

Evaluación de Software

Fecha: 8 de mayo de 2025

Nombres y Apellidos: Alexander Valentin Gordillo Rojas

Nota:

Instrucciones:

Utilice las siguientes credenciales para realizar las actividades solicitadas:

- Host (router): **`https://hga09pj16jk.sn.mynetname.net`**
- Usuario: **`1Prueb@USr`**
- Contraseña: **`Ap1P@ssw0rd.1S3gurO`**

Desarrolle cada actividad en un entorno funcional utilizando el framework que domine. Marque si la tarea fue realizada y adjunte la evidencia correspondiente para cada ítem.

Nota: Cada actividad tiene un valor de 2 puntos. Si implementa la solución utilizando Python (backend) y Angular (frontend), obtendrá 2 puntos adicionales como bonificación en la calificación final.

Actividades

1. ¿Implementó la autenticación básica (Basic Auth) para acceder al endpoint **`https://{router}}/rest/user`** utilizando el método GET en el backend?

☒ Realizado ☐ No realizado Calificación: _____

```
host = "https://hga09pj16jk.sn.mynetname.net"
```

```
endpoint = "/rest/user"
```

```
url = host + endpoint
```

```
usuario = "1Prueb@USr"
```

```
contraseña = "Ap1P@ssw0rd.1S3gurO"
```

```
response = requests.get(url, auth=HTTPBasicAuth(usuario, contraseña),  
verify=False)
```

```
print(f"Status Code: {response.status_code}")
```

```
print("Contenido:")
```

```
print(response.json())
```

2. ¿Consumió correctamente los datos del endpoint ***https://{router}}/rest/ppp/secret*** usando el método GET y mostró la respuesta en la consola del backend?

☒ Realizado

☐ No realizado

Calificación: _____

```
import requests
from requests.auth import HTTPBasicAuth

host = "https://hga09pj16jk.sn.mynetname.net"
endpoint = "/rest/ppp/secret"
url = host + endpoint

usuario = "1Prueb@USr"
contraseña = "Ap1P@ssw0rd.1S3gurO"

try:
    response = requests.get(url, auth=HTTPBasicAuth(usuario, contraseña), verify=False)

    if response.status_code == 200:
        print("Conexión exitosa. Datos obtenidos del endpoint /rest/ppp/secret:\n")
        print(response.json())
    else:
        print(f"Error: {response.status_code} - {response.json()}")
except Exception as e:
    print(f"Excepción al conectar: {e}")
```

3. ¿Realizó exitosamente la creación de un nuevo registro en el endpoint ***https://{router}}/rest/ppp/secret*** usando el método PUT con los siguientes parámetros?
- a. name: alias definido en texto plano
 - b. password: clave definida en texto plano
 - c. service: "pppoe"
 - d. profile: "Home-120Mbps"
 - e. remote-address: "192.168.121.15"
 - f. comment: nombre completo del postulante en texto plano

☒ Realizado

☐ No realizado

Calificación: _____

```
import requests
from requests.auth import HTTPBasicAuth
```

```
host = "https://hga09pj16jk.sn.mynetname.net"
endpoint = "/rest/ppp/secret"
url = host + endpoint
```

```
usuario = "1Prueb@USr"
contraseña = "Ap1P@ssw0rd.1S3gurO"
```

```
nuevo_usuario = {
    "name": "Valen",
    "password": "claveSegura123",
    "service": "pppoe",
    "profile": "Home-120Mbps",
    "remote-address": "192.168.121.15",
    "comment": "AlexanderValentinGordilloRojas"
}
```

```
response = requests.put(url, json=nuevo_usuario, auth=HTTPBasicAuth(usuario,
contraseña), verify=False)
```

```
if response.status_code in [200, 201]:
    print("Registro creado correctamente en /rest/ppp/secret.")
    print("Respuesta del router:")
    print(response.text)
else:
```

```
print(f"Error al crear el registro: {response.status_code}")  
print(response.text)
```

4. ¿Visualizó los datos obtenidos del EndPoint **/ppp/secret** en una vista del proyecto frontend, utilizando bindings dinámicos?

☒ Realizado

☐ No realizado

Calificación: _____

```
import requests
from requests.auth import HTTPBasicAuth
from flask import Flask, render_template, request

host = "https://hga09pj16jk.sn.mynetname.net"
endpoint = "/rest/ppp/secret"
url = host + endpoint

usuario = "1Prueb@USr"
contraseña = "Ap1P@ssw0rd.1S3gurO"

app = Flask(__name__)

@app.route("/")
def inicio():
    responsee = requests.get(url, auth=HTTPBasicAuth(usuario, contraseña),
    verify=False)
    response = responsee.json()
    return render_template("index.html", posts=response)

@app.route("/buscar")
def buscar():
    nombre = request.args.get("nombre", "")
    responsee = requests.get(url, auth=HTTPBasicAuth(usuario, contraseña),
    verify=False)
    response = responsee.json()

    resultados = [d for d in response if nombre.lower() in d["name"].lower()]
    return render_template("index.html", posts=resultados)

if __name__ == "__main__":
    app.run(debug=True)
```

Listado de Posts

Buscar por nombre

- **Usuario: isma**
Contraseña: isma
Servicio: pppoe
Perfil: PLAN 400 Mbps
Comentario: PRUEBAS NOC - ISMAEL
- **Usuario: cabs**
Contraseña: clave-pppoe-segura
Servicio: pppoe
Perfil: Home-120Mbps
Comentario: C. Andres Benavides
- **Usuario: Secreto 001**
Contraseña: Key001
Servicio: any
Perfil: default
Comentario:
- **Usuario: Valen**
Contraseña: claveSegura123
Servicio: pppoe
Perfil: Home-120Mbps
Comentario: AlexanderValentinGordilloRojas
- **Usuario: usuario_demo**
Contraseña: ClaveD3m0
Servicio: pppoe
Perfil: Home-120Mbps
Comentario: Juan Pérez
- **Usuario: luis_alias**

5. ¿Implementó un sistema de filtrado por el parámetro name dentro de la vista que muestra los datos del EndPoint **/ppp/secret** en el frontend?

☒ Realizado

☐ No realizado

Calificación: _____

```
import requests
from requests.auth import HTTPBasicAuth
from flask import Flask, render_template, request

host = "https://hga09pj16jk.sn.mynetname.net"
endpoint = "/rest/ppp/secret"
url = host + endpoint

usuario = "1Prueb@USr"
contraseña = "Ap1P@ssw0rd.1S3gurO"

app = Flask(__name__)

@app.route("/")
def inicio():
    responsee = requests.get(url, auth=HTTPBasicAuth(usuario, contraseña), verify=False)
    response = responsee.json()
    return render_template("index.html", posts=response)
```

```
@app.route("/buscar")
def buscar():
    nombre = request.args.get("nombre", "")
    responsee = requests.get(url, auth=HTTPBasicAuth(usuario, contraseña), verify=False)
    response = responsee.json()

    resultados = [d for d in response if nombre.lower() in d["name"].lower()]
    return render_template("index.html", posts=resultados)

if __name__ == "__main__":
    app.run(debug=True)
```

Listado de Posts

- **Usuario: Valen**

Contraseña: claveSegura123

Servicio: pppoe

Perfil: Home-120Mbps

Comentario: AlexanderValentinGordilloRojas

FIRMA:

Nombres y Apellidos: Alexander Valentin Gordillo Rojas

C.I. 1004178412