Procedimientos de Prueba:

## Para llevar a cabo las pruebas de conexión y autenticación, se comienza creando un archivo de pruebas para cada funcionalidad con el sufijo "Test". Por ejemplo, para evaluar el archivo 'DatabaseConnection.php', se genera un archivo denominado 'DatabaseConnectionTest.php'. En el caso de la prueba de conexión a la base de datos, se centra en el archivo 'DatabaseConnection.php'. Para ello, se crea un archivo 'DatabaseConnectionTest.php' y se construye una clase de prueba que recibe los parámetros de acceso a la base de datos.

## Prueba de Conexión a la base de datos:

La prueba de conexión a la base de datos se centra en el archivo 'DatabaseConnection.php' y, por lo tanto, se crea un archivo de prueba llamado 'DatabaseConnectionTest.php'. En este archivo, se define una clase de prueba que toma como parámetros los datos necesarios para la conexión a la base de datos.

En profundidad, primeramente, se importa el archivo la cual realizará la conexión a la base de datos, después crearemos una clase donde realizaremos las pruebas y dentro de esta clase tendrá un método donde ocurrirá la prueba a la conexión de la base de datos. Dentro de esta función se le pasaran las variables fundamentales para realizar la conexión a la base de datos de manera exitosa.

Guardaremos los datos en variables para la conexión a la base de datos, donde tendremos como host ‘localhost’, como usuario ‘root’, como contraseña ‘’ ya que la base de datos no tiene contraseña, y por ultimo la base de datos a conectar ‘leonbicicletas’.

Después crearemos un objeto de tipo databaseConnection que es la clase a probar, a esta clase la crearemos con las variables anteriormente creadas como host, usuario, contraseñas y base de datos.

Para finalizar realizaremos la conexión a la base de datos con el método para conectarla llamada ‘getConnection()’ y lo guardaremos en una variable donde verificaremos si se realizó correctamente la conexión a la base de datos.

Texto

Descripción generada automáticamente

## Prueba de Login a la base de datos:

Para realizar una prueba al Login de la base de datos se creará un archivo ‘Test’ donde se probará el modulo de inicio de sesión, donde se creará una clase con métodos donde se le pasarán los datos existentes de un usuario y verificara que los datos son correctos y crear la sesión a la base de datos.

Primeramente, para realizar la prueba de Login a la base de datos se creará el archivo ‘AuthControllerTest.php’ ya que se probará al archivo ‘AuthController.php’. Con el archivo ‘AuthControllerTest.php’ posteriormente creado, se importa el archivo la cual realizará el Login a la base de datos.

Dentro del archivo de pruebas ‘AuthControllerTest.php’ se crea una clase de pruebas donde se realizará las pruebas, dentro de esta clase tendrá diferentes métodos donde se probará diferentes formas de Login, una prueba con credenciales validas y otra con credenciales invalidas.

Texto

Descripción generada automáticamente

### Prueba con credenciales validas:

En la prueba de credenciales validas, se crea una conexión a la base de datos exitosa, dicha base de datos tiene guardado el usuario y la contraseña para ingresar. Después crearemos un objeto AuthController la cual autentificará si le pasaremos correctamente el usuario y la contraseña de la base de datos.

Para el usuario y contraseña guardaremos en variables el usuario y la contraseña que sabemos que son las correctas para ingresar a la base de datos.

Por último verificaremos si la autentificación a la base de datos es correcta con el método login de AuthController la cual se le pasará como parámetro el usuario, la contraseña y la base de datos a ingresar. Si todo esta correcto tendremos una verificación exitosa en la prueba con credenciales validas.

### Prueba con credenciales invalidas:

En la prueba de credenciales invalidas, se crea una conexión a la base de datos exitosa, dicha base de datos tiene guardado el usuario y la contraseña para ingresar. Después crearemos un objeto AuthController la cual autentificará si le pasaremos correctamente el usuario y la contraseña de la base de datos.

Para el usuario y contraseña guardaremos en variables el usuario y la contraseña que sabemos que son incorrectas para ingresar a la base de datos.

Por último, verificaremos si la autentificación a la base de datos es correcta con el método login de AuthController la cual se le pasará como parámetro el usuario, la contraseña y la base de datos a ingresar. Si efectivamente, el método de login de AuthController no nos deja conectarnos a la base de datos, tendremos una prueba exitosa en la prueba con credenciales invalidas. Esto es porque estamos verificando que el ingreso sea invalido a nuestra base de datos.

# Resultados de Pruebas:

Los resultados de la prueba son los esperados, en el apartado de la conexión tenemos que la función para crear la conexión a partir de una instancia de la clase de configuración cumple su propósito a cabalidad, permitiéndonos trabajar con una variable que nos ahorra tener que repetir estos mismos datos en diferentes partes del código.

## Resultados de Prueba a la conexión de la base de datos.

Los resultados de la prueba de conexión a una base de datos en PHP son una parte fundamental del proceso de desarrollo de aplicaciones. En este contexto, la realización de pruebas de conexión es esencial para garantizar que la comunicación entre la aplicación y la base de datos sea fluida y segura. Estos resultados se presentan en forma de "OK (1 test, 1 assertion)", lo que indica que la prueba se ha ejecutado con éxito y ha cumplido con su objetivo, esto ya que ingresamos correctamente el host, usuario, contraseña y base de datos y se verificó que el archivo para la conexión de datos funciona correctamente.

El resultado de la prueba en cuestión fue "OK (1 test, 1 assertion)." Esto indica que la conexión a la base de datos se estableció de manera exitosa. Dicho de otra manera, se pudo acceder a la base de datos utilizando los valores proporcionados para el host, usuario, contraseña y base de datos. La prueba tuvo un solo "assertion," lo que significa que se evaluó una única afirmación y se obtuvo el resultado esperado.Texto

Descripción generada automáticamente

Aunque la prueba haya arrojado resultados positivos, es fundamental continuar realizando pruebas periódicas de conexión a la base de datos. Los cambios en la infraestructura, contraseñas o configuración de la base de datos podrían afectar la conectividad en el futuro. La realización regular de pruebas de conexión ayuda a garantizar que la aplicación siga funcionando sin problemas.

## Resultados de prueba de Login:

Los resultados de tus pruebas de inicio de sesión en la base de datos en PHP son los siguientes: "OK (2 tests, 2 assertions)." Este resultado es indicativo de que ambas pruebas se ejecutaron con éxito y cumplieron sus criterios de validación. A continuación, se presenta una descripción de los resultados:

Se realizaron dos pruebas distintas para evaluar el comportamiento del proceso de autenticación. Cada prueba se centró en una situación específica: la primera prueba se diseñó para verificar una conexión exitosa utilizando credenciales válidas, mientras que la segunda prueba se diseñó para validar una conexión fallida con credenciales inválidas. Ambas pruebas se ejecutaron sin inconvenientes, y los resultados obtenidos indicaron que el proceso de autenticación se comporta según lo esperado.

El primer método de prueba utilizó la afirmación assertTrue para confirmar que la conexión se estableció con éxito utilizando credenciales válidas. La aserción fue exitosa, lo que significa que la conexión se verificó como válida, lo que es un resultado positivo.

El segundo método de prueba empleó assertFalse para evaluar una conexión fallida con credenciales inválidas. La aserción también se completó de manera exitosa, lo que indica que el sistema rechazó correctamente el intento de inicio de sesión con credenciales no válidas, lo que es un comportamiento esperado.

Texto

Descripción generada automáticamente

Los resultados actuales de las pruebas indican un inicio de sesión exitoso con credenciales válidas y una detección adecuada de credenciales inválidas. Sin embargo, es crucial recordar que las pruebas de autenticación deben ser continuas y abarcar casos más complejos, incluyendo pruebas de rendimiento y seguridad, para mantener la robustez del sistema a medida que evoluciona y enfrenta desafíos futuros.