendo uso de Python como lenguaje principal, el framework Flask del lado del servidor, que también utiliza el lenguaje Python, el framework Bootstrap para la visualización de la página y Javascript para las funciones que utilizan AJAX.

1.2. ESTABLECIMIENTO DE REQUERIMIENTOS

En las empresas proveedoras de internet, mantener a sus clientes conectados a la red es de gran importancia, esta tarea es realizada por los técnicos, los cuales se encargan de configurar los routers mediante la asignación de direcciones IP y creación de rutas, además de activar las interfaces respectivas para su correcto funcionamiento. Dichas tareas pueden realizar a través de una aplicación SDN, la cual permite resolver gestionar la red mediante la capacidad de programación de las redes [6].

La aplicación SDN para la gestión de asignación de IP, permitirá a los técnicos inexpertos realizar el enrutamiento de una topología de red pequeña en cuestión de minutos, esto ofrecería una ventaja competitiva contra los demás proveedores de internet, ya que los problemas que presenta el cliente se resolverán de manera rápida y eficiente.

Los técnicos involucrados en la configuración de las redes tendrán que asignarle una IP al dispositivo de red por la cual el sistema podrá conectarse, claro está que dicha conexión se logra utilizando también su respectivo usuario y contraseña. De esta manera, se tendrá acceso a la información del dispositivo, así la creación, edición o eliminación de IP o rutas no serán problema alguno.

1.3. JUSTIFICACIÓN DEL REQUERIMIENTO A SATISFACER

El objetivo de la presente propuesta tecnológica es desarrollar un sitio web para la gestión de asignación de IPs, lo cual se hará mediante el uso de API REST y se utilizará el lenguaje Python. Esto permitirá asignar IP a las interfaces del dispositivo de red, además de la creación de rutas estáticas a través de las cuales se podrán enviar paquetes de datos entre un router y otro.

Una aplicación SDN, permite controlar las configuraciones de un dispositivo de red, implementando así un enrutamiento estático que los proveedores de servicio de internet proporcionan a la red de sus clientes para satisfacer la demanda de uso de internet [7].

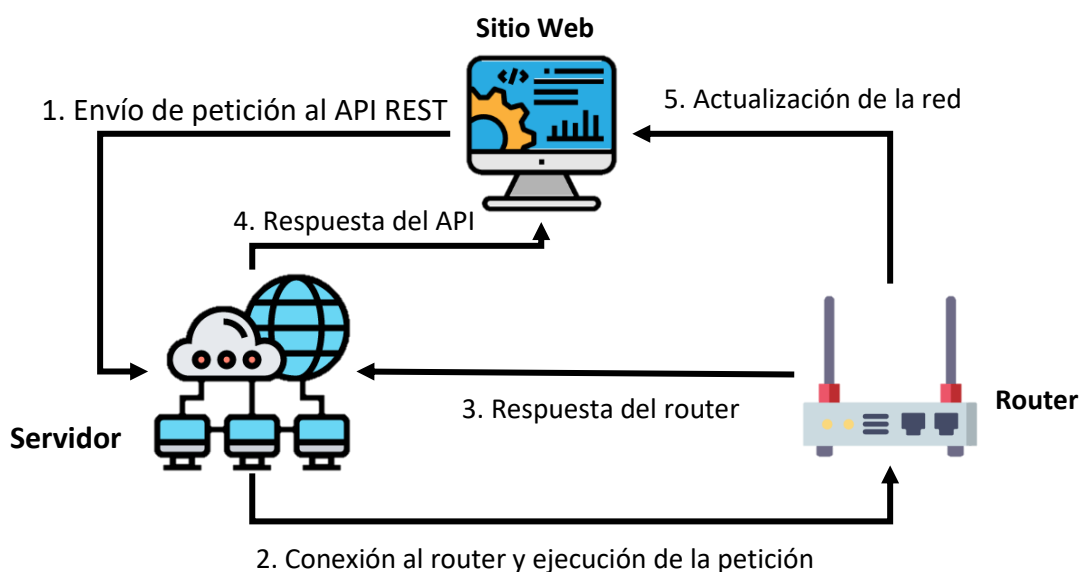
Las empresas proveedoras de internet, muchas veces se ven afectadas por la acumulación de los constantes reclamos de sus clientes, los cuales muchas veces son causas de la desconfiguración de los dispositivos de red, debido a que intentan resolver problemas de conectividad por sus propios medios, por esta razón, tener disponible una aplicación SDN, permitirá a los técnicos actuar de manera inmediata para la resolución del problema de red.

2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO

2.1. DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO

La arquitectura del prototipo tecnológico está basada en el modelo cliente-servidor, el cual permite al usuario acceder a las configuraciones del dispositivo de red para poder realizar una tarea en específica, sean estas la creación, edición, visualización o eliminación de la información de la red.

Figura 1. Arquitectura del prototipo



Fuente: Elaboración propia

En la **Figura 1** se puede apreciar el funcionamiento del prototipo.

1. El sitio web permite enviar una petición al API REST.
2. La petición es recibida por el router y ejecutada.
3. Se envía una respuesta del router al API.
4. El API envía una respuesta al sitio web, el usuario verifica si su petición se cumplió satisfactoriamente o no.
5. La petición alterará o no la configuración del router, lo cual será notable en la red del cliente.

2.1.1. Capa Cliente (presentación)

Esta capa posee interfaces que el usuario puede usar para navegar por el sistema y provee acceso a la capa del servidor [8].

Según [9], “es considerada la capa final, capa cliente o capa de presentación la cual puede ser presentada en navegadores u otro tipo de formato, ciertos

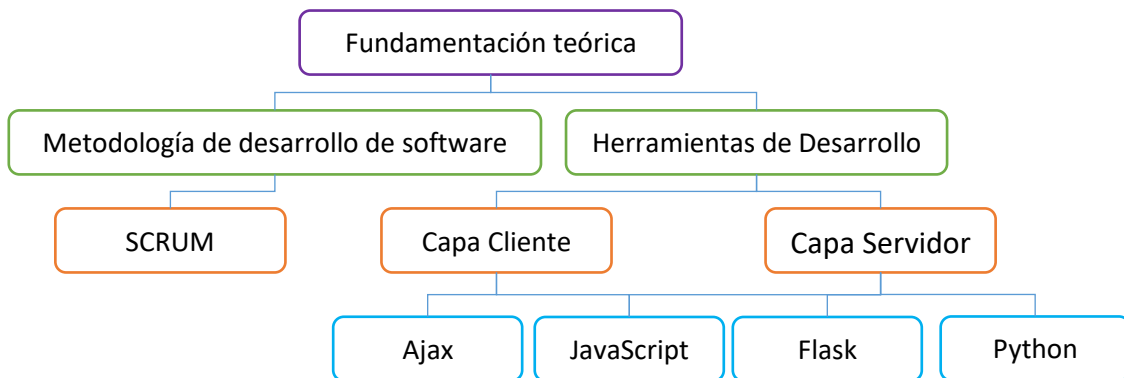
desarrolladores limitan sus sistemas a uno o dos navegadores, la comunicación es realizada por medio del protocolo HTTP”.

2.1.2. Capa Servidor

Esta capa se encarga de recibir las peticiones hechas por el cliente [10], dichas peticiones son realizadas a través del protocolo HTTP, quien se encarga de su comunicación y es considerado como un protocolo sin estado ya que no guarda información sobre las conexiones anteriores [11], HTTP hace uso de los métodos GET, POST, PUT y DELETE. De esta manera el cliente envía su petición y recibe una respuesta por parte del servidor, confirmando si se realizó satisfactoriamente la acción requerida [12].

2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO

Figura 2: Descripción gráfica de la fundamentación teórica



Fuente: Elaboración propia

2.2.1. Metodología de desarrollo de software

El uso de una metodología de desarrollo de software ayuda a que un proyecto sea organizado, ya que cuenta con estándares que garantizan el éxito de los resultados [13], desde el punto de vista de [14] las metodologías son una serie de fases facilitan el modelado y desarrollo de una aplicación, garantizando así un producto confiable, funcional y de buena calidad.

2.2.2. Metodología SCRUM

La metodología Scrum ofrece múltiples ventajas al momento de desarrollar un proyecto, facilita la colaboración en equipo, conformación de equipos de trabajo flexible y adaptativos [15], gestiona de manera eficiente las interacciones y mantiene el control en todo tipo de proyectos [16]. La participación activa de los usuarios en el proceso de desarrollo de software ayuda mejor a una planificación y análisis del proyecto [17].

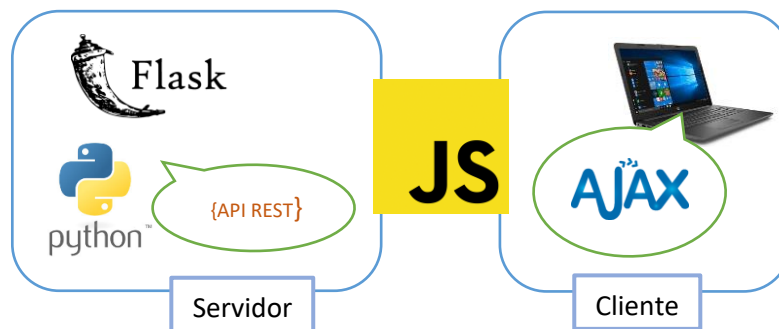
Dentro de la metodología se encuentran varias etapas que se deben cumplir para garantizar el éxito del proyecto a desarrollar, estas etapas son [18]:

- Reunión de planificación: El equipo se reúne para definir la meta que se desea alcanzar durante esa iteración y las actividades necesarias para alcanzarlas.
- Scrum Diario: Reunión donde el equipo busca mantener la incertidumbre del proyecto al mínimo y colaborar entre sus miembros.

- Revisión del Sprint: El equipo presenta sus avances y el dueño del producto revisa sus aportaciones para aceptarlas o agregar nuevas tareas al Backlog.
- Retrospectiva del Sprint: Reunión en donde se busca analizar aspectos relacionados con la finalización de la interacción, de esta manera se planean las posibles mejoras para trabajarlas en el próximo Sprint.

2.2.3. Herramientas de desarrollo

Figura 3: Arquitectura de la plataforma web



Fuente: Elaboración propia

2.2.3.1. Ajax

Es un conjunto de técnicas de desarrollo web que permite crear aplicaciones interactivas que se ejecutan en el cliente, manteniendo la comunicación asíncrona con el servidor [19], de esta manera se pueden realizar cambios sobre las páginas sin la necesidad de recargarlas [20].

2.2.3.2. JavaScript

JavaScript es un lenguaje basado en prototipos que utilizan los desarrolladores para efectuar cambios en un documento HTML directamente del lado del cliente (Navegador), evitando así las peticiones al servidor [21] [22].

2.2.3.3. Framework

El uso de frameworks es imprescindible en el desarrollo de software, estos incluyen lenguaje, bibliotecas, y soportes, los cuales facilitan el desarrollo de aplicaciones web [23].

2.2.3.4. Flask

El framework Flask permite la creación de microservicios basados en Python, como métodos independientes de un script y asociados a una interfaz, los cuales pueden ser invocados desde la capa de visualización [24]. HTML, se puede utilizar como lenguaje de programación orientado a la web, debido a que la implementación de Flask permite interpretar e incluir plantillas HTML con Bootstrap para un diseño estilizado [25].

2.2.3.5. Python

Python es un lenguaje de programación muy popular esto se debe a que es un lenguaje de alto nivel con una sintaxis muy limpia y sencilla [26] [27], además de poseer herramientas para la manipulación y análisis de datos [28].

2.2.3.6. GNS3

La herramienta GNS3 se emplea para la virtualización de laboratorios de redes reales, permite la emulación de redes de computadores por lo que es considerado un simulador gráfico de red [29] [30], también permite ejecutar múltiples sistemas operativos al mismo tiempo en diferentes máquinas virtuales.

2.2.3.7. API REST

Una API de REST se basa en el protocolo Http, el cual consiste realizar una petición para enviar datos o recursos y recibir una respuesta. Los métodos Http más comunes son GET, POST, PUT, DELETE, estos métodos se implementan para realizar acciones como leer, modificar y eliminar respectivamente [31] [32]. Un API REST se adapta a cualquier lenguaje de programación, normalmente usa el formato JSON para responder las solicitudes [33].

2.3. OBJETIVOS DEL PROTOTIPO

2.3.1. Objetivo General

Desarrollar un sitio web mediante el uso de API REST en lenguaje Python para la gestión de asignación de IP.

2.3.2. Objetivos Específicos

- Implementar una aplicación web para la conexión con dispositivos de red.
- Utilizar un API REST en lenguaje Python para la ejecución de las funciones del sitio web.
- Virtualizar una topología de red utilizando el software GNS3 para verificar la eficacia del sistema.

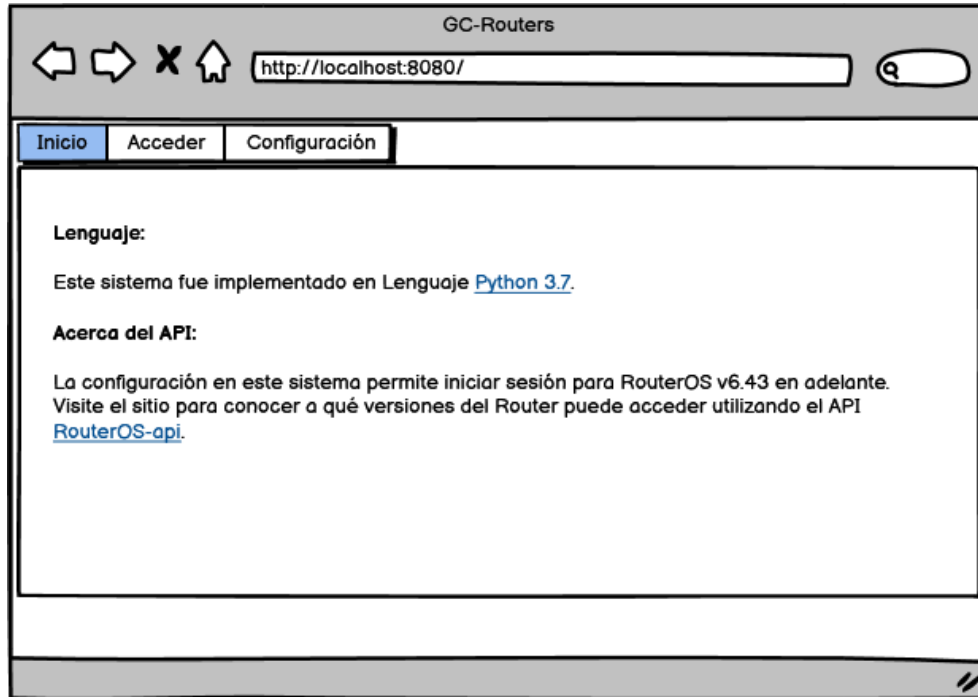
2.4. DISEÑO DEL PROTOTIPO

Para la creación de la presente propuesta tecnológica se ha optado por implementar la metodología SCRUM, ya que es una metodología ágil para trabajo en equipo gracias a sus iteraciones o Sprints, las cuales ayudan a un mejor control y planificación del proyecto, cada Sprint posee eventos o etapas y entre las más comunes tenemos:

- **Reunión de planificación de Sprint:** Estas reuniones ayudan a que todo el equipo se ponga de acuerdo con el tiempo que se tomará para realizar el Sprint y lo que se entregará al terminarlo.
- **El Scrum Diario:** Consiste en reuniones cortas diarias que ayudan a mantener sincronizadas las actividades a desarrollar.
- **Revisión del Sprint:** Se llevan a cabo reuniones con el dueño del proyecto para estudiar el Product Backlog, considerando así las próximas tareas a realizar en el siguiente Sprint.
- **Retrospectiva del Sprint:** El equipo de trabajo crea posibles soluciones o mejoras para ejecutarlas en el próximo Sprint.

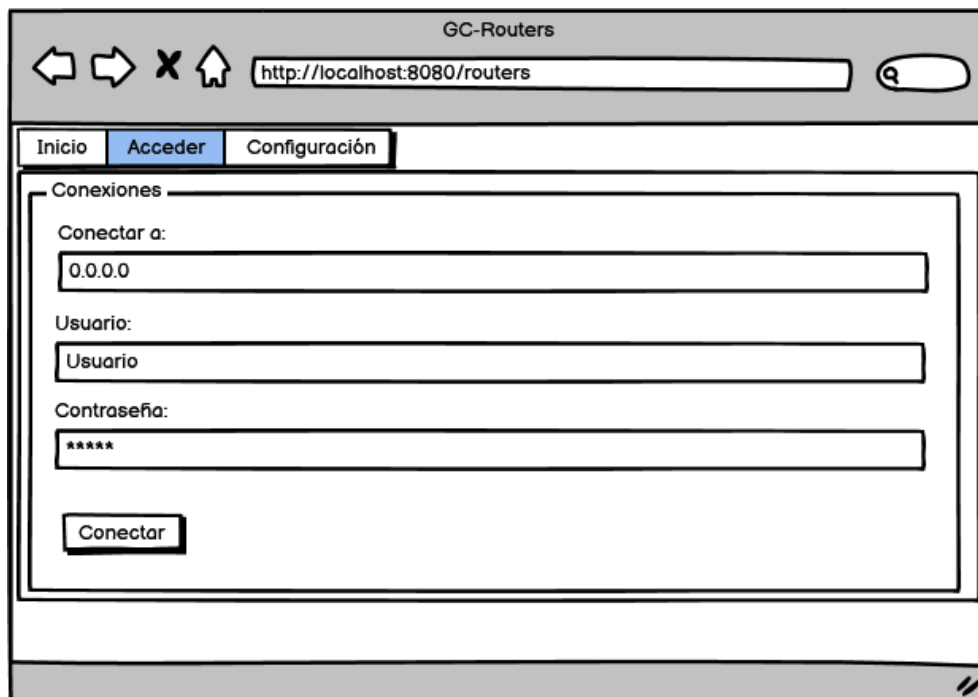
2.4.1. Diseño de interfaces

Figura 4: Diseño de interfaces – Pantalla principal



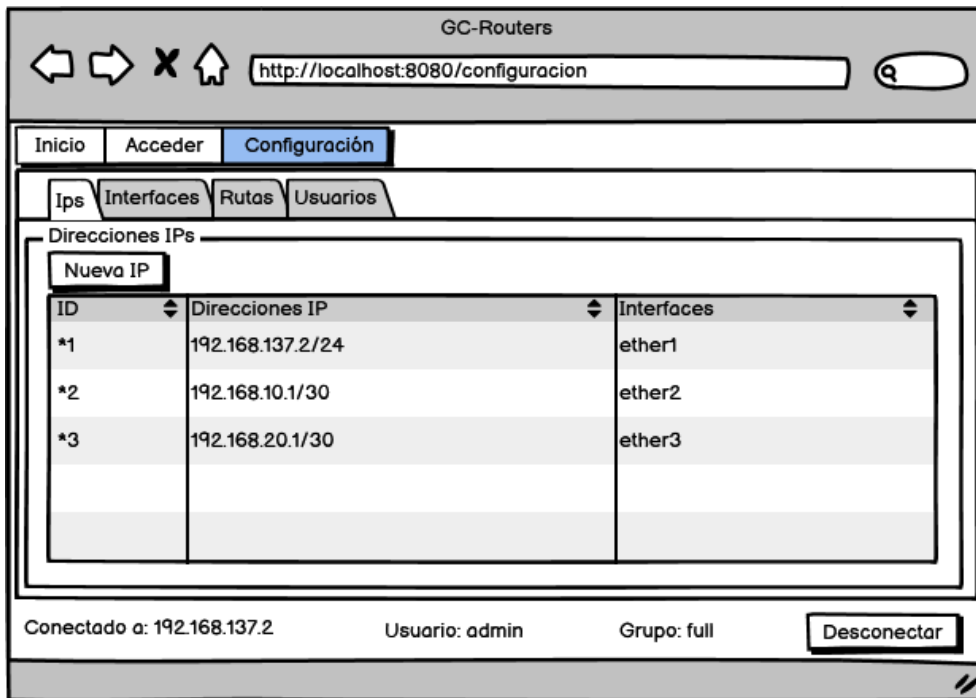
Fuente: Elaboración propia

Figura 5: Diseño de interfaces - Formulario de inicio de sesión en el router



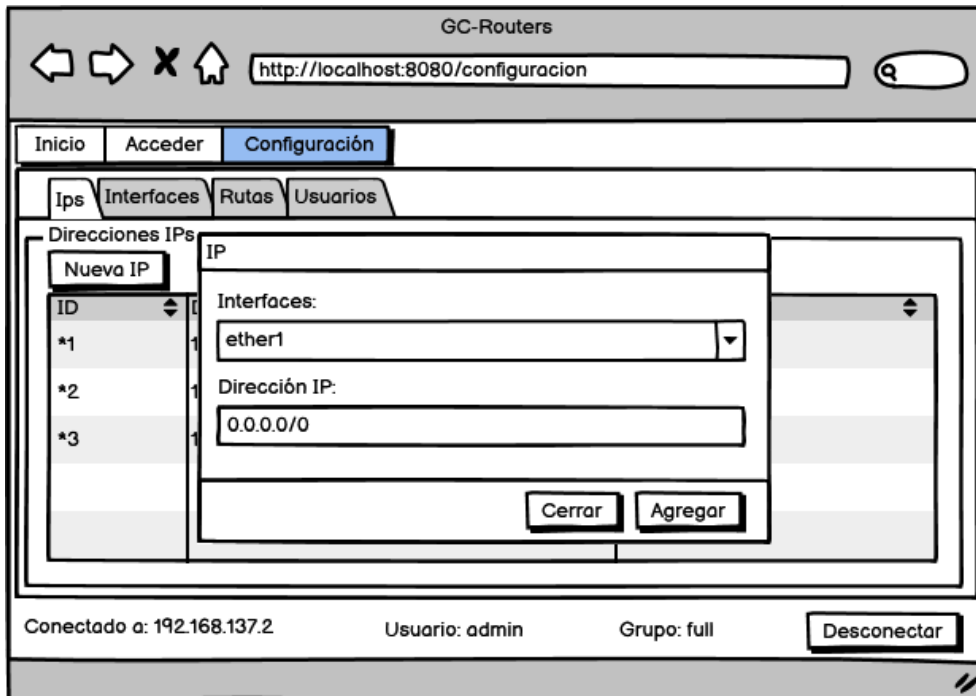
Fuente: Elaboración propia

Figura 6: Diseño de interfaces - Vista de direcciones IP del router



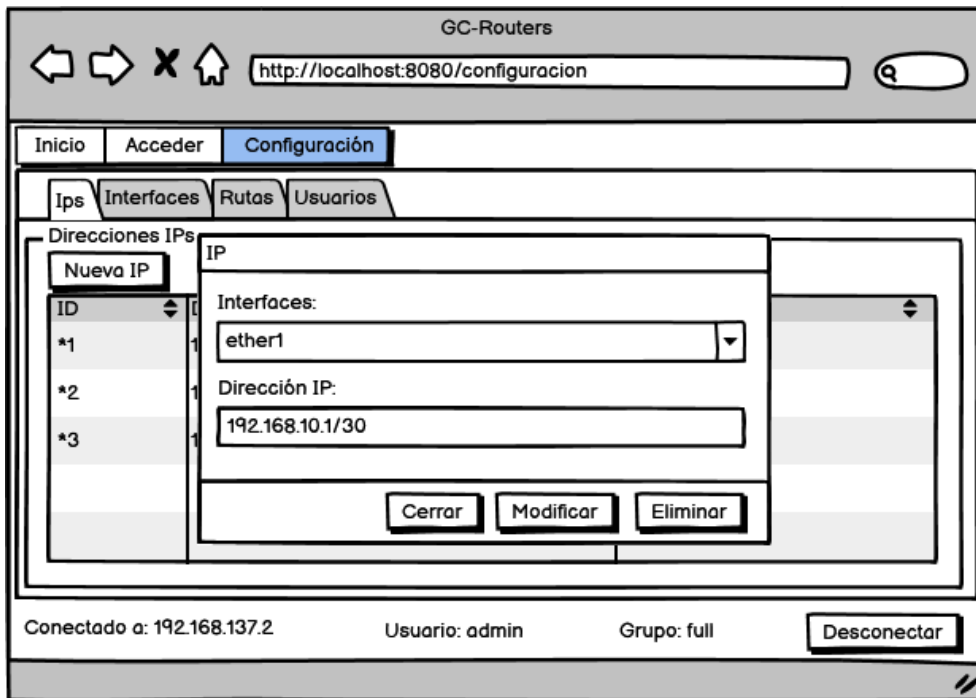
Fuente: Elaboración propia

Figura 7: Diseño de interfaces - Ingreso de dirección IP al router



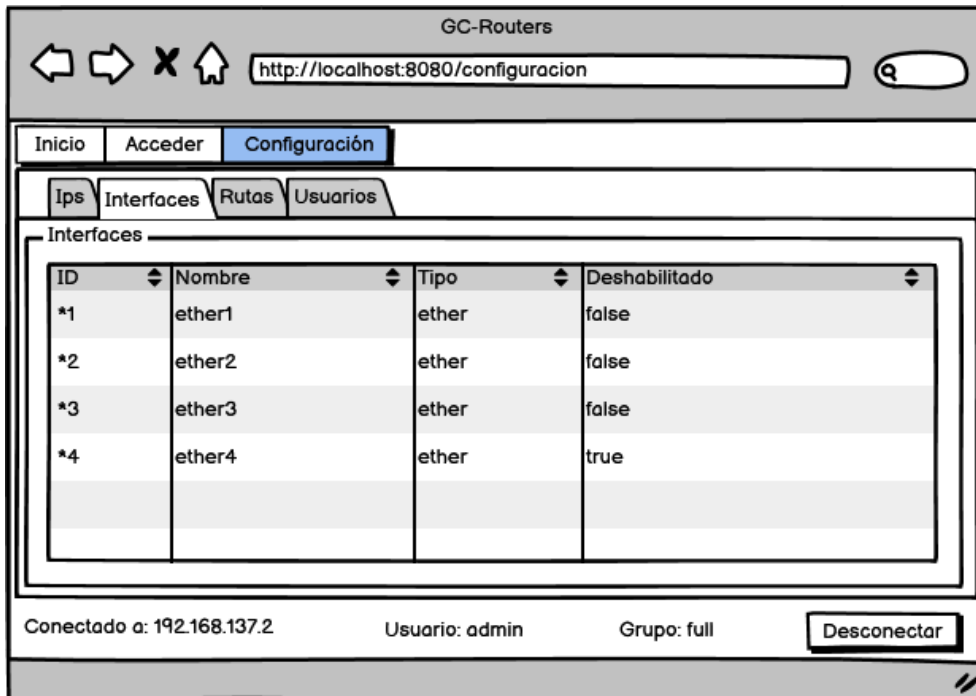
Fuente: Elaboración propia

Figura 8: Diseño de interfaces - Modificación o eliminación de dirección IP del router



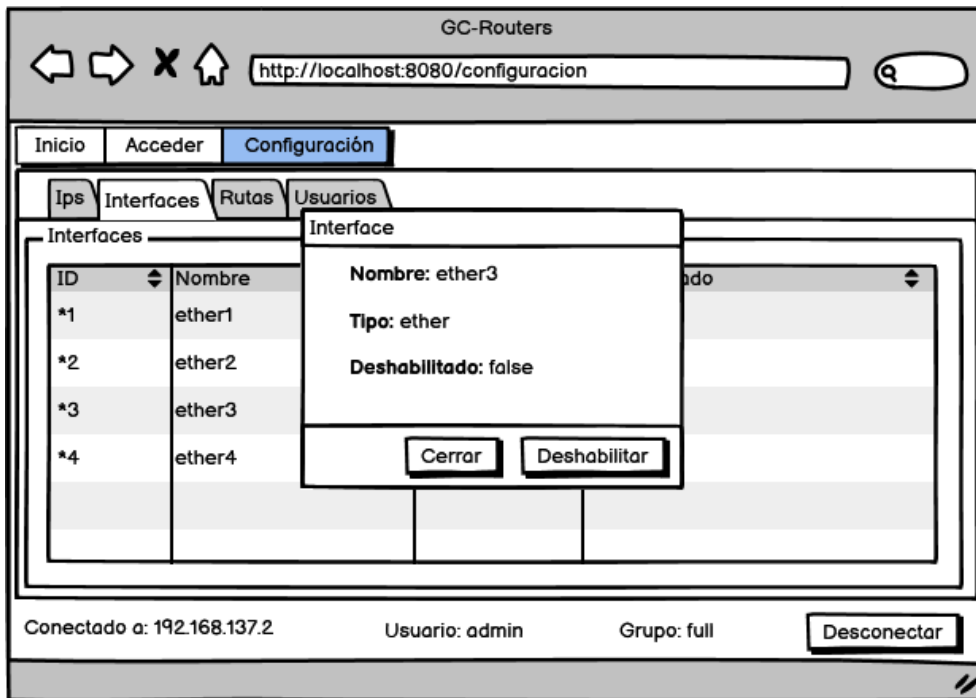
Fuente: Elaboración propia

Figura 9: Diseño de interfaces - Vista de interfaces del router



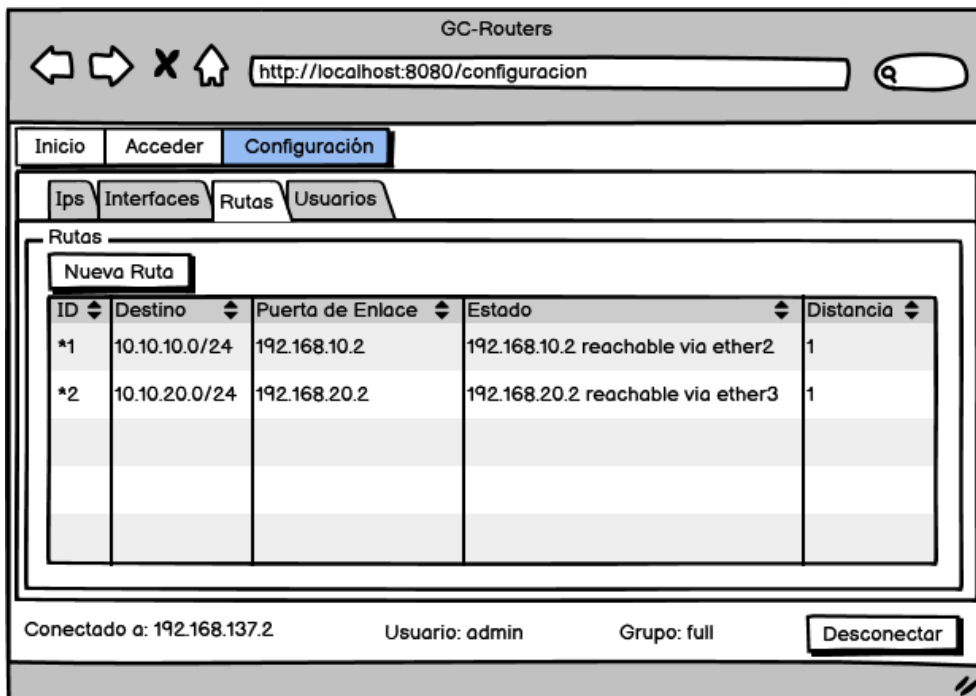
Fuente: Elaboración propia

Figura 10: Diseño de interfaces - Habilitar o deshabilitar interfaces del router



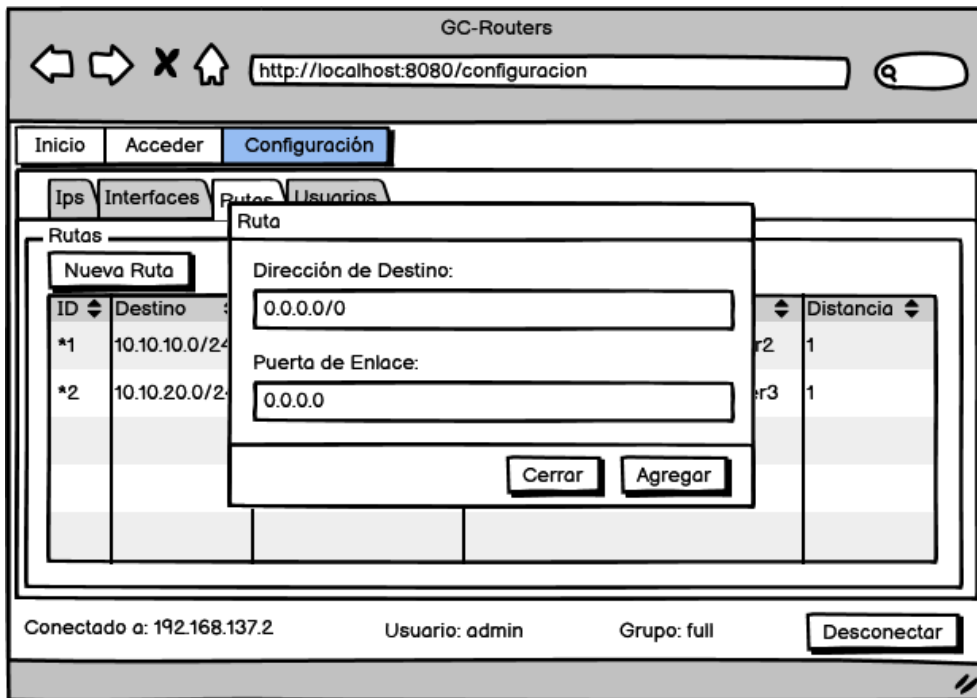
Fuente: Elaboración propia

Figura 11: Diseño de interfaces - Vista de rutas del router



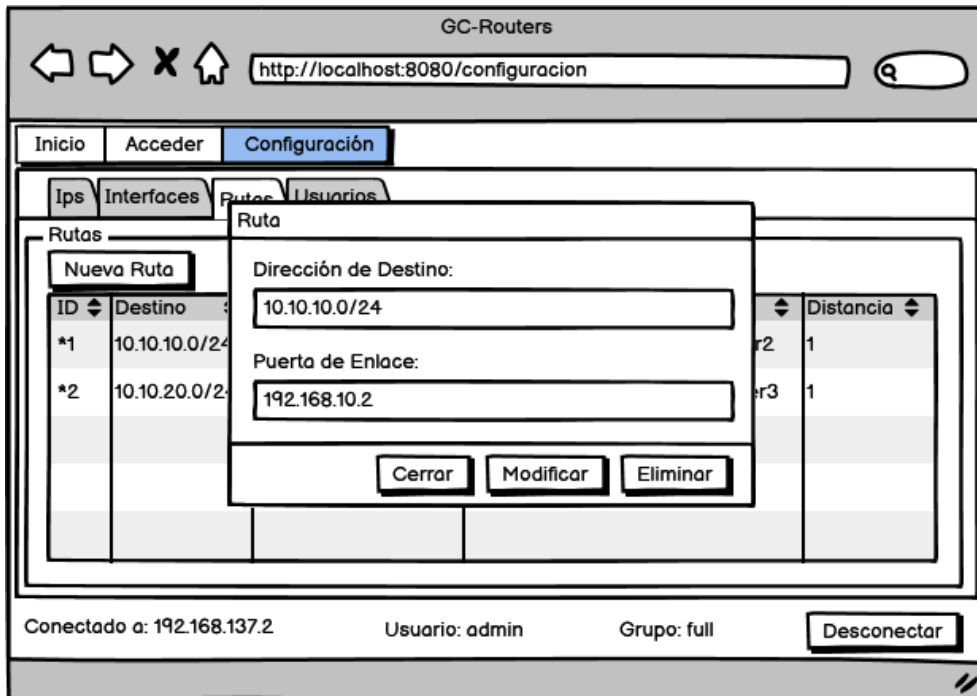
Fuente: Elaboración propia

Figura 12: Diseño de interfaces - Ingreso de ruta al router



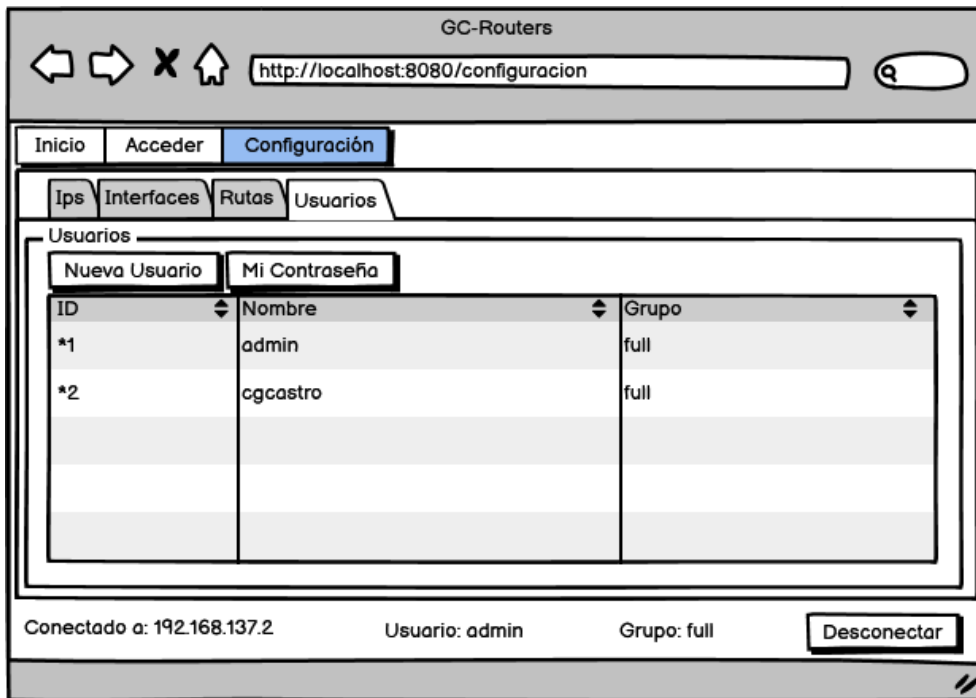
Fuente: Elaboración propia

Figura 13: Diseño de interfaces - Modificación o eliminación de ruta del router



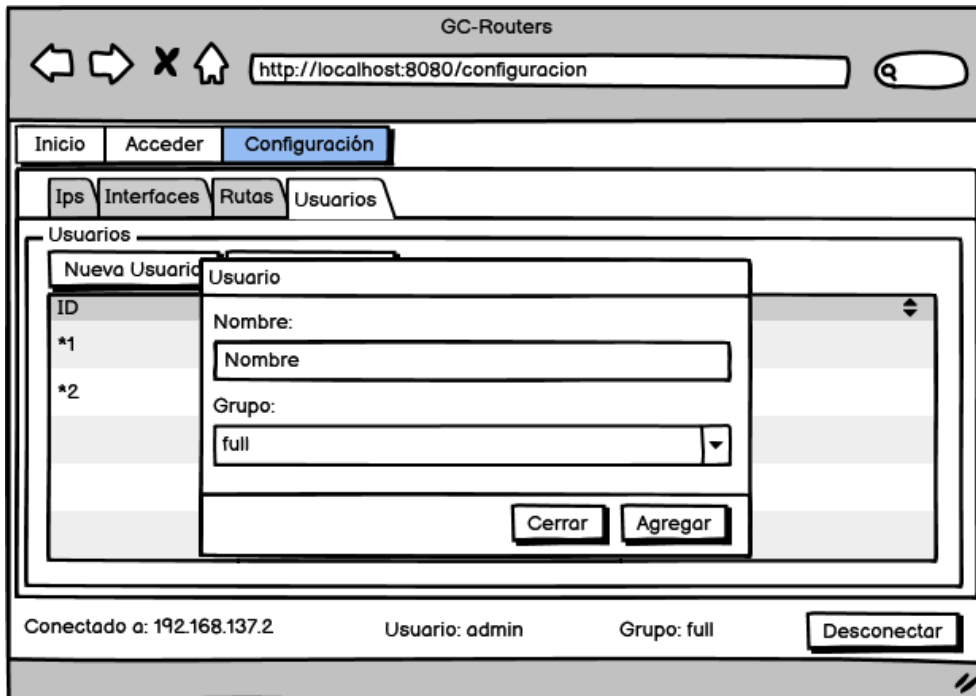
Fuente: Elaboración propia

Figura 14: Diseño de interfaces - Vista de usuarios del router



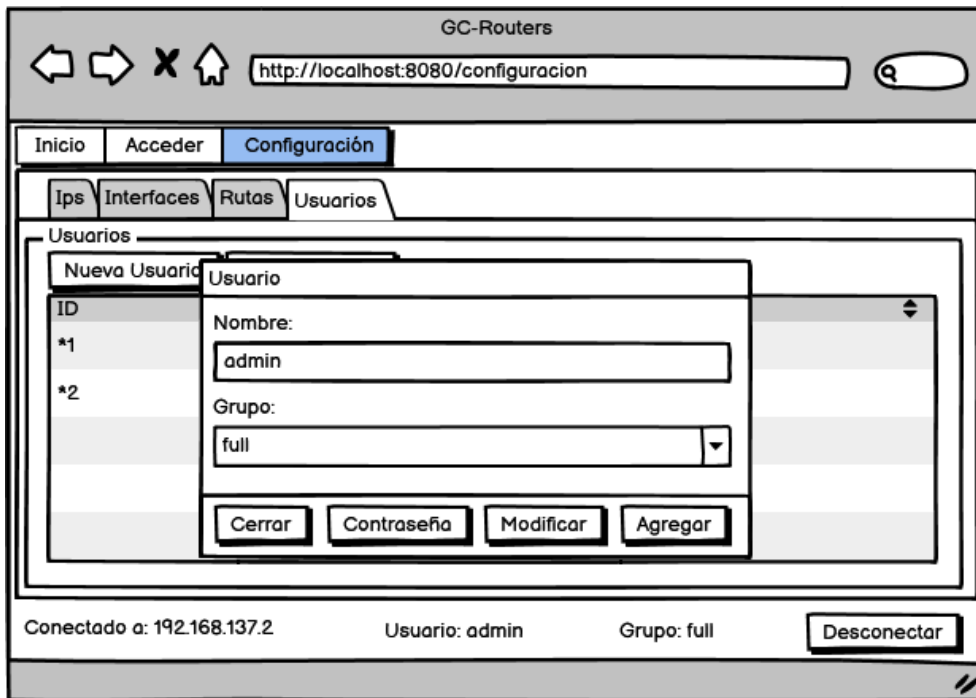
Fuente: Elaboración propia

Figura 15: Diseño de interfaces - Ingreso de usuario al router



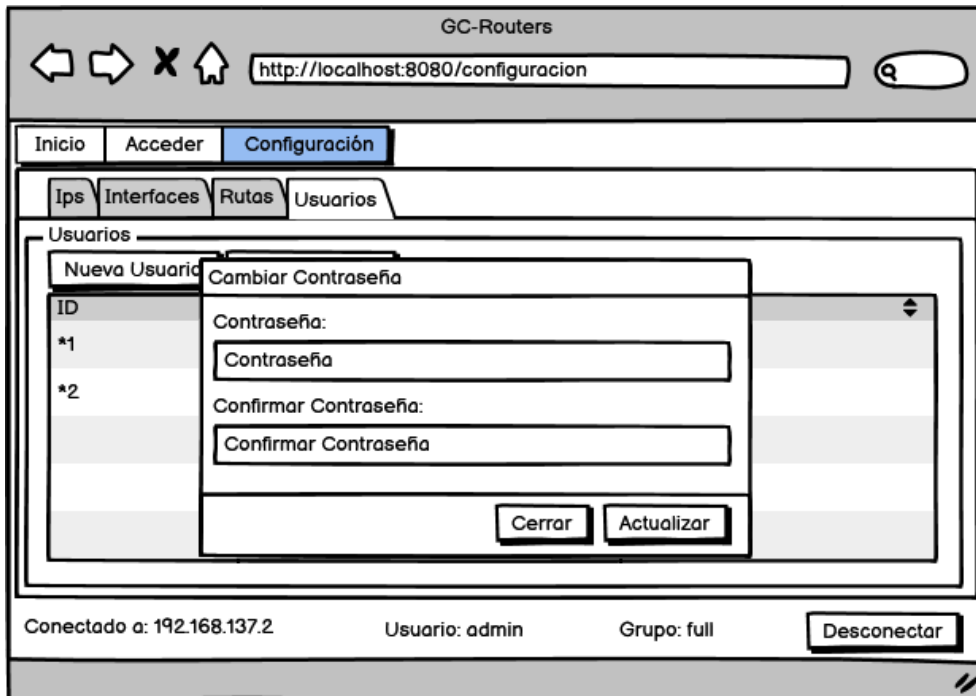
Fuente: Elaboración propia

Figura 16: Diseño de interfaces – Modificación o eliminación de usuario del router



Fuente: Elaboración propia

Figura 17: Diseño de interfaces – Cambio de contraseña de usuario del router



Fuente: Elaboración propia

2.5. EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO

En esta sección se describe el funcionamiento del sistema para la asignación de IP a un dispositivo de red, además de la configuración del enrutamiento estático para la conexión entre dispositivos.

2.5.1. Pantalla principal

En la pantalla principal se puede observar información acerca del lenguaje con el que fue creado el sistema, además del API utilizado para la creación de funciones que interactúan con el router.

Figura 18: Ejecución del prototipo - Pantalla principal

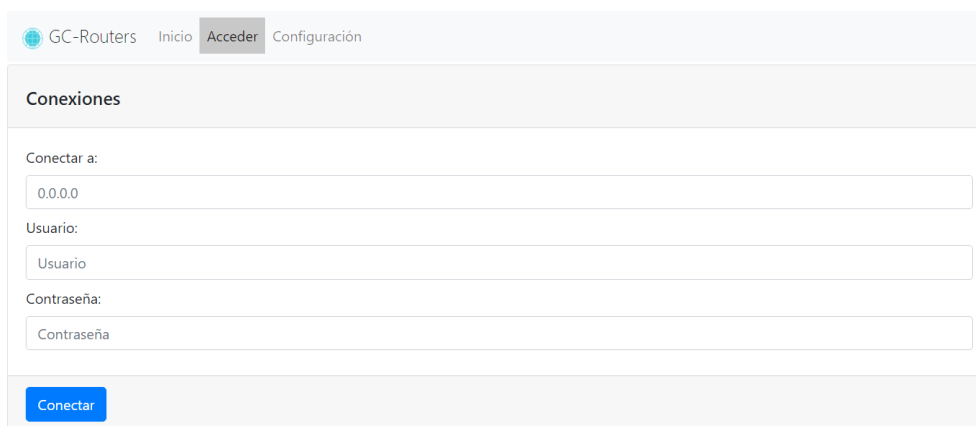


Fuente: Elaboración propia

2.5.2. Inicio de sesión del router

Para iniciar sesión en el router, se debe ir al apartado Acceder, el cual pide ingresar la dirección IP de acceso al router, usuario y contraseña.

Figura 19: Ejecución del prototipo - Inicio de sesión del router



Fuente: Elaboración propia

2.5.3. Vista de direcciones IP del router

Una vez iniciada sesión en el router, se pueden observar las direcciones IP y sus interfaces. La información del usuario se encuentra en la parte inferior, además de un botón para desconectarse del router y regresar al apartado Acceder.

Figura 20: Ejecución del prototipo - Vista de direcciones IP del router

The screenshot shows the 'Configuración' (Configuration) page of the GC-Routers application. The 'IPs' tab is selected, displaying a table of IP addresses and their associated interfaces. The table has three columns: 'ID', 'Direcciones IP', and 'Interfaces'. There are three entries in the table. Below the table, it indicates 'Mostrando del (1 al 3) de 3 registros' (Showing 1 to 3 of 3 records) and navigation buttons for 'Anterior' (Previous), '1', and 'Siguiente' (Next). At the bottom of the page, there is a status bar showing 'Conectado a 192.168.137.2', 'Usuario: admin', 'Grupo: full', and a 'Desconectar' (Disconnect) button.

ID	Direcciones IP	Interfaces
*1	192.168.137.2/24	ether1
*2	192.168.10.1/30	ether2
*4	192.168.20.1/30	ether3

Fuente: Elaboración propia

2.5.4. Ingreso de dirección IP del router

Al dar clic en el botón Nueva IP se abrirá un modal para ingresar una dirección IP y seleccionar la interfaz a la que será agregada. El sistema valida si la IP ingresada es correcta o no.

Figura 21: Ejecución del prototipo - Ingreso de dirección IP al router

The screenshot shows a modal window titled 'IP' for adding a new IP address. It contains two input fields: 'Interfaces:' with a dropdown menu showing 'ether1', and 'Dirección IP:' with a text input field containing '0.0.0.0/0'. At the bottom right, there are two buttons: 'Cerrar' (Close) and 'Agregar' (Add).

Fuente: Elaboración propia

2.5.5. Modificación o eliminación de dirección IP del router

Al dar clic en las filas de la tabla se abre un modal donde se puede modificar o eliminar la IP. Estas funciones alteran la información a través de un ID por lo que se puede cambiar en su totalidad la dirección IP o seleccionar cualquiera de las interfaces.

Figura 22: Ejecución del prototipo - Modificación o eliminación de dirección IP del router



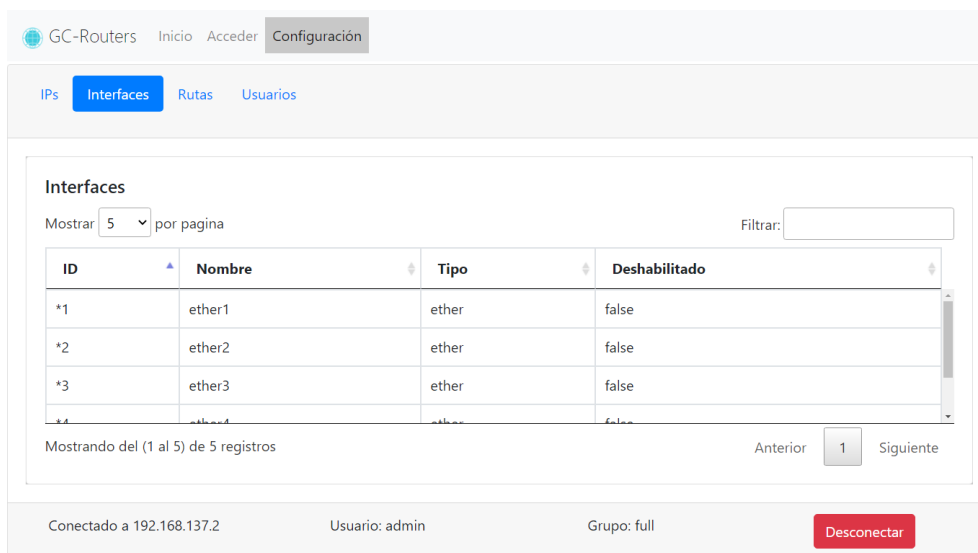
The modal window is titled "IP". It contains a dropdown menu labeled "Interfaces:" with "ether1" selected. Below it is a text input field labeled "Dirección IP:" containing "192.168.137.2/24". At the bottom right, there are three buttons: "Cerrar" (grey), "Modificar" (yellow), and "Eliminar" (red).

Fuente: Elaboración propia

2.5.6. Vista de interfaces del router

En el apartado de Interfaces se puede observar las interfaces disponibles en el router, donde se distinguen sus tipos y se verifica si están deshabilitadas o no.

Figura 23: Ejecución del prototipo - Vista de interfaces del router



The screenshot shows the "Configuración" section of the GC-Routers interface. The "Interfaces" tab is active. A table lists the available interfaces:

ID	Nombre	Tipo	Deshabilitado
*1	ether1	ether	false
*2	ether2	ether	false
*3	ether3	ether	false
*4	ether4	ether	false

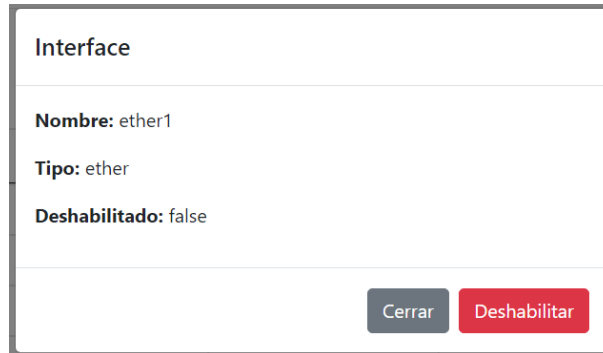
Below the table, it says "Mostrando del (1 al 5) de 5 registros". Navigation buttons "Anterior" and "Siguiete" are visible, with "1" highlighted. At the bottom, the user is logged in as "admin" with the group "full". A "Desconectar" button is present.

Fuente: Elaboración propia

2.5.7. Habilitar o deshabilitar interfaces del router

Al dar clic en alguna fila de la tabla de interfaces, se puede observar información de la interfaz seleccionada y habilitar o deshabilitar dependiendo de su estado.

Figura 24: Ejecución del prototipo - Habilitar o deshabilitar interfaces del router

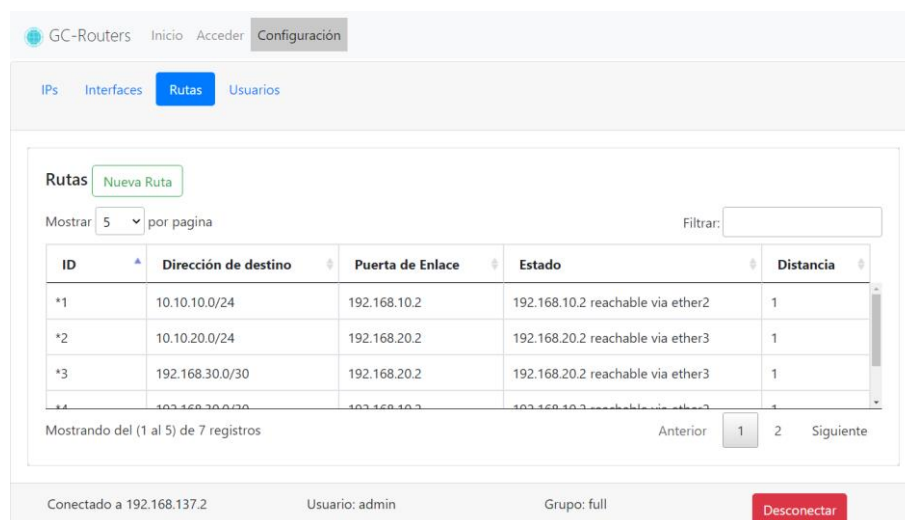


Fuente: Elaboración propia

2.5.8. Vista de rutas del router

Al ingresar al apartado de rutas se pueden observar las direcciones de destino, puertas de enlace, estado y distancia de las rutas existentes en el router. Algunas interfaces son creadas por defecto al ingresar una nueva dirección IP al router.

Figura 25: Ejecución del prototipo - Vista de rutas del router

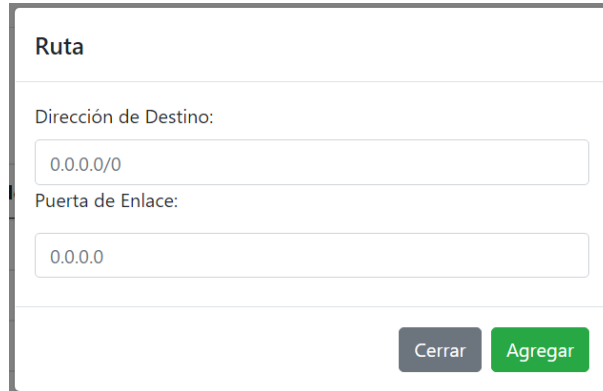


Fuente: Elaboración propia

2.5.9. Ingreso de ruta al router

Al dar clic al botón Nueva Ruta se abrirá un modal donde el usuario deberá ingresar la dirección IP de destino y la puerta de enlace a la que se quiera acceder.

Figura 26: Ejecución del prototipo - Ingreso de ruta al router



Ruta

Dirección de Destino:
0.0.0.0/0

Puerta de Enlace:
0.0.0.0

Cerrar Agregar

Fuente: Elaboración propia

2.5.10. Modificación o eliminación de ruta del router

Al seleccionar una de las filas de la tabla de rutas se abrirá un modal, el cual permitirá modificar la ruta registrada en el router o eliminarla.

Figura 27: Ejecución del prototipo - Modificación o eliminación de ruta del router



Ruta

Dirección de Destino:
10.10.10.0/24

Puerta de Enlace:
192.168.10.2

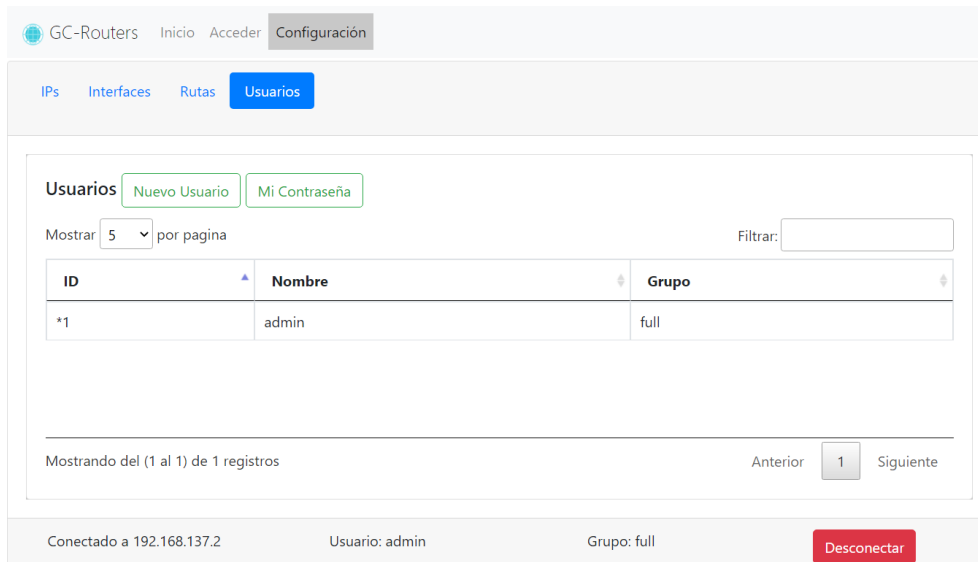
Cerrar Modificar Eliminar

Fuente: Elaboración propia

2.5.11. Vista de usuarios del router

Dentro de la sección de Usuarios podemos observar los usuarios del router y los grupos a los que pertenecen, el usuario admin dentro del grupo full ya viene creado por defecto en el router.

Figura 28: Ejecución del prototipo - Vista de usuarios del router

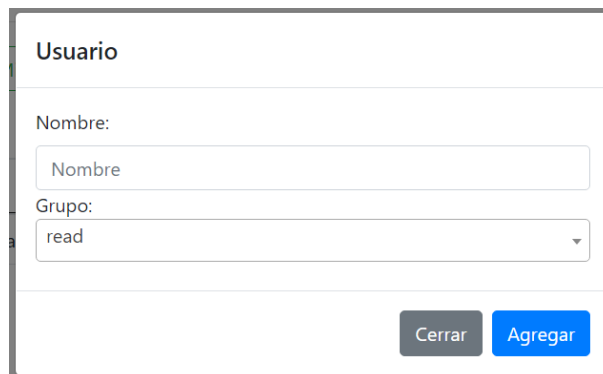


Fuente: Elaboración propia

2.5.12. Ingreso de usuario al router

Al dar clic en Nuevo Usuario se abrirá un modal donde se pueden registrar el nombre de usuario y el grupo al que pertenecerá, los grupos son los ya definidos dentro del router y cada uno otorgará sus respectivos permisos al usuario al momento de ser agregado.

Figura 29: Ejecución del prototipo - Ingreso de usuario al router



Fuente: Elaboración propia

2.5.13. Modificación o eliminación de usuario del router

Al seleccionar una fila de la tabla de usuarios se abrirá un modal donde se puede modificar el nombre de usuario y el grupo al que pertenece, también hay un botón que permitirá cambiar la contraseña de ese usuario sólo si el usuario con el que se inició la sesión en el router se encuentra dentro del grupo full.

Figura 30: Ejecución del prototipo - Modificación o eliminación de usuario del router



Usuario

Nombre:
admin

Grupo:
full

Cerrar Contraseña Modificar Eliminar

Fuente: Elaboración propia

2.5.14. Cambio de contraseña de usuario del router

El modal para el cambio de contraseña se abrirá si se invoca desde el botón dentro del modal de información del usuario o desde el botón llamado Mi Contraseña. Este modal actualizará la contraseña del usuario seleccionado o en su segundo caso de invocación, actualizará la contraseña del usuario con el que se inició sesión en el router.

Figura 31: Ejecución del prototipo - Cambio de contraseña de usuario del router



Cambiar Contraseña

Contraseña:
Contraseña

Confirmar Contraseña:
Confirmar Contraseña

Cerrar Actualizar

Fuente: Elaboración propia

3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

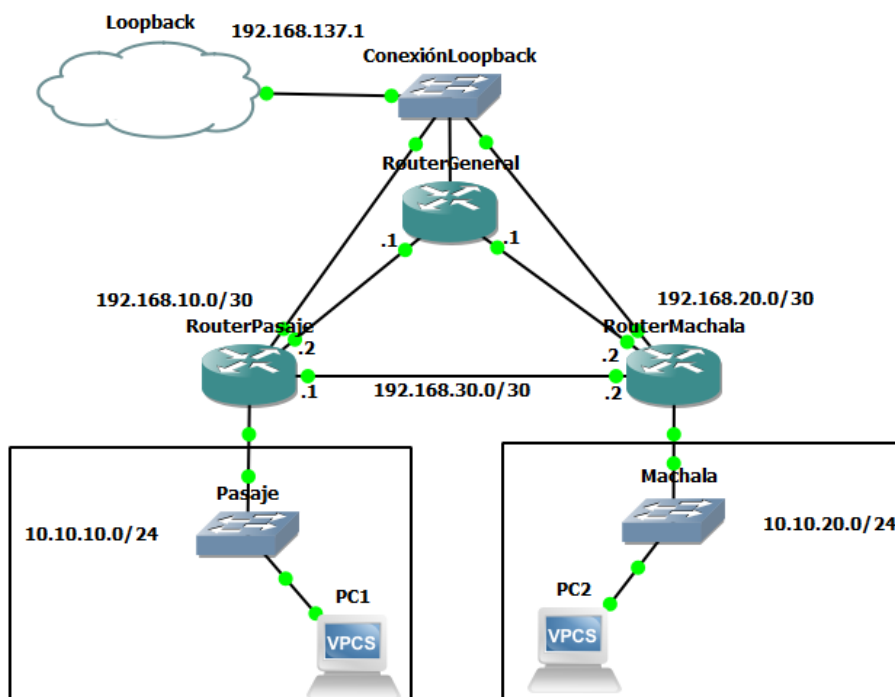
3.1. PLAN DE EVALUACIÓN

Para evaluar la funcionalidad del sitio web para gestión de asignación de IP se hizo una comparación entre el prototipo y Winbox, la cual es una aplicación Windows que permite administrar routers Mikrotik. Además del uso de la herramienta GNS3 para la virtualización de los routers. Los parámetros a evaluar fueron el inicio de sesión al router, el direccionamiento de la red, la creación de rutas y la creación de usuarios con sus respectivos permisos.

3.1.1. Preparación de herramientas

El software GNS3 permite la elaboración de un laboratorio virtual gracias a que puede emular dispositivos virtuales y reales. Una topología fue creada específicamente para probar la funcionalidad del prototipo tecnológico. Las imágenes de los routers usados están en su versión chr-6.47.6. Estas imágenes se encuentran disponibles en la página de descarga de Mikrotik.

Figura 32: Topología virtualizada para las pruebas del prototipo tecnológico



Fuente: Elaboración propia

3.1.2. Inicio de sesión

Para lograr tener acceso al router Mikrotik se necesitan ciertos parámetros específicos, los cuales constan de la dirección IP o MAC del router, el nombre de usuario que se va a conectar, el cual por defecto es admin al ser nuevo el dispositivo, y la contraseña en caso de tener una ya que al tratarse de un dispositivo nuevo no posee contraseña alguna.

3.1.3. Direccionamiento IP

El direccionamiento IP se logra introduciendo la dirección IP y la máscara de subred, también es necesario seleccionar la interfaz a la que serán agregados los campos ya especificados.

3.1.4. Administración de interfaces

Es importante tener el control de las interfaces del router para poder deshabilitarlas o habilitarlas en caso de necesario. Esta opción también está disponible en el prototipo.

3.1.5. Creación de rutas

La creación de rutas nos ayuda a conectarnos con otros routers y así poder compartir el acceso a direccionamientos externos a nuestros routers. Para ello es necesario especificar una dirección de destino y una puerta de enlace.

3.1.6. Administración de usuarios

En los routers la creación de usuarios nos permite administrar las tareas que se pueden o no realizar al momento de iniciar sesión, estos permisos incluyen la lectura, escritura o el control total sobre el dispositivo de red.

3.2. RESULTADOS DE LA EVALUACIÓN

3.2.1. Resultado de Inicio de sesión

Tabla 1: Resultado de inicio de sesión

Parámetros	Winbox	Prototipo tecnológico (API REST)
Dirección IP	X	X
MAC	X	
Usuario	X	X
Contraseña	X	X

Fuente: Elaboración propia

En la prueba de inicio de sesión, el API REST utilizado no podía acceder al router utilizando la dirección MAC, pero con la dirección IP no presentaba ni un tipo de problema. A diferencia de la herramienta Winbox que puede iniciar sesión con cualquiera de las direcciones ya mencionadas.

3.2.2. Resultado de Direccionamiento IP

Tabla 2: Resultado de Direccionamiento IP

Parámetros	Winbox	Prototipo tecnológico (API REST)
Dirección IP	X	X
Network	X	
Interface	X	X

Fuente: Elaboración propia

Para el direccionamiento IP, la herramienta Winbox requiere del ingreso de la dirección IP y presionando en el botón Apply se ingresa automáticamente el Network, el API REST también calcula el Network internamente y no requiere el ingreso directo del mismo, con la dirección IP e interface es suficiente.

3.2.3. Resultado de Administración de interfaces

Tabla 3: Resultado de Administración de interfaces

Parámetros	Winbox	Prototipo tecnológico (API REST)
Habilitar	X	X
Deshabilitar	X	X

Fuente: Elaboración propia

Las opciones de habilitar y deshabilitar las interfaces se encontraron tanto en la herramienta Winbox como en el API REST al seleccionar la interfaz a la que se realizará el cambio de estado.

3.2.4. Resultado de Creación de rutas

Tabla 4: Resultado de Creación de rutas

Parámetros	Winbox	Prototipo tecnológico (API REST)
Dirección de destino	X	X
Puerta de enlace (Gateway)	X	X

Fuente: Elaboración propia

La creación de ruta fue efectiva tanto para la herramienta Winbox como para el API REST. Se utiliza la dirección IP de destino y una puerta de enlace para crearla.

3.2.5. Resultado de Administración de usuarios

Tabla 5: Resultado de Administración de usuarios

Parámetros	Winbox	Prototipo tecnológico (API REST)
Nombre	X	X
Grupo	X	X
Cambio de Contraseña	X	X

Fuente: Elaboración propia

La administración de usuarios ayuda en la configuración de las opciones que posee el usuario que iniciará sesión en el router. Tanto la herramienta Winbox como el API REST utilizado permiten la creación de usuarios con permisos. Dichos usuarios pueden iniciar sesión con normalidad en el router una vez que son creados.

3.3. CONCLUSIONES

- El uso de API REST en la elaboración de aplicaciones web ayuda a reducir los procesos creados para las interacciones entre el cliente y servidor, ya que éstas contienen funciones que reciben información del cliente para ejecutar una acción y devolver un resultado.
- Mediante el uso del prototipo creado en esta propuesta tecnológica se ha realizado el direccionamiento IP, la creación de rutas y la creación de usuarios en un router, de esta manera se logró la conexión entre varios dispositivos pertenecientes a una topología virtualizada en el software GNS3.
- El uso de herramientas que simulan los dispositivos de red como los router, permiten a los desarrolladores poner a prueba el aplicativo que están desarrollando mediante el uso de API REST para asegurar su correcto funcionamiento.

3.4. RECOMENDACIONES

- Es importante definir el alcance de una aplicación web antes de elegir el API REST que se va a utilizar, ya que no todas poseen la misma capacidad. Además, dependiendo del lenguaje de programación que se utilizará, es ventajoso o no el uso de la misma.
- Tener en claro las funciones que el aplicativo web utilizará, ahorra tiempo a los desarrolladores y logra hacer que se destaque del resto de aplicaciones similares.
- Elegir las herramientas adecuadas para probar el funcionamiento de una aplicación que utiliza un API REST, para no tener inconvenientes al aplicarlo en el ámbito laboral profesional.

Bibliografía

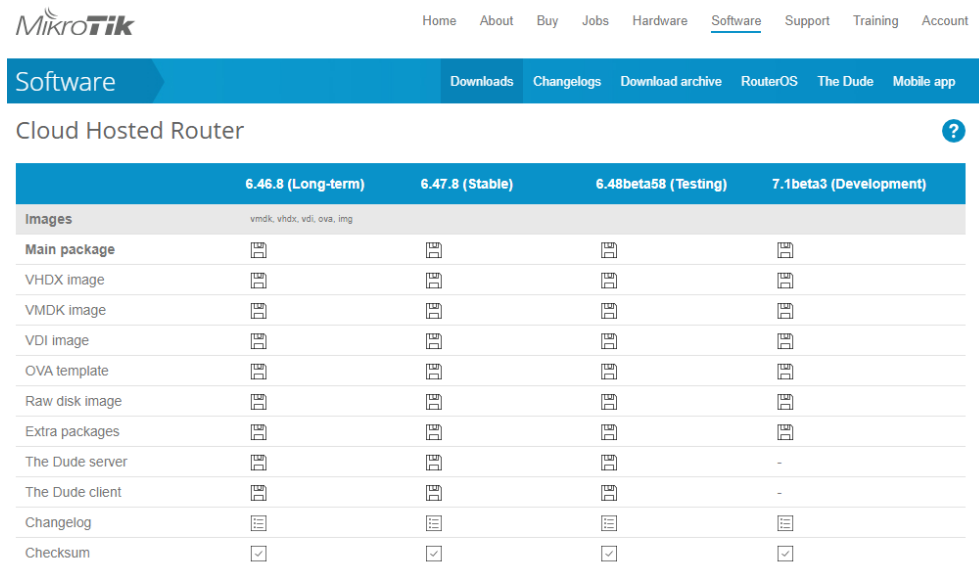
- [1] A. Arsaute, F. Zorzán, M. Daniele, A. González y M. Frutos, «Generación automática de API REST a partir de API Java, basada en transformación de Modelos (MDD),» 26 Abril 2018. [En línea]. Available: <http://sedici.unlp.edu.ar/handle/10915/67777>. [Último acceso: 4 Diciembre 2020].
- [2] G. Mondragón Toledo, «La Importancia de Internet como Fuente de Poder de la Sociedad,» *Pensamiento al margen*, vol. I, nº 3, p. 2, 2015.
- [3] J. Aguilar-Alvarado, R. Quezada-Sarmiento y K. García-Galarza, «Aplicación Java para el control de RB Mikrotik en empresas proveedoras de servicio de Internet,» *Ciencia UNEMI*, vol. 11, nº 26, pp. 161 - 169, 2018.
- [4] A. S. T. A. T. B. S. B. S. D. J. G. O. G. M. G. A. H. U. H. H. L. M. L. H. R. J. S. A. v. S. J. S. A. W. M. Grahl, «W7-X logbook REST API for processing experimental metadata and data enrichment at the Wendelstein 7-X stellarator,» *Fusion Engineering and Design*, vol. 160, 2020.
- [5] Y. Oktian, S.-G. Lee y J. Lam, «OAuthkeeper: An Authorization Framework for Software Defined Network,» *Journal of Network and Systems Management*, vol. 26, pp. 147-168, 2018.
- [6] A. Shirmaez y A. Ghaffari, «Performance issues and solutions in SDN-based data center: a survey,» *The Journal of Supercomputing*, vol. 76, pp. 7545-7593, 2020.
- [7] S. Ahmad y M. A. Hussain, «Scalability, Consistency, Reliability and Security in SDN Controllers: A Survey of Diverse SDN Controllers,» *Journal of Network and Systems Management*, vol. 29, nº 9, 2020.
- [8] A. Delgado, E. Chapman, O. Ivañez, J. Molina, R. Tasca y J. Caballero, «Servidor de Terminología Médica para el Hospital de Clínicas de Paraguay utilizando Apache Lucene,» *Revista Cubana de Informática Médica*, vol. 10, nº 2, 2018.
- [9] J. Honores y R. Dickson, «Implementación de un servicio ecommerce utilizando la metodología xp enfocado a la gastronomía de Machala,» Machala: Universidad Técnica de Machala, Machala, 2019.
- [10] T. Turc, «AJAX Technology for Internet of Things,» *Procedia Manufacturing*, vol. 32, pp. 613-618, 2019.
- [11] B. Oviedo, L. Suarez, Z. Emilio, A. Puris y R. Hernández, «Visualizador de tráfico de red de comunicación basadas en la Arquitectura TCP/IP,» *Revista Universidad y Sociedad*, vol. 11, nº 2, pp. 192-202, 2019.
- [12] F. E. Ríos Perez, F. Polcanco Carrillo y V. Moreno Vega, «Servidor web empotrado en un FPGA para configurar un Controlador Maestro del Sistema Inteligente de Tráfico Cubano FPGA,» *cuba cienc informat*, vol. 11, nº 2, pp. 16-28, 2017.

- [13] D. Vera, L. Córdova, R. López y S. Pacheco, «Análisis de la metodología RUP en el desarrollo de software académico mediante la herramienta DJANGO,» *RECIMUNDO: Revista Científica de la Investigación y el Conocimiento*, vol. 3, nº 2, pp. 964-979, 2019.
- [14] J. R. Molina Ríos, M. P. Zea Ordóñez, M. J. Contenido Segarra y F. G. García Zeda, «Estado Del Arte: Metodologías De Desarrollo En Aplicaciones Web,» *3c Tecnol. glosas innovación Apl. a la pyme*, vol. 6, nº 3, pp. 305-307, 2017.
- [15] S. P. Castillo Landínez, P. E. Caicedo Rodríguez, D. F. Sánchez Gómez y C. Carvalho, «Diseño e implementación de un software para la trazabilidad del proceso de beneficio del café,» *Ciencia y Tecnología Agropecuaria*, vol. 20, nº 3, pp. 523-536, 2019.
- [16] M. R. Valarezo Pardo y D. X. Sarango Yunga, «Desarrollo de plataforma web para la evaluación de software basado en la metodología SCRUM,» Machala: Universidad Técnica de Machala, Machala, 2020.
- [17] P. E. De la Cruz, M. Espinoza y O. Cuba, «Propuesta de arquitectura de microservicios, metodología Scrum para una aplicación móvil de control académico: Caso escuela profesional de obstetricia de la universidad Nacional Mayor de San Marcos,» *Hamut'ay*, vol. 6, nº 2, pp. 141-158, 2019.
- [18] F. Sáenz Blanco, F. Gutierrez Sierra y J. C. Ramos Rivera, «CONFORMACIÓN DE EQUIPOS ÁGILES PARA EL DESARROLLO DE SOFTWARE: REVISIÓN DE LITERATURA,» *Dimension Empresarial*, vol. 16, nº 2, pp. 39-54, 2018.
- [19] B. Kumar, K. Abhishek, A. Deepak y M. Singh, «Implementation of Interactive Real Time Online Co-shopping Using Push AJAX,» *Procedia Computer Science*, vol. 89, pp. 473-482, 2016.
- [20] M. Morales, C. Huanca y R. Casilla, «Aplicación web: sistema de información para la gestión académica (SIGA),» *Revista Investigación y Tecnología*, vol. 3, pp. 21-24, 2015.
- [21] H. A. Quintana-Cruz, «Implementación de aplicaciones isomórficas con Javascript,» *Revista Interfases*, nº 8, pp. 143-161, 2015.
- [22] A. Mesbah, «Advances in Testing JavaScript-Based Web Applications,» *Advances in Computers*, vol. 97, pp. 201-235, 2015.
- [23] J. Molina, N. Loja, M. Zea y E. Loaiza, «Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python,» *Revista Latinoamericana de Ingeniería de Software*, vol. 4, nº 4, pp. 201-207, 2016.
- [24] G. Chanchí, L. Sierra y W. Campo, «Propuesta de una plataforma académica portable para la construcción de microservicios en entornos de IoT,» *Revista Ibérica de Sistemas e Tecnologías de Informação*, nº 27, pp. 1-13, 2020.
- [25] D. Tejada y E. Méndez, «Sistema de control académico basado en python y cloud computing, a nivel de secundaria en la ciudad de Santo Domingo,» Santo Domingo, 2020.

- [26] J. Ramos, O. Hernández y R. Silverio, «Sistema de supervisión para el monitoreo de redes eléctricas inteligentes,» *Ingeniería Energética*, vol. 40, nº 3, pp. 264-272, 2019.
- [27] Y. García Ruiz, V. López y G. Miñana, «Programación y uso de Python en el desarrollo del periodismo de datos.,» *TecCom Studies*, vol. 8, nº 6, pp. 25-33, 2017.
- [28] B. Yannik Jadoul, B. Thompson y B. Boer, «Introducing Parselmouth: A Python interface to Praat,» *Journal of Phonetics*, vol. 71, pp. 1-15, 2018.
- [29] G. J. Garcia, P. Gil, F. A. Candelas-Herías, M. J. Banes Payá, M. Á. Baquero Crespo, B. Alacid Soto y A. Torres Murcia, «Virtualización de Redes de Computadores con GNS3: Evaluación de soluciones para el aprendizaje a distancia,» Universidad de Alicante, Alicante, 2015.
- [30] J. Ortiz Zamora, J. J. Galiana-Merino, J. D. López Martí, V. Alavés Baeza, I. Sentana Gadea y E. Rincón Carrero, «Creación de nuevos escenarios prácticos para redes de comunicación,» Universitat d'Alacant. Institut de Ciències de l'Educació, Alicante, 2019.
- [31] J. Castro, S. Sepúlveda, B. Medina, D. Guevara y O. López-Bustamante, «Sistema de geolocalización de vehículos a través de la red GSM/GPRS y tecnología Arduino,» *Revista EIA*, vol. 16, nº 31, pp. 145-157, 2019.
- [32] h. Tao, Z. Zhen, Y. Peng, L. Dong, L. Ziyong, R. Quan, H. Yuxiang y L. Julong, «SEAPP: A secure application management framework based on REST API access control in SDN-enabled cloud environment,» *Journal of Parallel and Distributed Computing*, vol. 147, pp. 108-123, 2020.
- [33] K. Rodríguez, «Desarrollo de un envoltorio del api-rest de mendeley con GraphQL,» 19 Marzon 2020. [En línea]. Available: <http://repositorio.utn.edu.ec/handle/123456789/10292>. [Último acceso: 4 Diciembre 2020].

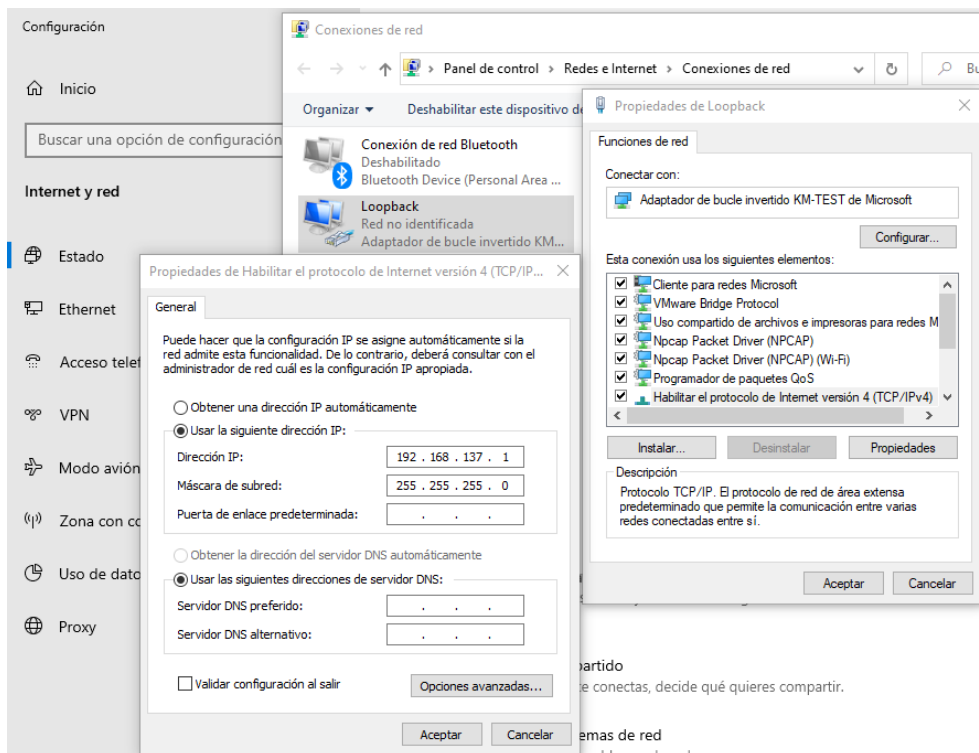
Anexos

Anexo 1: Descarga de imágenes CHR Mikrotik para su virtualización den el Software GNS3



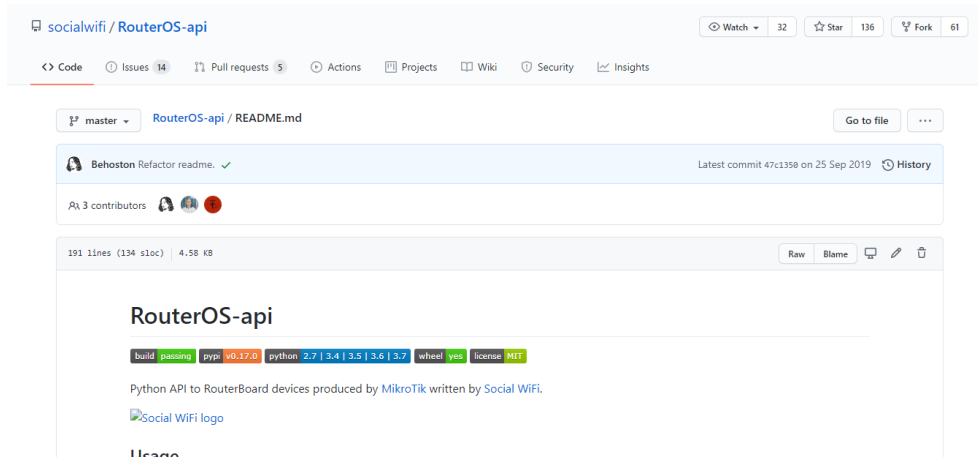
	6.46.8 (Long-term)	6.47.8 (Stable)	6.48beta58 (Testing)	7.1beta3 (Development)
Images vmdk, vhdx, vdi, ova, img				
Main package				
VHDX image				
VMDK image				
VDI image				
OVA template				
Raw disk image				
Extra packages				
The Dude server				-
The Dude client				-
Changelog				
Checksum	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Anexo 2: Configuración de adaptador de red Loopback para conexión con los routers virtuales en el software GNS3



Anexo 3: Página para la obtención de RouterOS-api desarrollada en Python y utilizada en el prototipo tecnológico.

<https://github.com/socialwifi/RouterOS-api/blob/master/README.md>



The screenshot shows the GitHub repository page for 'socialwifi/RouterOS-api'. The repository has 32 watchers, 136 stars, and 61 forks. The current branch is 'master', and the file being viewed is 'RouterOS-api/README.md'. The latest commit was made by Behoston on 25 Sep 2019. The README content includes the title 'RouterOS-api', a badge for 'build passing', and a list of supported Python versions (2.7, 3.4, 3.5, 3.6, 3.7). It also mentions that the API is for RouterBoard devices produced by Mikrotik and written by Social WiFi. The Social WiFi logo is displayed at the bottom of the README content.