draftlint の設定を外部に保持する

背景

文章校正機能(draftlint)の設定はソースコードにべた書きしている。

- 設定変更 = ソースコードの変更 = リポジトリの変更
- 変更の履歴が残るのは良い
- ブランチを分けたりなどは面倒

もう少し手軽に許可リストを変更したい。

設定の保存先

保存先を変更することで回避できないか。

- 環境変数
  - 手軽だが構造のあるデータは保持しにくい
- ストレージ系のサービス
  - 構造の定義は自由だがフェッチ処理の追加が必要

CMS へ保存

CMS(コンテンツの保存先)に設定も保存する。

- 良い
  - コンテンツと設定がセットになる
- 良くない
  - リソースを消費(Hobby プランなのでその辺に敏感)

今回はリソースに余裕があるので CMS へ保存を選択。

CMS 設定の利用形態

CMS へ保存された設定をどのように取得し利用するか。

初期設定時

- npm スクリプトで fetch し `.json` として保存
- 設定変更時にスクリプト実行が必要
- CodeSandbox で実行すると保存されたファイルが見える

初期コストは低いが問題もある。

CMS 設定の利用形態

CMS へ保存された設定をどのように取得し利用するか。

リアルタイム

- ソースコードに fetch 処理追加が必要
- 設定変更は CMS で保存するだけ
- 基本的に設定が外部から見えることない

初期コストはそれなりだが問題は少ない。

現状の対応

現状では様子見の面もあるので、初期コストが低い方法を選択。

- 設定の保存先はコンテンツと同じ CMS
- 設定は初期化時にスクリプトで fetch
- rules (配列)のマージだけは専用の処理を作成

現状の対応

```
#!/bin/sh
set -e

curl "${API_BASE_URL}api/v1/config/draftlint-config?fields=config" -s -H "X-API-KEY:${GET_API_KEY}" \
  > src/\$draftlint-config.json
```

```
export function mergeRules(r1: any[], r2: any[]): any[] {
  if (r2) {
    const ret = new Array(r1.length);
    r1.forEach((r, i) => {
      const idx = r2.findIndex(({ ruleId }) => r.ruleId === ruleId);
      if (idx >= 0) {
        ret[i] = merge(r, r2[idx]);
        return;
      }
      ret[i] = r;
    });
```

その他

今後も、状況に応じて構成を変更することはある。

少し使ってみたが、「設定追加後のスクリプト実行」が既に面倒に感じている。

次ページからその後の所感的なものを追記していく。

所感

モバイル環境では難易度高すぎ

- ジョグのときに判明、モバイル環境では以下の問題がある
  - スマホで JSON の編集はかなり難易度高い
  - スクリプトの実行が難しい
    - Sandbox の操作が必要(ハイバネートに落としてからのスタートくらいか)

下書きに保存してしまいがち

- 許可ワードを追加するときはコンテンツを下書きで編集している
- その流れで、許可ワードを追加した後に「下書きを追加」で保存
- 設定を反映させようとコンテナの再起動を繰り返した後に気が付くことが多い

自分の不注意なので「慣れろ」という話ではあるが。