readme.md for @design.estate/dees-editor

an advanced editor for markdown documents based on monaco.

Install

To install `@design.estate/dees-editor`, you need to run the following command using npm or Yarn. Make sure you have Node.js installed on your system before you proceed.

```
npm install @design.estate/dees-editor
# or if you prefer Yarn
yarn add @design.estate/dees-editor
```

Usage

This guide walks you through the usage of `@design.estate/dees-editor`, an advanced editor for markdown documents based on Monaco Editor. This editor not only allows for high customization but also enhances your markdown editing with the robust features of Monaco Editor.

Integrating `@design.estate/dees-editor` in Your Project

First, ensure you have installed `@design.estate/dees-editor` as described in the installation section.

Setting Up Your Project

Create a TypeScript file (e.g. `app.ts`) and import necessary functionalities using ESM syntax:

```
import { DeesEditor, DeesTerminal, DeesEditorMarkdown, DeesEditorMarkdownOutlet } from
'@design.estate/dees-editor';
```

Displaying the Markdown Editor

To display the markdown editor in your web application, you will need to incorporate it into your HTML structure. Typically, you can use custom element tags provided by the library. Here's a simple example:

HTML Structure

Add the following HTML structure to your application's main HTML file (e.g., `index.html`):

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Markdown Editor</title>
    <script type="module" src="./app.ts"></script>
</head>
<body>
    <dees-editor></dees-editor>
</body>
</html>
```

Initializing the Editor

Within your `app.ts` file, initialize the markdown editor with desired options:

```
const editorElement = document.querySelector('dees-editor');
if (editorElement) {
    editorElement.language = 'markdown'; // Set the language to markdown
    editorElement.content = '# Markdown Editor\n\nStart typing your markdown content here...';
// Initial content
}
```

Advanced Usage

`@design.estate/dees-editor` offers an advanced markdown editor that leverages the Monaco Editor. This means you can utilize Monaco's extensive API for further customization and functionality enhancement.

Configuring the Editor

The editor can be tailored to meet your needs, such as theming, read-only mode, and more. Explore Monaco's `IStandaloneEditorConstructionOptions` for all available configurations.

Example - Setting Dark Theme and Read-Only Mode:

```
(async () => {
    const editorInstance = await editorElement.editorDeferred.promise; // Wait for the editor to initialize
    editorInstance.updateOptions({
        theme: 'vs-dark', // Set theme to dark
        readOnly: true // Make the editor read-only
    });
})();
```

Integrating Terminal Emulator

`@design.estate/dees-editor` also includes a powerful terminal emulator, `DeesTerminal`, which can be embedded alongside your editor:

```
<dees-terminal></dees-terminal>
```

And in your TypeScript file:

```
const terminalElement = document.querySelector('dees-terminal');
if (terminalElement) {
    // Initialize terminal with options or use it directly
}
```

Creating a Live Markdown Preview

`@design.estate/dees-editor` facilitates creating live markdown previews. Utilize `DeesEditorMarkdown` for binding the editor with a live preview outlet.

```
<dees-editormarkdown></dees-editormarkdown>
```

This component divides the viewport into an editor pane and a live preview pane. As you type in the editor, the markdown content automatically renders in the live preview.

Conclusion

`@design.estate/dees-editor` is a feature-rich markdown editor built on top of Monaco Editor, offering extensive customizability and advanced features such as a terminal emulator and live markdown preview. Its integration into your project is straightforward, whether embedding the editor directly into your webpage or leveraging its components for advanced functionality. The versatility of Monaco Editor ensures that your markdown documentation needs are met with efficiency and style.

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.

Revision #6
Created 2026-03-28 10:49:08 UTC by foss.global Team
Updated 2026-03-28 12:14:31 UTC by foss.global Team