

API

Evidencia

GA7-220501096-AA5-EV02

Presentado

Cristian Baldovino

Análisis y Desarrollo de Software

Ficha: 2721400

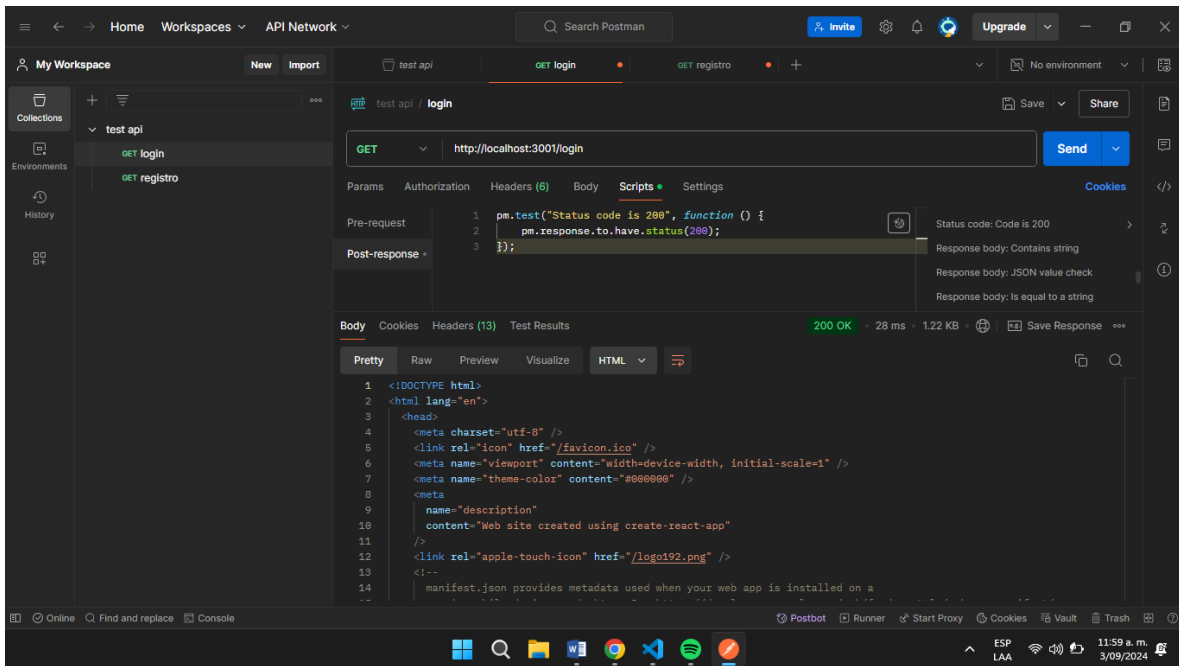
Instructora

Jovanna Erazo

Introducción

Basándonos en la evidencia anterior, se harán las respectivas pruebas con los EndPoints generados usando Postman y también se podrá visualizar a través de un video.

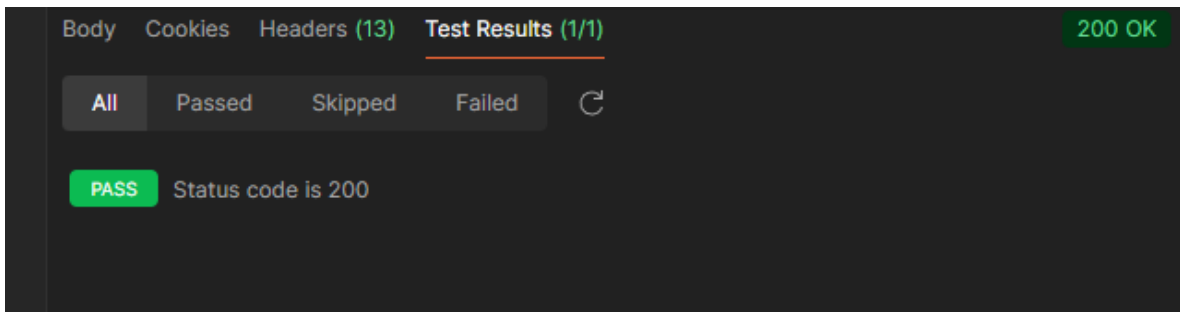
Test en Postman



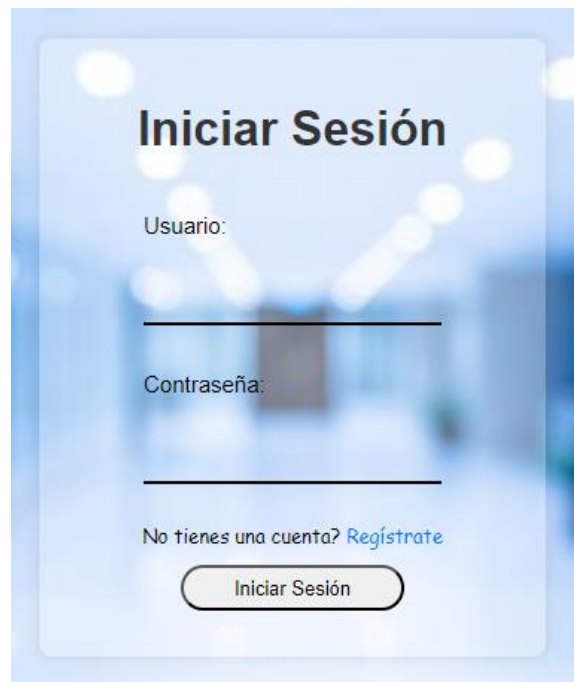
Realizamos pruebas en postman para la verificación y respuesta de la web de formulario y inicio de sesión, utilizando los siguientes scripts:

```
pm.test("Status code is 200", function () {  
  pm.response.to.have.status(200);  
});
```

Con este test verificamos que la solicitud ha tenido éxito. Es un código de respuesta de estado satisfactorio



Podemos visualizar como efectivamente pasa todos los Test realizados



```
JS Login.js x
src > componentes > JS Login.js > [x] Login
1 // src/Login.js
2 import React, { useState } from 'react';
3 import { authenticateUser } from './authentication';
4
5 const Login = () => {
6   // Estado para los campos de entrada
7   const [username, setUsername] = useState('');
8   const [password, setPassword] = useState('');
9   const [message, setMessage] = useState('');
10
11   // Manejar la lógica de inicio de sesión
12   const handleLogin = (e) => {
13     e.preventDefault();
14
15     // Llamar al servicio de autenticación
16     const result = authenticateUser(username, password);
17
18     // Mostrar el mensaje de resultado
19     setMessage(result.message);
20   };
21 }
```

Registro

Usuario:

Contraseña:

Confirmar Contraseña:

```
JS Register.js x
src > componentes > JS Register.js > Register
1 // src/Register.js
2 import React, { useState } from 'react';
3 import { registerUser } from './authentication';
4
5 const Register = () => {
6   // Estado para los campos de entrada
7   const [username, setUsername] = useState('');
8   const [password, setPassword] = useState('');
9   const [confirmPassword, setConfirmPassword] = useState('');
10  const [message, setMessage] = useState('');
11
12
13  // Manejar la lógica de registro
14  const handleRegister = (e) => {
15    e.preventDefault();
16
17    // Llamar al servicio de registro
18    const result = registerUser(username, password);
19
20    // Mostrar el mensaje de resultado
21    ...
22  }
23
24  return (
25    <div>
26      <input type="text" value={username} onChange={setUsername} />
27      <input type="password" value={password} onChange={setPassword} />
28      <input type="password" value={confirmPassword} onChange={setConfirmPassword} />
29      <button onClick={handleRegister}>Registrarse</button>
30    </div>
31  );
32
33 export default Register;
```

CONCLUSION

Se dio una breve explicación de parte del código elaborado con REACT para la evidencia realizando lo solicitado donde vimos el login y registro de usuarios el código completo estará en el ZIP donde se podrá visualizar la funcionalidad y el resto de comentarios de acuerdo a lo solicitado en la evidencia , también las pruebas realizadas con postman con sus endpoints y GET verificando el debido funcionamiento de la api