



日本語ドキュメント作成環境

利用マニュアル 1.0 版

改版履歷

版数	改版日	改版内容
1.0	2016/10/15	初版

1. はじめに

日本語ドキュメント作成環境（以下「本環境」と表記）は`AsciiDoc`により日本語ドキュメントを作成するための環境です。バージョン管理が容易なテキストファイルから、印刷に耐え得る日本語ドキュメントのPDFを出力する事が目標です。

本環境は、ビルドツールとして`Gradle`を使用しています。また、`AsciiDoc`のRubyによる実装である`Asciidoctor`と、`Gradle`から`Asciidoctor`を利用可能とする`Asciidoctor Gradle Plugin`を使用しています。

本環境が使用している各プロダクトの詳細は「[参考文献](#)」に記載されている各サイトを参照してください。

1.1. このドキュメントの目的

本環境の使用方法を説明します。また、このドキュメント自体を、本環境を使用して作成したドキュメントのサンプルとします。

なお、本ドキュメントをサンプルとしてビルドするためには、各種外部ツールのインストールおよびパスの設定が必要になります。詳細は「[各機能の使用方法](#)」を参照してください。

1.2. このドキュメントの表記



補足事項を表します。



注意事項を表します。

コマンドラインやソースコードを表します。コマンドラインはUnix系OSのシンタックスになります。

1.3. 対象読者

- ・ドキュメント作成に疲弊している開発者
- ・ドキュメント管理に疲弊している管理者
- ・ドキュメント作成および管理のコストを削減したいと考えている経営者

1.4. ライセンス

本環境はMITライセンスのもとで公開されています。

また、本環境に含まれるフォントデータはSIL Open Font License 1.1のもとで公開されています。

2. 提供機能

2.1. 基本機能

[AsciiDoctor PDF](#)が提供する機能が使用可能です。

また、日本語を使用可能とするためのデフォルトのテーマが定義済みですが、デフォルトのテーマのカスタマイズおよびドキュメント毎の個別のテーマの指定が可能です。詳細は「[テーマの使用方法](#)」を参照してください。

なお、日本語の太字や斜体が使用できます。

表 1. 日本語の書式サンプル

書式	日本語例	英語例
標準	これは 標準 の書式です。	This is regular format.
太字	これは 太字 の書式です。	This is bold format.
斜体	これは 斜体 の書式です。	This is <i>italic</i> format.
太字斜体	これは 太字斜体 の書式です。	This is <i>bold and italic</i> format.

既知の問題として、日本語等のマルチバイト文字を使用すると以下の警告が出力されます。



```
warning: regexp match /.../n against to UTF-8 string
```

詳細は[AsciiDoctor PDFのIssues](#)を参照してください。

2.2. 図形描画

[AsciiDoctor Diagram](#) がサポートしている各種ツールを使用した図形描画が可能です。
使用方法是「[図形描画の使用方法](#)」を参照してください。

2.3. PDFテンプレート

ロゴやウォーターマーク等を配置したPDF形式のテンプレートとの合成が可能です。
使用方法是「[PDFテンプレートの使用方法](#)」を参照してください。

2.4. 動的テンプレート

SVG形式のテンプレートの一部の文字列を置換した後に、PDFテンプレートとして合成する事が可能です。
使用方法是「[動的テンプレートの使用方法](#)」を参照してください。

3. 基本的な使用方法

3.1. 導入

下記のコマンドを入力して、本環境のローカルコピーを取得し、取得したディレクトリに移動してください。

```
git clone https://github.com/develhack/jpdoc.git # 本環境のローカルコピーを取得
cd jpdoc # 本環境のローカルコピーに移動
```



以降の説明は全て、本環境のローカルコピーがカレントディレクトリである事が前提です。

本環境の構成は以下となっています。

```
jpdoc/
|   build.gradle ①
|   :
|   |
|   ├── fonts/ ②
|   ├── themes/ ③
|   ├── templates/ ④
|   └── manual/ ⑤
|       |   build.gradle ⑥
|       |
|       └── src/docs/asciidoc/ ⑦
|           |   manual.adoc ⑧
|           |
|           ├── _includes/ ⑨
|           └── _diagrams/ ⑩
```

- ① 本環境を使用するためのビルドスクリプトです。全てのサブプロジェクトに対する設定を行います。
- ② 日本語を出力するためのフォントが格納されています。
- ③ PDFに適用するテーマが格納されています。
- ④ PDFに合成するテンプレートが格納されています。
- ⑤ このドキュメントを作成するためのサブプロジェクトです。本環境の使用例として参照してください。
- ⑥ サブプロジェクト固有の設定を行うビルドスクリプトです。
- ⑦ PDFのソースファイルを配置するディレクトリです。
- ⑧ このドキュメントのソースファイルです。
- ⑨ ドキュメントの断片となる（manual.adocに埋め込まれる）ソースファイルが格納されています。
- ⑩ ドキュメントに埋め込まれる図形のソースファイルが格納されています。



（アンダーバー）で始まるディレクトリ配下のソースファイルはコンパイル対象から除外される設定になっています。

3.2. ドキュメントの作成

ドキュメントを作成する際には、作成するドキュメント用のサブプロジェクトを作成します。

<サブプロジェクト>/src/docs/asciidocディレクトリ配下に作成した、拡張子が.adocのファイルがPDFのソースファイルとして認識されます。

3.3. ドキュメントの設定

<サブプロジェクト>/build.gradleを作成する事で、そのサブプロジェクトにより出力されるPDFの設定を変更する事が出来ます。設定可能な項目は以下の3つです。

表 2. サブプロジェクトの設定項目

設定項目	規定値	説明
theme	'default'	ドキュメントに適用するテーマを指定します。 テーマの使用方法は「 テーマの使用方法 」を参照してください。
template	''	ドキュメントに適用するPDFテンプレートもしくは動的テンプレートを指定します。 下記のtemplateValuesの指定有無によって、PDFテンプレートを使用するか動的テンプレートを使用するかが決まります。 PDFテンプレートの使用方法は「 PDFテンプレートの使用方法 」を、動的テンプレートの使用方法は「 動的テンプレートの使用方法 」をそれぞれ参照してください。
templateValues	[]	動的テンプレートに適用する置換文字列を指定します。 動的テンプレートの使用方法は「 動的テンプレートの使用方法 」を参照してください。

このドキュメントの設定は下記の通りです。

```
asciidoctor {  
    // デフォルトのテーマ ('default') を使用するため theme は未指定  
  
    // 'develhack-template-'の接頭辞を持つ動的テンプレートを使用  
    template = 'develhack'  
    // 動的テンプレートの ${copyright} を置換  
    templateValues = [ 'copyright': 'Copyright © 2016 Develhack K.K. All Rights reserved.' ]  
}
```

3.4. PDFの出力

下記コマンドを実行するとPDFが出力されます。

```
./gradlew asciidoctor # 全てのサブプロジェクトのPDFを出力  
./gradlew example:asciidoctor # exampleサブプロジェクトのPDFを出力
```

PDFは <サブプロジェクト>/build/asciidoc/pdf ディレクトリ配下に出力されます。

PDFテンプレート機能および動的テンプレート機能を使用する場合は、テンプレートと合成されたPDFが <サブプロジェクト>/build/asciidoc/pdf/finalized ディレクトリ配下に出力されます。

4. 各機能の使用法

4.1. テーマの使用法

[AsciiDoctor PDF](#) のテーマを定義する事により、ドキュメント全体のスタイルを指定出来ます。

「[ドキュメントの設定](#)」に示したthemeにfooを指定した場合foo-theme.ymlが下記の順番で検索され、最初に見つかったfoo-theme.ymlが適用されます。foo-theme.ymlが見つからなかった場合にはエラーとなります。

1. <サブプロジェクト>/themes/foo-theme.yml
2. themes/foo-theme.yml

なお、themeの指定を省略した場合はdefaultが指定された場合と同等の動作（default-theme.ymlを検索する）となります。

テーマファイルの記述方法は「[AsciiDoctor PDF](#)」の [AsciiDoctor PDF Theming Guide](#) を参照して下さい。

4.2. 図形描画の使用方法

Asciidoctor Diagram が対応している各種図形を使用可能ですが、PlantUMLのみ動作確認しています。

PlantUMLの機能の一部を使用したUMLのサンプルを以下に示します。

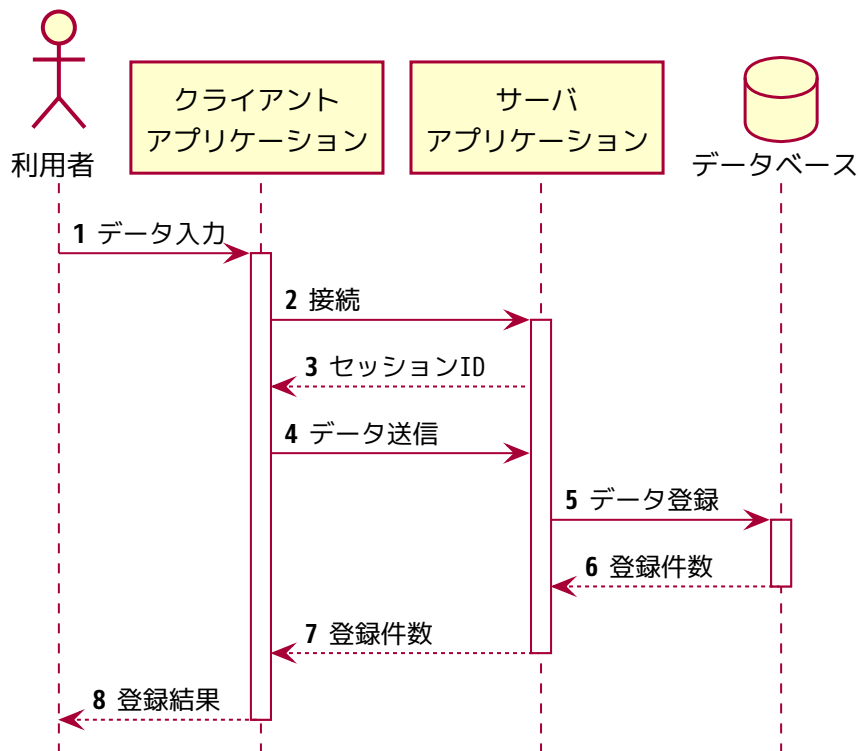


図 1. シーケンス図

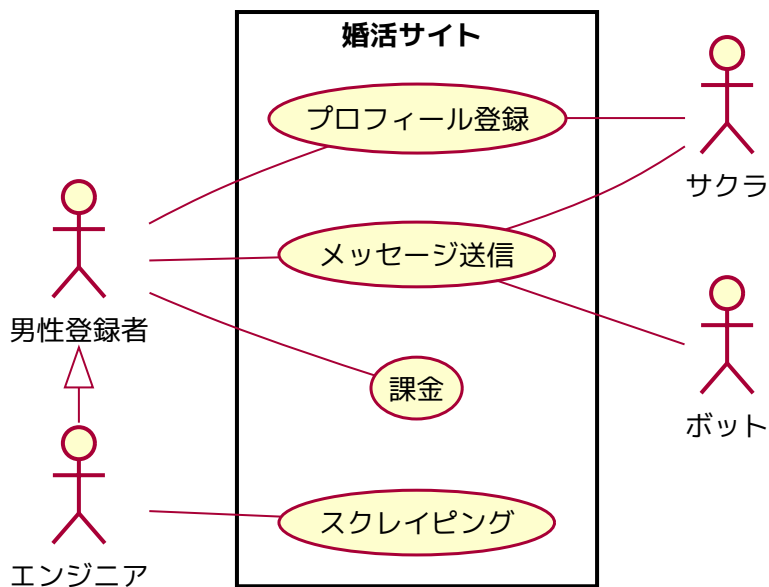


図 2. ユースケース図

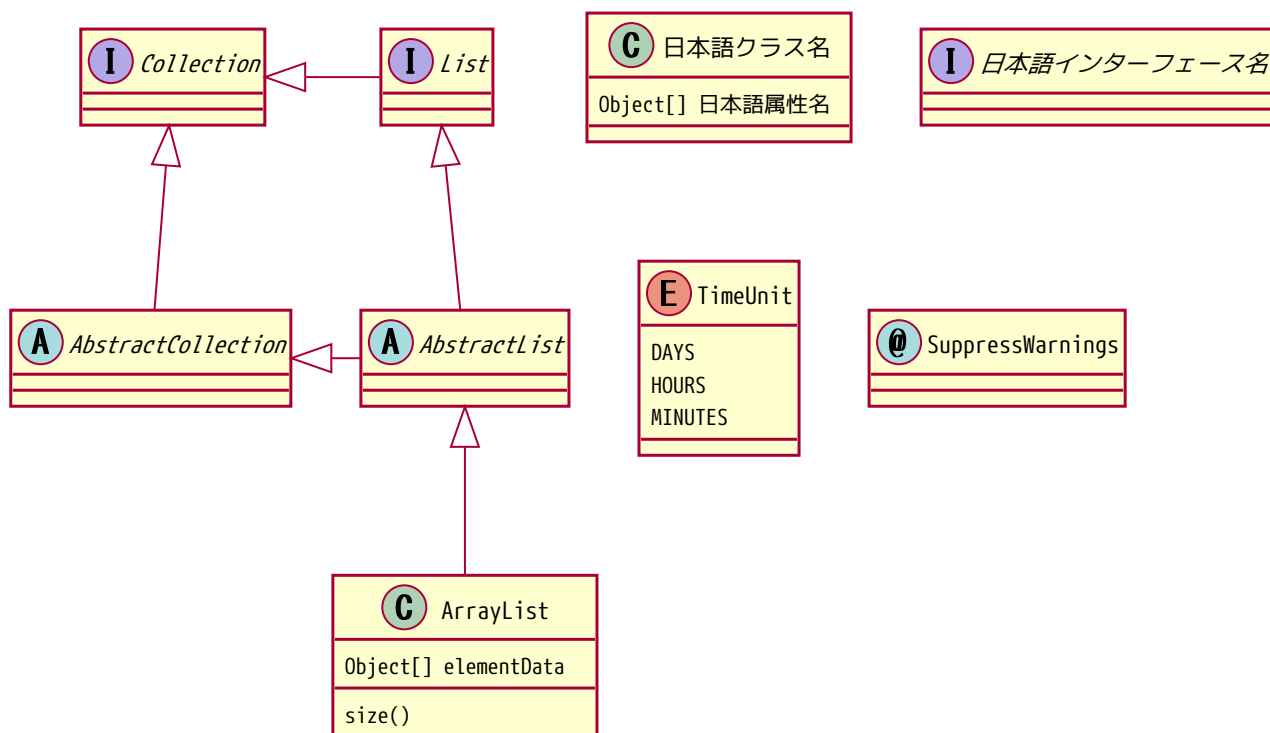


図 3. クラス図

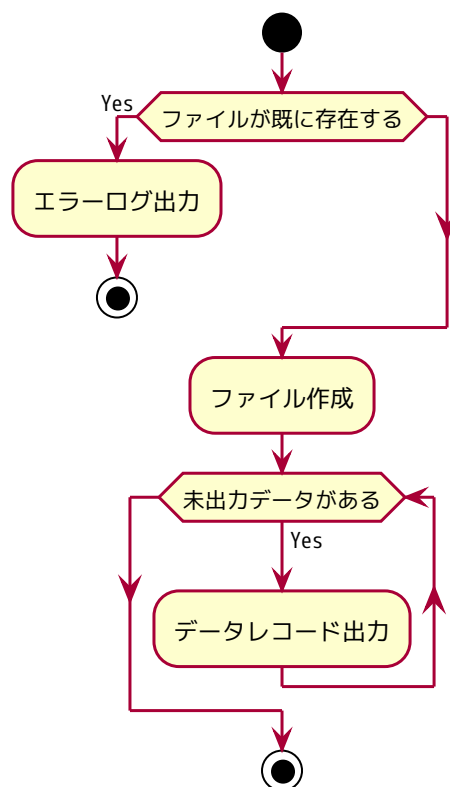


図 4. アクティビティ図



- PlantUMLを使用して図形を描画する場合には[Graphviz](#)のインストールとパスの設定が必要です。
- PlantUML以外を使用して図形を描画する場合には、図形の種類に応じた外部ツールのインストールとパスの設定が必要です。図形毎に必要な外部ツールの詳細は[AsciiDoctor Diagram](#)の公式サイトを参照してください。

4.3. PDFテンプレートの使用方法

テンプレートとなるPDFファイルをドキュメントに合成する事が出来ます。

「[ドキュメントの設定](#)」に示したtemplateにfooを指定した場合foo-template.pdfが下記の順番で検索され、最初に見つかった物が適用されます。

1. <サブプロジェクト>/templates/foo-template.pdf
2. templates/foo-template.pdf

foo-template.pdfが見つからなかった場合にはエラーとなります。

PDFテンプレートが単一ページの場合、ドキュメントの全てのページに同一のテンプレートが合成されます。

PDFテンプレートが複数のページを含む場合、PDFテンプレートの各ページがドキュメントの対応するページに合成され、ドキュメントのページ数がPDFテンプレートのページ数よりも多い場合には、PDFテンプレートの最後のページが残りの全てのページに合成されます。

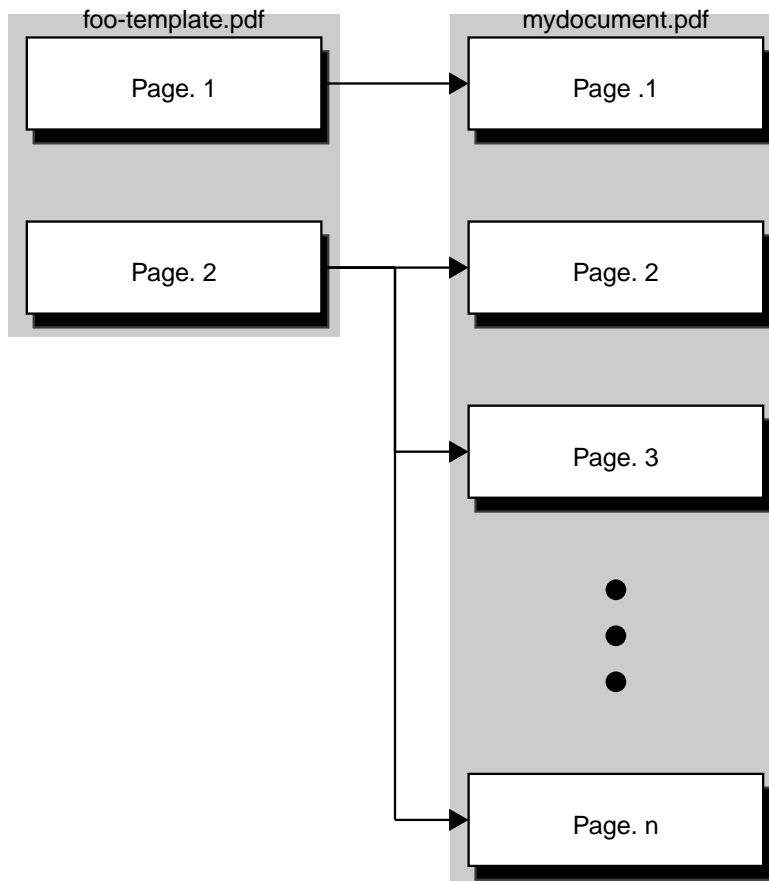


図 5. PDFテンプレートのページとドキュメントのページとの対応



テンプレートとの合成には[PDFtk](#)のインストールとパスの設定が必要です。

4.4. 動的テンプレートの使用方式

テンプレートとなるSVGファイルをドキュメントに合成する事が出来ます。また、合成する際にSVGファイル中の文字列を置換する事が出来ます。

「[ドキュメントの設定](#)」に示したtemplateにfooを指定した場合foo-template-*.svgのパターンにマッチするファイルが下記の順番で検索され、最初に見つかった物が適用されます。

1. <サブプロジェクト>/templates/foo-template-*.svg
2. templates/foo-template-*.svg

パターンにマッチするファイルが見つからなかった場合は「[PDFテンプレートの使用方式](#)」に示した動作となります。

パターンにマッチするファイルが複数見つかった場合はファイル名順に、ドキュメントの各ページに合成されます。

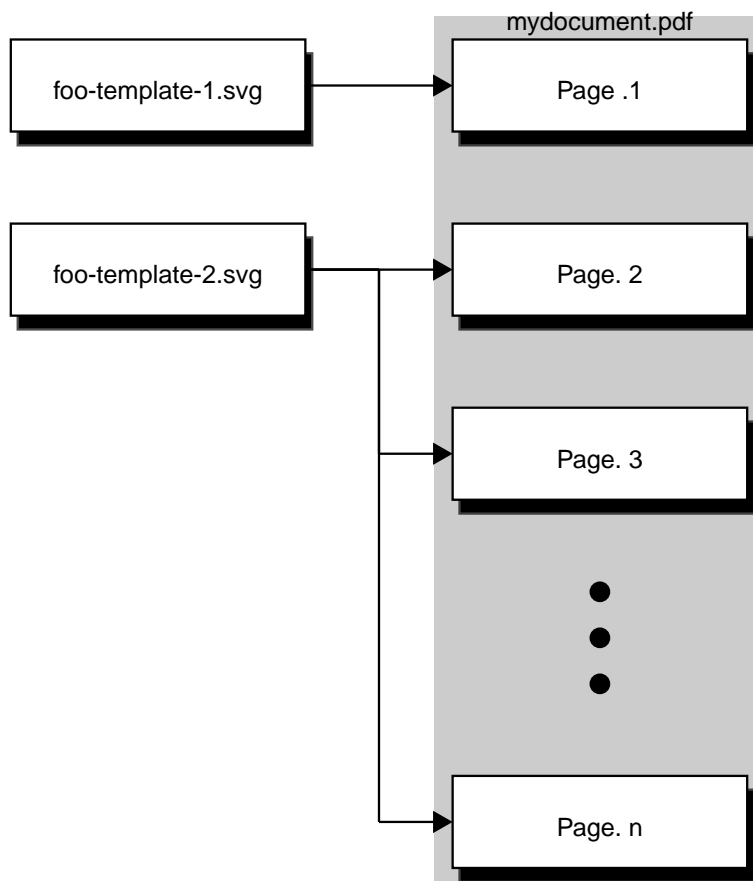


図 6. 動的テンプレートのファイルとドキュメントのページとの対応



動的テンプレートとの合成には[PDFtk](#)および[Inkscape](#)のインストールとパスの設定が必要です。

付録 A: 参考文献

AsciiDoc User Guide (<http://www.methods.co.nz/asciidoc/userguide.html>)

AsciiDocのユーザーガイドです。本環境において全ての記法が使用できる訳ではありませんが、多くの記法が使用可能です。

Gradle User Guide (<https://docs.gradle.org/current/userguide/userguide.html>)

Gradleのユーザーガイドです。このドキュメントではGradleの使用方法については説明していませんので、必要に応じて参照してください。

Asciidoctor Documentation (<http://asciidoctor.org/docs/>)

Asciidoctorの公式ドキュメントです。

Asciidoctor Gradle Plugin (<http://asciidoctor.org/docs/asciidoctor-gradle-plugin/>)

Asciidoctor Gradle Plugin の公式サイトです。

[GitHub] Asciidoctor PDF (<https://github.com/asciidoctor/asciidoctor-pdf>)

Asciidoctor PDF のリポジトリです。

Asciidoctor Diagram (<http://asciidoctor.org/docs/asciidoctor-diagram/>)

Asciidoctor Diagram の公式サイトです。

[GitHub] Asciidoctor (<https://github.com/asciidoctor>)

AsciidoctorのGitHubアカウントです。Asciidoctorに関連するプロダクトのリポジトリが登録されています。

PlantUML (<http://plantuml.com/>)

PlantUMLの公式サイトです。

Graphviz (<http://www.graphviz.org/>)

Graphvizの公式サイトです。

[PDF Labs] PDFtk (<https://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/>)

PDFtkの公式サイトです。

Inkscape (<https://inkscape.org/ja/>)

Inkscapeの公式サイトです。