










AsciiDocを使った要求仕様書のテンプレートの要求仕様書

Table of Contents

テンプレートの主機能	1
[INPUT001]AsciiDocを使ったテンプレートにしたい 	1
[INPUT002]シンプルなディレクトリ構成にしたい 	1
[TRANS001]マクロを用意して要求仕様書の枠組みを簡単に書きたい 	2
[OUTPUT001]USDMを表現したい 	3
[UI001]要求と仕様のIDをAsciiDocのアンカーに設定し、ページ内リンクを簡単に貼れること 	3
GitHubと連携する機能	3
[GITHUB001]GitHubへIssueを新規投稿するためのリンクを、各要求と仕様の末尾に配置したい  ..	3
[GITHUB011]GitHubに投稿されているオープンなIssueを要求仕様IDで検索し、該当したIssueへのリンクを各要求仕様貼り付けたい 	4
[GITHUB021]GitHub Workflowを使って、GitHub Pageに要求仕様書を出力したい 	4
[GITHUB031]Pull Requestの作成をトリガーにGitHub Workflowを実行し、PRのコメントに要求仕様書のPDFを添付したい 	4
カテゴリー一覧	5

テンプレートの主機能

[INPUT001]AsciiDocを使ったテンプレートにしたい

[Issue #6](#), [Issue #3](#),

要求の補足	内容
理由	AsciiDocは * markdownよりも表現能力が高いため * GitHub ActionsなどのCI/CDツールでHTMLファイルを自動生成できるため
説明	AsciiDocのコンパイル環境を用意しなければならないというデメリットもある

[INPUT002]シンプルなディレクトリ構成にしたい

[Issue #8](#),

要求の補足	内容
理由	ディレクトリ構成が複雑だと、テンプレートの使い方を覚えるのが難しくなるため
説明	

- [INPUT002-01] テンプレートの最上位層に置く adoc ファイルは requirements_specifications.adoc のみとする

Issue #7, Issue #5,

- [INPUT002-02] chapters ディレクトリに子ページを置くこと
- [INPUT002-03] images ディレクトリに画像を置くこと
- [INPUT002-04] images ディレクトリを imagesdir 属性に設定すること

これにより、`image:画像.png[]` のように記述するだけで画像を表示できる



[TRANS001] マクロを用意して要求仕様書の枠組みを簡単に書きたい



要求の補足	内容
理由	AsciiDocの記法を最小限にするため
説明	AsciiDocの表の書き方は、機能が良い反面、記法が複雑であるため

- [TRANS001-01] 要求のマクロとして snippets/requirement.adoc を用意する

- [TRANS001-02] 仕様のマクロとしてsnippets/specification.adocを用意する 

[OUTPUT001]USDMを表現したい

要求の補足	内容
理由	USDMを使った要求仕様書を作成するのがこのテンプレートの目的であるため
説明	

- [OUTPUT001-01] 要求、理由、説明が記載できる 
- [OUTPUT001-02] 要求と仕様のそれぞれにIDを付与できる 

[UI001]要求と仕様のIDをAsciiDocのアンカーに設定し、ページ内リンクを簡単に貼れること

要求の補足	内容
理由	仕様書内で、ある仕様が別の仕様を参照することがよくあるため
説明	AsciiDocのアンカーを使うことで"<<ID>>"のような記法でリンクを貼れる



- [UI001-01] マクロファイル（[TRANS001]で定義したもの）にアンカーを設定すること 




GitHubと連携する機能

- ここではGitHubとの連携機能に関する要求仕様を記載する
- 要求仕様IDはGITHUB**とする

[GITHUB001]GitHubへIssueを新規投稿するためのリンクを、各要求と仕様の末尾に配置したい



要求の補足	内容
理由	要求仕様について改善提案や質問を簡単に投稿したいため
説明	

- [GITHUB001-01] 要求仕様書マクロファイル（[TRANS001]で定義したもの）に、投稿用のリンクを設定すること 
- [GITHUB001-11] 最上位層のadocファイル（[INPUT002-01]で定義したもの）で投稿先を設定できること 

- [GITHUB001-12] 投稿先がセットされていない場合、リンクが表示されないこと 
- [GITHUB001-21] 投稿するIssueのタイトルは[{要求また仕様のID}]で始まること 
- [GITHUB001-31] 投稿用のリンクを開くと、新しいウィンドウで投稿画面が表示されること 

[GITHUB011]GitHubに投稿されているオープンなIssueを要求仕様IDで検索し、該当したIssueへのリンクを各要求仕様に貼りたい

要求の補足	内容
理由	要求仕様に対してどれだけ問題定義されているのかをすぐに把握するため
説明	

- [GITHUB011-01] Issue検索しadocファイルを書き換えるスクリプトファイルを用意すること 
- [GITHUB011-02] スクリプトには、検索対象のリポジトリ名とadocファイルへのパスを指定できると 

[GITHUB021]GitHub Workflowを使って、GitHub Pageに要求仕様書を出力したい

要求の補足	内容
理由	AsciiDocファイルを更新するたびに要求仕様書を自動で生成するため
説明	

[GITHUB031]Pull Requestの作成をトリガーにGitHub Workflowを実行し、PRのコメントに要求仕様書のPDFを添付したい

要求の補足	内容
理由	レビュアーが要求仕様書を確認する手間を省くため
説明	

- [GITHUB031-01] asciidoctor-pdfのdockerイメージを使用してPDFを生成する 

カテゴリー一覧

記号	親カテゴリ	小カテゴリ
INPUT	主機能	入力処理 (Input)
OUTPUT	主機能	出力処理 (Output)
TRANS	主機能	変換処理 (Transformation)
CONT	主機能	制御処理 (Control)
UI	主機能	表示/UI処理 (UI)
DIAG	主機能	診断処理 (Diagnosis)
QFUNC	品質特性	機能性 (Functionability) 合目的性 正確性 接続性 整合性 セキュリティ
QRELI	品質特性	信頼性 (Reliability) 成熟性 障害許容性 回復性
QUSAB	品質特性	使用性 (Usability) 理解性 習得性 操作性
QEFFI	品質特性	効率性 (Efficiency) 実行効率性 資源効率性
QMAIN	品質特性	保守性 (Maintainability) 解析性 変更作業性 安定性 試験性
QPORT	品質特性	移植性 (Portability) 環境適応性 移植作業性 規格準拠性 置換性