

Custom HTTP API Source - Developer/Admin Guide

TODO: in the thesis text I should also state the benefits of having a custom HTTP API source (custom authorization, self-hosting, etc.)

API Requirements

Your HTTP API needs to provide **two endpoints**:

1. Tree Endpoint - Directory Structure

Endpoint: GET /tree?branch={branch}

Returns the complete directory structure of your code repository.

Response Format:

```
[
  {
    "name": "folder-name",
    "path": "whole/path/to/folder",
    "type": "folder",
    "children": [
      {
        "name": "file.ts",
        "path": "whole/path/to/folder/file.ts",
        "type": "file" | "folder",
        "children": [],
        "isExpanded": false
      }
    ],
    "isExpanded": false
  }
]
```

Field Descriptions:

- `name` (string): The name of the file or folder
- `path` (string): The full path from the repository root
- `type` (string): Either "file" or "folder"
- `children` (array): Array of child nodes (empty for files)
- `isExpanded` (boolean): Default false (UI will manage this)

2. File Endpoint - File Contents

Endpoint: GET `/file?branch={branch}&path={filePath}`

Returns the content and metadata for a specific file.

Response Format:

```
{
  "displayType": "text",
  "content": "The actual file content as a string",
  "downloadUrl": "https://example.com/download/path/to/file.ts",
  "previewUrl": null,
  "fileName": "file.ts"
}
```

Field Descriptions:

TODO: the display type could be improved to the classic "mime type" format, not the custom strings I use

- `displayType` (string): One of: "text", "image", "pdf", "binary"
- `content` (string|null): The file content for text files, null for others
- `downloadUrl` (string|null): A URL to download the file
- `previewUrl` (string|null): A URL to preview images/PDFs
- `fileName` (string): The name of the file