

Action Plan:

1. Group Documentation Repositories Separately:

- We should create distinct documentation repositories for OnTrack, Splashkit, and Courseflow.
- Each repository should include all relevant files, ensuring clear separation between projects to avoid overwhelming new students.

2. Remove Duplication:

- We need to consolidate redundant files across repositories and within folders.
- A major example is the duplication between documentation-1 and documentation-2, where
 1. 90% of the content is identical across both repositories.
 2. Files like index.md appear multiple times with little to no unique content.
 3. Unique content, such as the **Password Guideline** in documentation-1 but missing in documentation-2, should be merged into one repository.
- Eliminating these redundancies will make the documentation easier to maintain and more efficient to use.

3. Rename Files and Add Brief Descriptions:

- We need to rename files to more intuitive names. For example
 1. Change ambiguous names like Aspose.Words.c1eae0e... to meaningful ones such as contribution-guide.png.
 2. Replace generic names like image1.png with descriptive names like project-dashboard.png.
- We should add a one-line brief description for each repository and file to help new students quickly understand their purpose.

4. Ensure Consistent Repository Format:

- We should unify the structure of all repositories. For example
 1. Use consistent naming conventions, such as lowercase with hyphens for file names (e.g., courseflow-dfd-0.png).
 2. Add a README.md file to directories like src to explain the folder structure and its purpose.

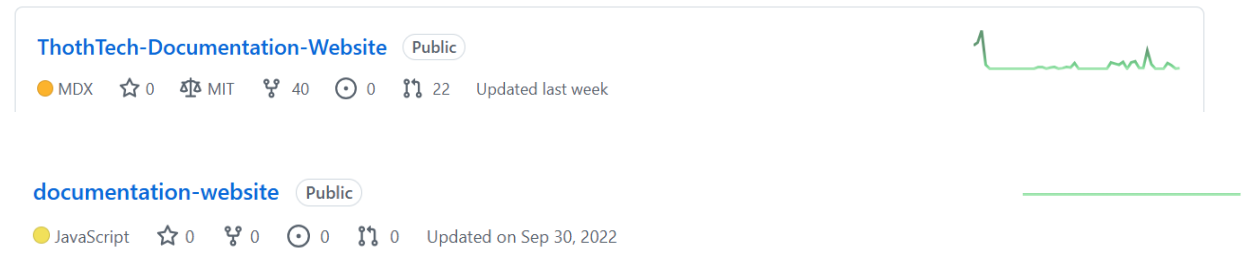
5. Correct All Minor Discrepancies:

- We need to address all documented minor inconsistencies, such as
 1. Resolving naming mismatches like setting.md vs. "settings" in the SPIS list.
 2. Harmonizing endpoint naming conventions to follow singular/plural rules for better clarity.
 3. Aligning duplicated API content, such as those in tutorials.md and similar files across repositories.

By following this action plan, we can make the documentation more organized, user-friendly, and easier to maintain, while minimizing confusion for new users.

Here is the list of major discrepancies in detail -

1. The names of the repositories don't clearly describe their contents, which can confuse a new student. For example, "doubtfire-astro" and "doubtfire-io."
2. There are no separate documentation repositories for OnTrack, Splashkit, and Courseflow, which can easily overwhelm a new student. We should consider creating three distinct documentation repositories for them.
3. Courseflow documentation is located within the "doubtfire-astro" repository, which adds to the confusion.
4. There is no one-line brief description for the "ThothTech-Documentation-Website" and "documentation-2" repositories.



Similarly, most files, except "doubtfire-astro" and "doubtfire-io," do not clearly indicate which project's documentation they contain. Even after opening these files, we only see folders like "github" or "src," but they still don't mention the project names that a new student might be searching for.

Minor discrepancy – There are numerous empty index.md files that we must consider deleting

Duplication

The documentation and documentations repos have a lot in common. Here are some of the similarities. It is important to note that I have not listed all the duplications, as 90% of the content is duplicated.

Product Solutions Resources Open Source Enterprise Pricing

thoth-tech / documentation Public

Notifications Fork 128

<> Code Issues Pull requests 53 Actions Projects Security Insights

Files

main

Go to file

> .astro

> .github

> workflows

ci.yml

pull_request_template.md

> .netlify

> .vale

> .vscode

> Arcade Games Bugs and Impro...

> Images

> docs

> .vale

> thothtech

vale-ison.tmdl

documentation / .github / workflows / ci.yml

tancnle ci: add Vale lint to CI b0f2f26 · 2 years ago

Code Blame 29 lines (26 loc) · 601 Bytes

Raw

```
1  name: CI
2
3  on:
4    pull_request:
5    push:
6      branches: [main]
7
8  jobs:
9    build:
10     runs-on: ubuntu-latest
11
12     steps:
13       - uses: actions/checkout@v2
14       - name: Linelint
15         uses: fernandrone/linelint@master
16       - name: Set up Node
17         uses: actions/setup-node@v1
18       with:
19         node-version: 16.14.2
20       - name: Install dependencies
```

documentation

Files

main

Go to file

> .github

> workflows

ci.yml

pull_request_template.md

> .vscode

> docs

.editorconfig

.gitignore

.nvmrc

.prettiignore

.prettierrc

.vale.ini

CONTRIBUTING.md

README.md

package-lock.json

package.json

documentations / .github / workflows / ci.yml

tancnle ci: add Vale lint to CI b0f2f26 · 2 years ago

Code Blame 29 lines (26 loc) · 601 Bytes

Raw

```
1  name: CI
2
3  on:
4    pull_request:
5    push:
6      branches: [main]
7
8  jobs:
9    build:
10     runs-on: ubuntu-latest
11
12     steps:
13       - uses: actions/checkout@v2
14       - name: Linelint
15         uses: fernandrone/linelint@master
16       - name: Set up Node
17         uses: actions/setup-node@v1
18       with:
19         node-version: 16.14.2
20       - name: Install dependencies
21         run: npm install
22       - name: Prettier
23         run: npm run format:check
24       - name: Vale
25         run: |
26           sudo apt update
27           sudo apt install snapd
28           sudo snap install vale --edge
29           npm run prose:check
```

documentations

Product Solutions Resources Open Source Enterprise Pricing

thoth-tech / documentation Public

Notifications Fork 128 Star 6

Code Issues Pull requests 53 Actions Projects Security Insights

Files

main

Go to file

.astro

.github

workflows

pull_request_template.md

.netlify

.vale

.vscode

extensions.json

documentation / .vscode / extensions.json

tancnle feat: add Vale VSCode extension e2f0af3 · 2 years ago History

Code Blame 4 lines (4 loc) · 110 Bytes

Raw

```
1 {
2   "recommendations": ["errata-ai.vale-server", "esbenp.prettier-vscode"],
3   "unwantedRecommendations": []
4 }
```

documentation

Files

main

Go to file

.github

workflows

ci.yml

pull_request_template.md

.vscode

extensions.json

documentations / .vscode / extensions.json

tancnle feat: add Vale VSCode extension e2f0af3 · 2

Code Blame 4 lines (4 loc) · 110 Bytes

Raw

```
1 {
2   "recommendations": ["errata-ai.vale-server", "esbenp.prettier-vscode"],
3   "unwantedRecommendations": []
4 }
```

documentations

Consistent Naming Pattern

- We should use lowercase and hyphens for file names (e.g., courseflow-dfd-0.png instead of CourseFlow-DFD-0.png).

Rename Ambiguous Files

- We should rename files with unclear names like Aspose.Words.c1eaed0e... should be renamed to reflect their purpose (e.g., contribution-guide.png).
- We also need to replace generic names like image1.png with descriptive names, such as project-dashboard.png.

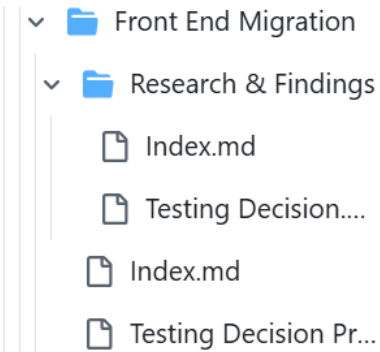
Src folder

- We should unify the directory structure across repositories to simplify navigation - **src/ assets/content/(docs/projects/products/refrence/teams-and-leadership/meeting-minutes) /styles**
- Unify Directory Structure -
Both repositories should adopt a consistent structure to simplify navigation:
- We should add a README.md in the src directory explaining its structure

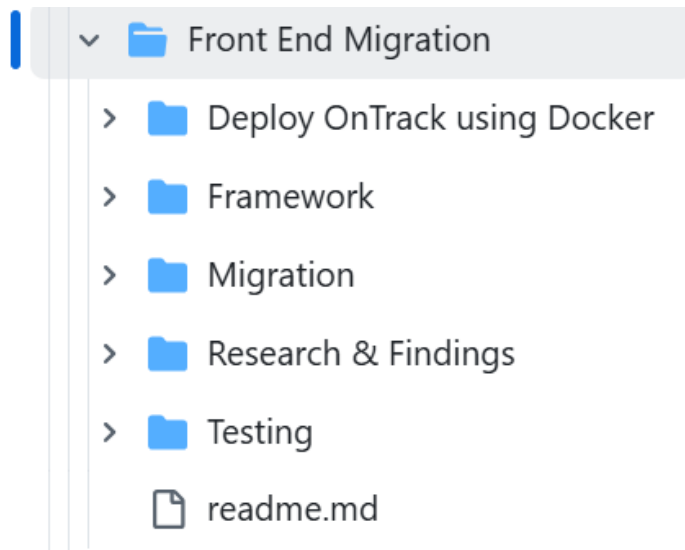
The only differences between documentation-1 and documentation-2 are as follows. As I have identified these, we should consider merging the unique content into one file and deleting the other -

1. Password Guideline:
 - Exists in the Company Documentation Team folder but is missing in documentation-2.
2. Empty index.md Files:
 - documentation-2 contains two empty index.md files, whereas documentation-1 has only one empty index.md file.

document-1 -



document-2 –



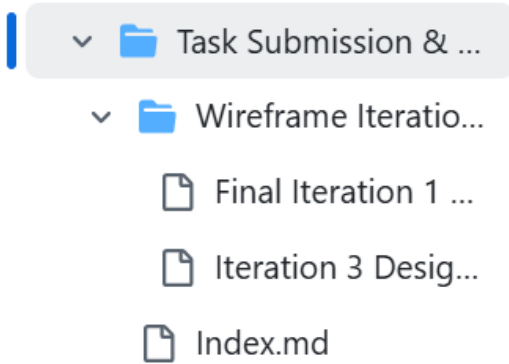
In documentation-1, the **Research & Findings** folder in the **Front End Migration** directory contains two very similar files: `Testing_Decision.md` and `Testing Decision Process.md`.

- Both files share **90% of their content**, making them largely redundant.
- The only difference is that `Testing Decision Process.md` includes an additional **conclusion section** and provides extra details, such as well-defined steps like `npm install` and `npm test`.

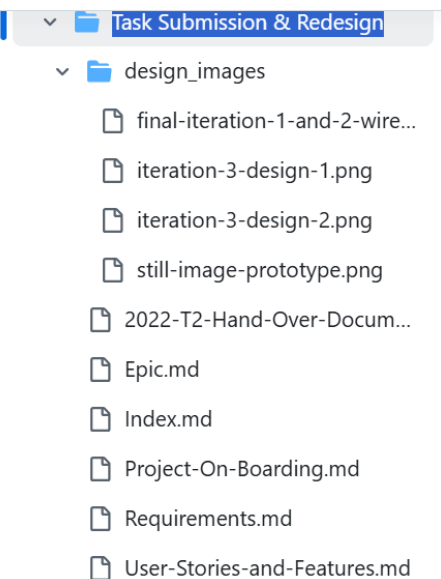
To avoid duplication, **one consolidated file** should be sufficient.

Task Submission & Redesign

In document-1



In document-2



Here again, every folder, sub-file, and the content of all files, except for the two differences mentioned above, are the same. We should consider consolidating them into **one repository** to eliminate redundancy.

Doubtfire-astro

Detailed and deep analysis

Firstly I checked all the links in all the files and can confirm that all the links were working.

1. Duplication

The files projects.md, projects_1.md, and projects_2.md all document similar API

endpoints for the "Projects" API, with overlapping or identical content. These redundancies should be consolidated into a single comprehensive file for easier maintenance and clarity.

Details:

- **Repeated Endpoints** The same endpoints, such as GET /api/projects/{id}, POST /api/projects, and PUT /api/projects/{id}, are listed across all files.

Action Needed

We need to combine overlapping endpoints into a single file for better clarity. We need to remove duplicate parameters and responses to avoid redundancy. We need to highlight unique functionalities while merging similar ones.

Minor Change in Names - setting.md

Overview

The setting.md file documents API endpoints related to settings, including privacy policy details and configurable frontend options.

- The file is named setting.md, but the API name in the specification list (SPIS list) uses "settings" (plural). This inconsistency between "setting" (singular) and "settings" (plural) could lead to confusion. It is recommended to standardize the naming to "settings" across both the documentation and the file name for consistency.

Action Needed

- We should resolve the inconsistency between setting.md and "settings" in the SPIS list.
- We need to consider standardizing the naming convention will improve clarity and make the documentation easier to understand and maintain.

Key Issues Identified:

1. Overlap with Other APIs:

- Endpoints like DELETE /api/units/{unit_id}/tutorials/{tutorial_abbr}/enrolments/{project_id} and POST /api/units/{unit_id}/tutorials/{tutorial_abbr}/enrolments/{project_id} are duplicated in tutorials.md. This redundancy could lead to maintenance issues and should be addressed.

2. Inconsistent Naming:

- Certain endpoint names, such as `/api/units/{unit_id}/task_definitions/task_pdfs`, lack clarity and consistency. Consider standardizing plural/singular conventions across all endpoint names to improve readability and understanding.