readme.md for @push.rocks/smartwebdav

A package for interacting with WebDAV servers, providing easy file and directory operations.

Install

To install `@push.rocks/smartwebdav` in your project, use npm or Yarn as follows:

```
npm install @push.rocks/smartwebdav
```

or

```
yarn add @push.rocks/smartwebdav
```

Usage

The `@push.rocks/smartwebdav` package offers a high-level abstraction for interacting with WebDAV servers. It integrates seamlessly with the `@push.rocks/smartfile` ecosystem for file handling, and `@push.rocks/smartpath` for path operations, providing a straightforward interface for file and directory operations on a remote WebDAV server.

First, you must import the package and other necessary libraries in your TypeScript file:

```
import { WebdavClient, authType } from '@push.rocks/smartwebdav';
import * as smartfile from '@push.rocks/smartfile';
```

Creating a WebDAV Client Instance

Before performing any operations, create an instance of `WebdavClient`. You need to provide server details including the URL, authentication type, and credentials (if required).

```
const webdavClient = new WebdavClient({
  serverUrl: 'https://your.webdavserver.com/',
  authType: authType.Password,
  username: 'your_username',
  password: 'your_password'
});
```

Listing Directory Contents

To list the contents of a directory on your WebDAV server:

```
async function listDirectoryContents() {
  const contents = await webdavClient.listDirectory('/path/to/directory');
  console.log(contents);
}

listDirectoryContents();
```

Creating Directories

Ensure a directory exists (creating it and its parents if necessary):

```
async function ensureDirectory() {
  await webdavClient.ensureDirectory('/path/to/ensure');
  console.log('Directory ensured.');
}

ensureDirectory();
```

Uploading Files

Uploading files is made easy by integrating with the `@push.rocks/smartfile` package. This allows for uploading from various sources like local files, buffers, and more.

```
async function uploadFiles() {
  const smartFiles = await smartfile.smartfileLocal.getSmartfileArrayFromGlob([
    'path/to/local/files/**/*'
```

```
  ]);

  await webdavClient.uploadSmartFileArray(smartFiles);
  console.log('Files uploaded successfully.');
}

uploadFiles();
```

Deleting Files and Directories

You can also delete specific files or entire directories:

```
async function deleteDirectory() {
  await webdavClient.deleteDirectory('/path/to/delete');
  console.log('Directory deleted.');
}

deleteDirectory();
```

Moving Files

Moving files from one location to another is also supported:

```
async function moveFile() {
  await webdavClient.move('/path/to/file.txt', '/new/path/to/file.txt');
  console.log('File moved.');
}

moveFile();
```

Handling Errors

When performing WebDAV operations, it's crucial to handle potential errors. This can involve retrying operations or logging errors for later analysis. For instance, ensuring directory creation could include error handling as follows:

```
async function ensureDirectorySafe() {
  try {
```

```
    await webdavClient.ensureDirectory('/path/to/ensure');
    console.log('Directory ensured safely.');
  } catch (error) {
    console.error('Failed to ensure directory:', error);
    // Implement retry logic or error handling here
  }
}

ensureDirectorySafe();
```

By leveraging `@push.rocks/smartwebdav`, developers can simplify complex WebDAV interactions, making it easier to manage remote file and directory operations in TypeScript applications. The combination of easy-to-use methods and integration with powerful file handling packages provides a robust solution for working with WebDAV servers.

License and Legal Information

Trademarks

Company Information

Task Venture Capital GmbH
Registered at District court Bremen HRB 35230 HB, Germany

For any legal inquiries or if you require further information, please contact us via email at hello@task.vc.

By using this repository, you acknowledge that you have read this section, agree to comply with its terms, and understand that the licensing of the code does not imply endorsement by Task Venture Capital GmbH of any derivative works.

Revision #3
Created 2026-03-28 11:13:35 UTC by foss.global Team
Updated 2026-03-28 12:20:20 UTC by foss.global Team