

JekyllでWebページを作る

-カスタムページを元に Web ページを作り GitHub Pages で公開する-

桐山 和彦*

kiri@TrueFC.org

TrueFC Project

2022年8月20日

概要

コンテンツは(↓)に置いてあります.

<http://www.truefc.org/~kiri/kbug/bof/2022/No.4/cwpj.html>^{注1)}

目次

1	はじめに	1
2	Jekyll とは？	1
3	とりあえず Jekyll チュートリアル	1
3.1	まず Jekyll の実行環境を構築	1
3.2	Jekyll チュートリアルの簡潔バージョンで	2
4	Jekyll について (まとめ)	10
4.1	Jekyll とは	10
4.2	Jekyll のルール	11
5	テーマからカスタムページを作る	13
5.1	まず自分好みのページを見付ける (テーマ選択の条件)	13
5.2	理想的なテーマ	14
5.2.1	ページのカスタマイズ	14
5.2.2	Gem ベーステーマを元にしたカスタムページ	22
5.2.3	GitHub Pages で立ち上げる (GitHub へのデプロイ)	26
5.3	一般的に良くあるテーマ	28
5.3.1	ロケール部分を変数化する	28
5.3.2	Gem ベーステーマの gem を作る	34
5.3.3	Gem ベーステーマの作成から GitHub Pages へのデプロイまで	37
6	最後に	37

*TrueFC Project

注1)<http://www.truefc.org/~kiri/kbug/bof/2022/No.4/cwpj.html>

1 はじめに

- GitHub Pages ではデフォルトで Jekyll (ジキル) によるコード生成 (*.yml ⇒ HTML) を自動で行なってくれるらしい。
- もちろん WEBrick を立ち上げてウェブサーバーを起動してくれる
- さらにコアな部分を別途登録しておけば、それをマスターとして差分情報だけのリポジトリを作れば OK らしい

ということでさっそくやってみる。

2 Jekyll とは？

- Web ページの構造化管理システム
 - ▷ 最小限の労力で静的ウェブページを作る手法
 - ▷ Ruby 化した人にはスンナリ入れます:-)
 - ▷ Liquid なる制御言語を駆使すればコンパクトに作れます
- 実際には既にあるイケてるページから作るのが普通;-p
 - ▷ たとえば [3] や [5] には美しいページが散々あります
 - ▷ それらを自分好みのページにする方法
 - (1) まず git clone してコンテンツを変更
 - (2) 変更部分 (カスタムページ) とそれ以外 (Gem ベーステーマ) に分ける
 - (3) Gem ベーステーマを git push ⇒ Gem ベースリポジトリ
 - (4) カスタムページを git push し Gem ベースリポジトリを参照する ⇒ カスタムリポジトリ

3 とりあえず Jekyll チュートリアル

3.1 まず Jekyll の実行環境を構築

- *BSD 以外でローカルに実行されたい方はこちらをどうぞ
<http://jekyllrb-ja.github.io/docs/installation/>^{注2)}
- *BSD でローカルに実行されたい方
 - (1) Ruby をインストール: lang/ruby30
FreeBSD の ports-current (813d404c7d5c) ではデフォルトは ruby30-3.0.4.2,1 のようです。
 - (2) gem をインストール: devel/ruby-gems
 - (3) bundler をインストール: sysutils/rubygem-bundler
 - (4) rake をインストール: devel/rubygem-rake
 - (5) jekyll をインストール: www/rubygem-jekyll

^{注2)}<http://jekyllrb-ja.github.io/docs/installation/>

- とりあえず”Jekyll VM”を用意しました

▷ ”Jekyll VM”マシン: t.truefc.org

▷ アカウント名: guest[1-5]

パスワードは直前に UsePAM=yes にしますので、その時に御伝えします。なお, sudoers で ”%sudo ALL=(ALL) NOPASSWD: ALL” していますので何でもできます :) 御使用にあたっては自己責任でお願いします。

▷ アプリケーションのインストールは既にある packages または ports からお願いします

パッケージは /var/ports/packages にあります。無かったら /var/ports/jdtpkxb 以下の ports-tree で直接 make install してください。

なお, pkg add -r * などしてバージョンの異なるライブラリ等をインストールして、特定のアプリケーションが動作しなくなったとしても当方は一切の責任を負いません :-p

▷ Jekyll サイトのテストはポート番号を付けて立ち上げてください

Jekyll コンテンツを作成したら、そのディレクトリに移動して、ポート番号を -P オプションで指定して

```
% jekyll serve -Ht.truefc.org -P8080
```

なかんじで立ち上げてください。外からは <http://t.truefc.org:8080/> で見れます。なお、ポート番号は他の人とバッティングしないように話し合ってください :-)

3.2 Jekyll チュートリアル of 簡潔バージョンで

Jekyll 自体のチュートリアルは [7] にあります。ここでは時間がないのでザクっと

(1) とりあえず”Hello World!”

1) index.html を作る

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Home</title>
  </head>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```

2) サイトを構築しウェブサーバー (WEBrick) を立ち上げる

- ローカルなら

```
% jekyll serve
```

- Jekyll VM なら

```
% jekyll serve -Ht.truefc.org -P8080
```

このとき、_site ディレクトリに HTML コンテンツが吐き出されたことを確認

3) できたページを確認

- ローカルなら
http://localhost:4000^{注3)}
- Jekyll VM なら
http://t.truefc.org:8080^{注4)}

(2) "Hello World!" を小文字に

フロントマター (front matter)^{注5)} というのを追加して, liquid^{注6)} という言語の `downcase` フィルターを使う。

```
---
---
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Home</title>
  </head>
  <body>
    <h1>{{ "Hello World!" | downcase }}</h1>
  </body>
</html>
```

- 変数やデータの編集結果の内容を出力するときには `{{ と }}` で囲む⇒オブジェクト

(3) About ページを追加

1) About ページとの切替えのためのナビゲーションを `_includes/navigation.html` に作成

```
<nav>
  <a href="/">Home</a>
  <a href="/about.html">About</a>
</nav>
```

2) 共通のレイアウト `default.html` を `_layouts` ディレクトリに作成

ナビゲーションファイルを liquid の `include` タグでインクルード. 制御はタグと呼ばれる `"{%"` と `"%}"` で括った操作で行なう。

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>{{ page.title }}</title>
  </head>
  <body>
    {% include navigation.html %}
```

^{注3)} <http://localhost:4000.tex>

^{注4)} <http://t.truefc.org:8080>

^{注5)} <http://jekyllrb-ja.github.io/docs/step-by-step/03-front-matter/>

^{注6)} <https://shopify.github.io/liquid/>

```
    {{ content }}
  </body>
</html>
```

- `page.title` :このレイアウトを呼び出したコンテンツのフロントマターで定義される `title` の中身
- `content` :このレイアウトを呼び出したフロントマター以下の内容

3) `index.html` の修正

```
---
layout: default
title: Home
---
<h1>{{ "Hello World!" | downcase }}</h1>
```

- `_layouts/default.html` はフロントマターで `layout: default` とすることで呼び出せる

4) `about.md` を作成

```
---
layout: default
title: About
---
# About page
```

This page tells you a little bit about me.

- `about.md` の内容はマークダウンであることに注意

(4) 現在のページを強調

現在のページの URL を保持している Jekyll 変数^{注7)} を用いて `navigation.html` を変更

```
<nav>
  <a href="/" {% if page.url == "/" %}style="color: red;"{% endif %}>
    Home
  </a>
  <a href="/about.html" {% if page.url == "/about.html" %}style="color: red;"{% endif %}>
    About
  </a>
</nav>
```

(5) 現在のページを緑色で強調

Sass^{注8)} を使います。CSS のソースファイルを `_sass/main.scss` に置き、`assets/css/styles.scss` でそれをインポートします。

ついでにページデータを `_data/navigation.yml` に入れ一元管理。

^{注7)} <http://jekyllrb-jp.github.io/docs/variables/>

^{注8)} <https://sass-lang.com/>

1) navigation.yml

```
- name: Home
  link: /
- name: About
  link: /about.html
```

2) main.scss

```
.current {
  color: green;
}
```

3) styles.scss

```
---
---
@import "main";
```

4) navigation.html

```
<nav>
  {% for item in site.data.navigation %}
    <a href="{{ item.link }}" {% if page.url == item.link %}class="current"{% endif
  {% endfor %}
</nav>
```

5) 最後に _layouts/default.html でスタイルシートを参照して終了

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>{{ page.title }}</title>
    <link rel="stylesheet" href="/assets/css/styles.css">
  </head>
  <body>
    {% include navigation.html %}
    {{ content }}
  </body>
</html>
```

(6) ブログページを追加

個々のブログ(ブログポスト)は `_posts` ディレクトリ下に `YYYY-MM-DD-タイトル.md` という名前で作成し, そのレイアウトを `_layouts/post.html` で指定すると, `_site/YYYY/MM/DD/ タイトル.html` に作られる.

個々のブログポストの URL およびタイトルは `site.posts` の要素の `url`, `title` で参照できる.

1) ブログポストを作成

ここでは, `bananas`, `apples`, `kiwifruit` というタイトルの 3 つを作成

```
---
layout: post
author: jill
---
```

A banana is an edible fruit - botanically a berry - produced by several kinds of large herbaceous flowering plants in the genus *Musa*.

In some countries, bananas used for cooking may be called "plantains", distinguishing them from dessert bananas. The fruit is variable in size, color, and firmness, but is usually elongated and curved, with soft flesh rich in starch covered with a rind, which may be green, yellow, red, purple, or brown when ripe.

```
---
layout: post
author: jill
---
```

An apple is a sweet, edible fruit produced by an apple tree.

Apple trees are cultivated worldwide, and are the most widely grown species in the genus *Malus*. The tree originated in Central Asia, where its wild ancestor, *Malus sieversii*, is still found today. Apples have been grown for thousands of years in Asia and Europe, and were brought to North America by European colonists.

```
---
layout: post
author: ted
---
```

Kiwifruit (often abbreviated as kiwi), or Chinese gooseberry is the edible berry of several species of woody vines in the genus *Actinidia*.

The most common cultivar group of kiwifruit is oval, about the size of a large hen's egg (5-8 cm (2.0-3.1 in) in length and 4.5-5.5 cm (1.8-2.2 in) in diameter). It has a fibrous, dull greenish-brown skin and bright green or golden flesh with rows of tiny, black, edible seeds. The fruit has a soft texture, with a sweet and unique flavor.

2) ポストレイアウト (`_layouts/post.html`) の作成

```
---
layout: default
---
<h1>{{ page.title }}</h1>
<p>{{ page.date | date_to_string }} - {{ page.author }}</p>

{{ content }}
```

- `date_to_string` は日付を良きに図らうフィルター

3) ブログレイアウト (_layouts/blog.html) の作成

```
---
layout: default
title: Blog
---
<h1>Latest Posts</h1>

<ul>
  {% for post in site.posts %}
    <li>
      <h2><a href="{ post.url }">{ post.title }</a></h2>
      { post.excerpt }
    </li>
  {% endfor %}
</ul>
```

- `post.excerpt` は最初の一段落

4) ナビゲーションにブログを追加

```
- name: Home
  link: /
- name: About
  link: /about.html
- name: Blog
  link: /blog.html
```

(7) スタッフページを追加

ブログの著者と、そのブログの一覧を表示する。

まず、ファイルをまとめて入れるディレクトリ名(コレクション名)を `collections` キーで `_config.yml` で定義。それぞれの著者別の説明ファイルを `jekyll_root_dir/_コレクション名` に入れ、後はブログと同じ。ここでは、コレクション名は `authors`

1) コレクション名 (`authors`) を定義

```
collections:
  authors:
    output: true
```

- `output: true` しないと出力されないらしい

2) 各著者のドキュメント (`_authors/*.md`) を作成

ここでは `jill` と `ted`

```
---
short_name: jill
name: Jill Smith
position: Chief Editor
---
Jill is an avid fruit grower based in the south of France.
```

```

---
short_name: ted
name: Ted Doe
position: Writer
---
Ted has been eating fruit since he was baby.

```

- 3) 著者のレイアウト (_layouts/author.html) を作成
 ついでに著者の作成したブログポストを表示

```

---
layout: default
---
<h1>{{ page.name }}</h1>
<h2>{{ page.position }}</h2>

{{ content }}

<h2>Posts</h2>
<ul>
  {% assign filtered_posts = site.posts | where: 'author', page.short_name %}
  {% for post in filtered_posts %}
    <li><a href="{{ post.url }}">{{ post.title }}</a></li>
  {% endfor %}
</ul>

```

- content はマークダウンであるが、ここでは自動で Markdown ⇒ HTML 変換される
- where: A, B は「A == B である配列のみを抽出」

- 4) スタッフページ (staff.html) の作成

```

---
layout: default
title: Staff
---
<h1>Staff</h1>

<ul>
  {% for author in site.authors %}
    <li>
      <h2><a href="{{ author.url }}">{{ author.name }}</a></h2>
      <h3>{{ author.position }}</h3>
      <p>{{ author.content | markdownify }}</p>
    </li>
  {% endfor %}
</ul>

```

- author.content は content じゃないので markdownify によって Markdown ⇒ HTML 変換が必要

5) ブログページ (_layouts/post.html) に著者のページへのリンクを追加

```
---
layout: default
---
<h1>{{ page.title }}</h1>

<p>
  {{ page.date | date_to_string }}
  {% assign author = site.authors | where: 'short_name', page.author | first %}
  {% if author %}
    - <a href="{{ author.url }}">{{ author.name }}</a>
  {% endif %}
</p>

{{ content }}
```

6) ナビゲーション (_data/navigation.yml) にスタッフを追加

```
- name: Home
  link: /
- name: About
  link: /about.html
- name: Blog
  link: /blog.html
- name: Staff
  link: /staff.html
```

7) ドキュメントファイル (*.md) のデフォルトレイアウトを_config.yml で設定

```
collections:
  authors:
    output: true

defaults:
  - scope:
      path: ""
      type: "authors"
    values:
      layout: "author"
  - scope:
      path: ""
      type: "posts"
    values:
      layout: "post"
  - scope:
      path: ""
    values:
      layout: "default"
```

4 Jekyll について (まとめ)

4.1 Jekyll とは

(1) 要は Web ページの構造化管理システム

【チュートリアル の例】

```
/jekyll_root_dir +- Gemfile
                  +- _config.yml ← 全体のページ構成の設定
                  +- index.html ← メインページの記述
                  +- about.md ← About ページの記述
                  +- blog.html ← ブログページの記述
                  +- staff.html ← スタッフページの記述
                  +- _layouts ← ページレイアウト格納ディレクトリ
                    | +- default.html ← デフォルトページのレイアウト
                    | +- post.html ← ブログページのレイアウト
                    | +- author.html ← スタッフページのレイアウト
                  +- _includes ← インクルードファイル格納ディレクトリ
                    | +- navigation.html ← 各ページの切り替えの指示
                  +- _data ← データファイル格納ディレクトリ
                    | +- navigation.yml ← navigation.html で用いるデータ (YAML 形
式)
                  +- _posts ← ブログ形式ファイル格納ディレクトリ
                    | +- 2018-08-20-bananas.md ← 2018 年 8 月 20 日の bananas フ
ファイル
                    |
                    | +- 2018-08-21-apples.md ← 2018 年 8 月 21 日の apples フ
ファイル
                    |
                    | +- 2018-08-22-kiwifruit.md ← 2018 年 8 月 22 日の kiwifruit フ
ファイル
                    |
                  +- _authors ← _config の authors コレクションに対応するディ
レクトリ
                    | +- jill.md ← コレクションアイテム
                    | +- ted.md ← コレクションアイテム
                  +- assets ← 部品格納ディレクトリ
                    | +- css ← CSS ディレクトリ
                    | +- styles.scss ← 全体のスタイルシート用 sass
                  +- _sass ← Sass による SCSS 格納ディレクトリ
                    | +- main.scss ← assets/css/styles.scss により生成され
た SCSS
                  +- _site ← Jekyll により生成されたファイル格納ディレクトリ
                    +- index.html ← メインページ
                    +- about.html ← About ページ
                    +- blog.html ← ブログページ
                    +- staff.html ← スタッフページ
                    +- feed.xml
                    +- sitemap.xml
                    +- robots.txt
```

```

+- 2018
|   +- 08
|   |   +- 20
|   |   |   +- bananas.html ← bananas ページ
|   |   +- 21
|   |   |   +- apples.html ← apple ページ
|   |   +- 22
|   |       +- kiwifruit.html ← kiwifruit ページ
+- assets ← 部品格納ディレクトリ
|   +- css
|       +- styles.css
|       +- styles.css.map
+- authors
    +- jill.html
    +- ted.html

```

(2) 基本的な構成

- まず、全ての処理が実行された結果は`_site`に出力される
- レイアウトファイルは`_layouts`に入れる
- インクルードファイルは`_includes`に入れる
- ブログファイルは`_posts`に入れる
- ロードするデータ (YAML, JSON, CSV 等) は`_data`に入れる
- 必要な部品 (CSS, JS, 画像等) は`assets`に入れる

(3) 基本的な作り方

- 1) サイトの構成を決める
- 2) 各ページを作成する
 - ブログページは所定の様式で`_posts`に作成
 - ブログのような複数ファイルに亘るページはコレクションで作成
 - その他はベタでマークダウンで作成
- 3) 共通のレイアウトは`_layouts`に入れる
- 4) 最後にデフォルトのレイアウトをまとめて`_config.yml`に記述

4.2 Jekyll のルール

(1) 基本的なルール

- 全ての jekyll ファイル^{注9)}は「フロントマター (front matter) + 内容 (content)」で構成される

【index.html の例】

^{注9)}ここで「jekyll ファイル」とは、特定の仕様を除く全てのファイルを意味し、HTML, Markdown, Sass ファイルを含む。

```

---
layout: default  └→ フロントマター
title: Home      ┘
---
<h1>{{ "Hello World!" | downcase }}</h1> → 内容

```

- ▷ フロントマターは”—”で区切られた部分
 - ここには「変数名:値」という形で変数に値を設定できる^{注10)}. なお フロントマターは必須ではないが、これがないと Jekyll はスルーします ;-)

- 全てのページのデータはオブジェクトになる
 - ▷ 当該ページの内容⇒ content
 - ▷ 当該ページのフロントマターの変数⇒ page. 変数名
 - ▷ ブログサイトおよびコレクションで作成したサイトはそれぞれ固有名で指定
- ページ内容は Liquid 言語 [6] で制御できる
 - ▷ オブジェクトの出力⇒ {{ オブジェクト名 }}
 - ▷ コンテンツの制御 (タグ) ⇒ {%制御文%}…{%終了文%}
 - {% assign foo = "bar" %}
 - {% if…%}…{% endif %}
 - {% for foo in bar %}…{% endfor %}
 - ▷ オブジェクトの編集 (フィルター) ⇒ {{ foo | フィルター名 […] }}

(2) ブログサイト (_posts) のルール

- _posts に YYYY-MM-DD-タイトル名.md という名前のファイル名のファイルにマークダウン形式で書く
- 各ファイル名は変換されて site.posts 配列に格納 (ここではその要素を post とする)
 - ▷ そのファイルに対応するページの URL ⇒ post.url (YYYY/MM/DD/タイトル名.html)
 - ▷ そのファイルに対応するページのタイトル名 ⇒ post.title (タイトル名)
 - ▷ そのファイルの内容の最初の段落 ⇒ post.excerpt

(3) データサイト (_data) のルール

- YAML の入れ子マッピング形式で書くとマッピングの配列になる
 - 【_data/navigation.yml の例】

```

- name: Home
  link: /
- name: About
  link: /about.html
- name: Blog
  link: /blog.html
- name: Staff
  link: /staff.html

```

^{注10)}YAML として見ると、ドキュメント内容にマッピングを指定した形と言える。ただし、YAML では”—”以下がドキュメント内容になるが、2 番目のドキュメントは YAML のドキュメントではなくでは HTML コンテンツとなる。YAML については詳しくは [10] を参照。

各シーケンスを要素(ここでは item とする)として site.data.navigation 配列で参照できる。各要素の変数は, item.name および item.link で参照できる。

(4) コレクションサイトのルール

- コレクションサイトの作成手順

- 1) コレクションを定義する (.config.yml)

```
collections:  
  コレクション名:  
    output: true
```

collections キー以下はインデントして, それぞれのコレクション名のスコープを考慮しないとイケない。デフォルトではコレクションはドキュメントページを出力しないようなので, 必ず最後に "output: true" を付けておくこと。

- 2) コレクションサイトを作成する

"_コレクション名" のディレクトリを掘る。この中にデータファイルを入れる。

- 3) データを作成する

コレクションサイトに, "ファイル名.md" のデータファイルを作成する。

▷ データファイルの中身は"site. コレクション名"配列に格納

(5) 各ファイルのデフォルト設定を一括管理

_config.yml の中の defaults キーで設定できる

- 設定方法

```
defaults:  
  - scope:  
    path: "対象とするファイルのパス"  
    type: "対象とするファイルのタイプ"  
  values:  
    layout: "適用するレイアウト"
```

▷ path: "" の場合は全てのファイルが対象になる

▷ type: Jekyll の定義するタイプで, posts およびコレクションで定義されたコレクション名

5 テーマからカスタムページを作る

5.1 まず自分好みのページを見付ける (テーマ選択の条件)

[3] や [5] に散々のページがありますので, 自分の目的に合ったイケてるページを探しましょう。参照するテーマ (Gem ベーステーマ) の作り方および, それを参照してカスタムページを作る方法は [4] を参考にしました。

なお, ページの選定に当たっては, 以下の点に注意されたし。

- (1) プロプライエタリでないこと

できれば GPL なやつも避ける方が良い :-)

(2) できるだけ多くのバリエーションのあるページを作っているもの

作成の幅ができます

(3) 既に Gem パッケージ化されているもの

Gem ベーステーマを作る必要がなくなるので、これにこしたことはないです。ルートに *.gemspec ファイルがあれば OK

(4) ロケールを設定できるようになっているもの

メニューやコンテンツの内容を一つのファイルに言語別に切り分けて変数で切替えられるようになっているものがベストです。_data/*.yml あたりを調べてみれば良いかも :-)

5.2 理想的なテーマ

テーマ選択の条件 5.1 節のほぼ全てを満たしているテーマとして raviriley/agency-jekyll-theme [2] があります。デモページは [1] です。

5.2.1 ページのカスタマイズ

(1) ベースとするテーマを展開

```
kiri@smtp:~/projects/jekyll[1017]% git clone https://github.com/raviriley/agency-jekyll-theme
Cloning into 'agency-jekyll-theme'...
remote: Enumerating objects: 1213, done.
remote: Counting objects: 100% (25/25), done.
remote: Compressing objects: 100% (22/22), done.
remote: Total 1213 (delta 5), reused 22 (delta 3), pack-reused 1188
Receiving objects: 100% (1213/1213), 7.43 MiB | 11.65 MiB/s, done.
Resolving deltas: 100% (620/620), done.
kiri@smtp:~/projects/jekyll[1018]%
```

(2) とりあえず Gem ファイルを作成

1) スペック名を jekyll-theme-agency に変更

```
kiri@smtp:~/projects/jekyll/agency-jekyll-theme.org[1075]% diff -u jekyll-agency.gemspec
--- jekyll-agency.gemspec~      2022-07-29 10:08:24.422123000 +0900
+++ jekyll-agency.gemspec      2022-08-01 15:37:05.588142000 +0900
@@ -1,5 +1,5 @@
  Gem::Specification.new do |spec|
-   spec.name      = "jekyll-agency"
+   spec.name      = "jekyll-theme-agency"
    spec.version   = "1.1.0"

    spec.authors  = ["Ravi Riley"]
kiri@smtp:~/projects/jekyll/agency-jekyll-theme.org[1076]%
```

2) Gem ファイルを作成

```

kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1060]% gem build jekyll-agency.gemspec
Successfully built RubyGem
Name: jekyll-agency
Version: 1.1.0
File: jekyll-agency-1.1.0.gem
kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1061]% ll
total 3311
drwxr-xr-x 10 kiri admin    22  7月 29 10:10 .
drwxr-xr-x 12 kiri admin    12  7月 29 10:08 ..
drwxr-xr-x  8 kiri admin    13  7月 29 10:08 .git
-rw-r--r--  1 kiri admin    66  7月 29 10:08 .gitattributes
drwxr-xr-x  3 kiri admin     6  7月 29 10:08 .github
-rw-r--r--  1 kiri admin    90  7月 29 10:08 .gitignore
-rw-r--r--  1 kiri admin   417  7月 29 10:08 404.html
-rw-r--r--  1 kiri admin   247  7月 29 10:08 Gemfile
-rw-r--r--  1 kiri admin  1077  7月 29 10:08 LICENSE.txt
-rw-r--r--  1 kiri admin  6716  7月 29 10:08 README.md
-rw-r--r--  1 kiri admin  1469  7月 29 10:08 _config.yml
drwxr-xr-x  2 kiri admin     5  7月 29 10:08 _data
drwxr-xr-x  2 kiri admin    14  7月 29 10:08 _includes
drwxr-xr-x  2 kiri admin     5  7月 29 10:08 _layouts
drwxr-xr-x  2 kiri admin     5  7月 29 10:08 _portfolio
drwxr-xr-x  5 kiri admin     5  7月 29 10:08 _sass
drwxr-xr-x  5 kiri admin     5  7月 29 10:08 assets
-rw-r--r--  1 kiri admin    21  7月 29 10:08 index.md
-rw-r--r--  1 kiri admin 2074624  7月 29 10:10 jekyll-themr-agency-1.1.0.gem
-rw-r--r--  1 kiri admin   841  7月 29 10:08 jekyll-agency.gemspec
-rw-r--r--  1 kiri admin  1485  7月 29 10:08 legal.md
-rw-r--r--  1 kiri admin 1161443  7月 29 10:08 screenshot.PNG
kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1062]%

```

(3) ロケールの設定

1) メニューの設定

_data/navigation.yml に ja を追加

```

diff --git a/_data/navigation.yml b/_data/navigation.yml
index e0d1561..ff9aa52 100644
--- a/_data/navigation.yml
+++ b/_data/navigation.yml
@@ -75,3 +75,22 @@ de: &DEFAULT_DE
     section: Kontakt
     - title: "Quellcode des Themas"
       url: https://github.com/raviriley/agency-jekyll-theme/
+
+# Japanese
+# -----
+ja: &DEFAULT_JA

```



```

+     role: "□□□□□□□□"
+     image: assets/img/team/500x500.jpg
+     social:
+       - url: https://twitter.com
+         icon: fab fa-twitter
+       - url: https://facebook.com
+         icon: fab fa-facebook-f
+       - url: https://linkedin.com
+         icon: fab fa-linkedin-in
+   - name: "□□□□□□□□"
+     role: "□□□□□□□□"
+     image: assets/img/team/500x500.jpg
+     social:
+       - url: https://twitter.com
+         icon: fab fa-twitter
+       - url: https://facebook.com
+         icon: fab fa-facebook-f
+       - url: https://linkedin.com
+         icon: fab fa-linkedin-in
+
+   contact:
+     title: "私達への連絡"
+     text: "OpenAGR への緊急のお問い合わせは 123456789 まで"
+     section: contact
+     name: "お名前 (必須) "
+     name-validation: "お名前が記入されていません"
+     email: "メールアドレス (必須) "
+     email-validation: "メールアドレスが記入されていません"
+     phone: "お電話番号"
+     phone-validation: "お電話番号が記入されていません"
+     message: "ご連絡内容 (必須) "
+     message-validation: "メッセージが記入されていません"
+     subject: "お問い合わせフォームの送信"
+     submit: "メッセージを送る"
+
+   footer:
+     legal: "プライバシーポリシー"
+     social:
+       - url: https://twitter.com
+         icon: "fab fa-twitter"
+       - url: https://facebook.com
+         icon: "fab fa-facebook-f"
+       - url: https://linkedin.com
+         icon: "fab fa-linkedin-in"
+       - url: https://github.com/raviriley/agency-jekyll-theme
+         icon: "fab fa-github"

```

```
+      - url: https://instagram.com
+      icon: "fab fa-instagram"
```

3) ロケールを ja に切り替え

_config.yml の locale キーを ja にする

```
diff --git a/_config.yml b/_config.yml
index 3cfc332..85a54e2 100644
--- a/_config.yml
+++ b/_config.yml
@@ -16,7 +16,7 @@ theme: jekyll-agency
  url      : "" # the base hostname & protocol for your site, e.g. http://example.com
  baseurl: "" # the subpath of your site, e.g. /blog

-title      : Your Awesome Website
+title      : OpenAGR
  email     : your-email@example.com #this is also the email contact forms will go to
  description: "Site description"
  author    : Your Name
@@ -24,7 +24,7 @@ author      : Your Name
  # path: assets\img\clients\creative-market.jpg
  # height: 60 #height in px, defaults to 52px

-locale: "en-US" # See available languages in _data/sitetext.yml
+locale: "ja" # See available languages in _data/sitetext.yml

analytics:
  google: #Google Analytics tracking code here
```

4) 最後にメインページの背景画像を変更

_data/style.yml の header-image キーで切替えられます

```
diff --git a/_data/style.yml b/_data/style.yml
index 376339d..37c1b8a 100644
--- a/_data/style.yml
+++ b/_data/style.yml
@@ -4,7 +4,7 @@
  #black: "#000"

  #Image Customization
-header-image: "../img/header.png"
+header-image: "../img/openagr.png"
  contact-image: "../img/contact.png"
  #Font Awesome icon for portfolio grid when an item is hovered over
  #portfolio-icon: fas fa-expand-arrows-alt fa-3x #default: fas fa-plus fa-3x
```

(4) ページのテスト

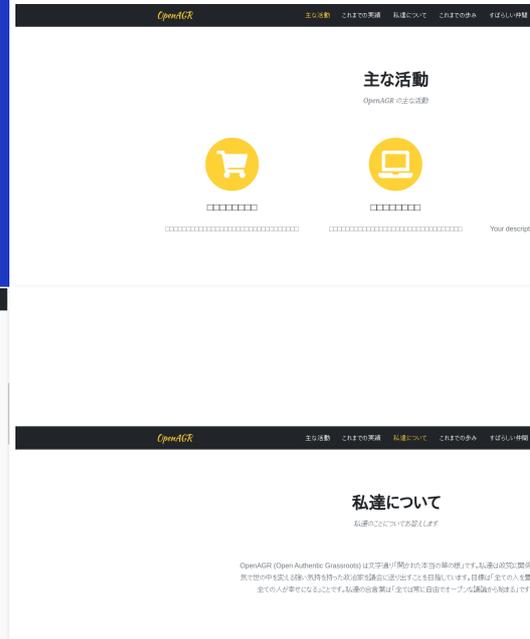
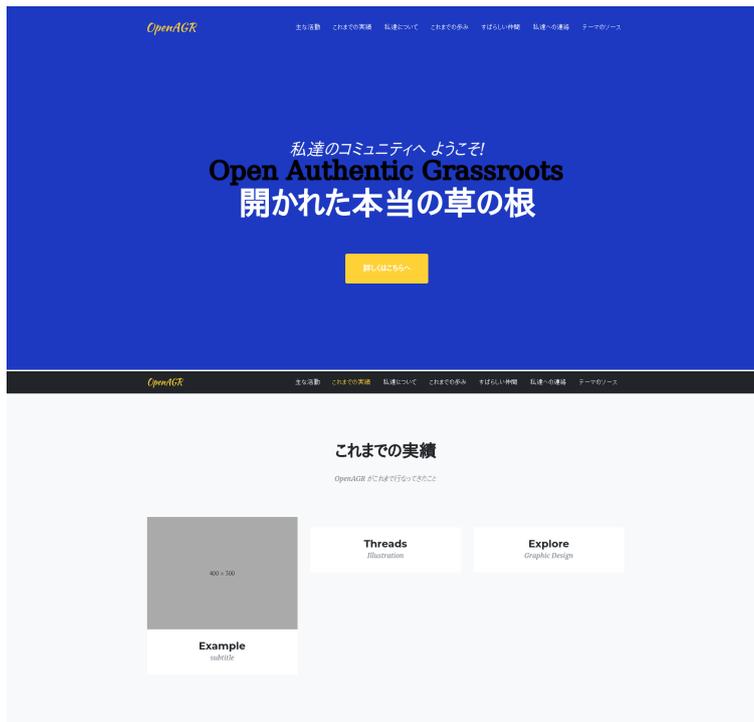
1) テストサイトにコピー

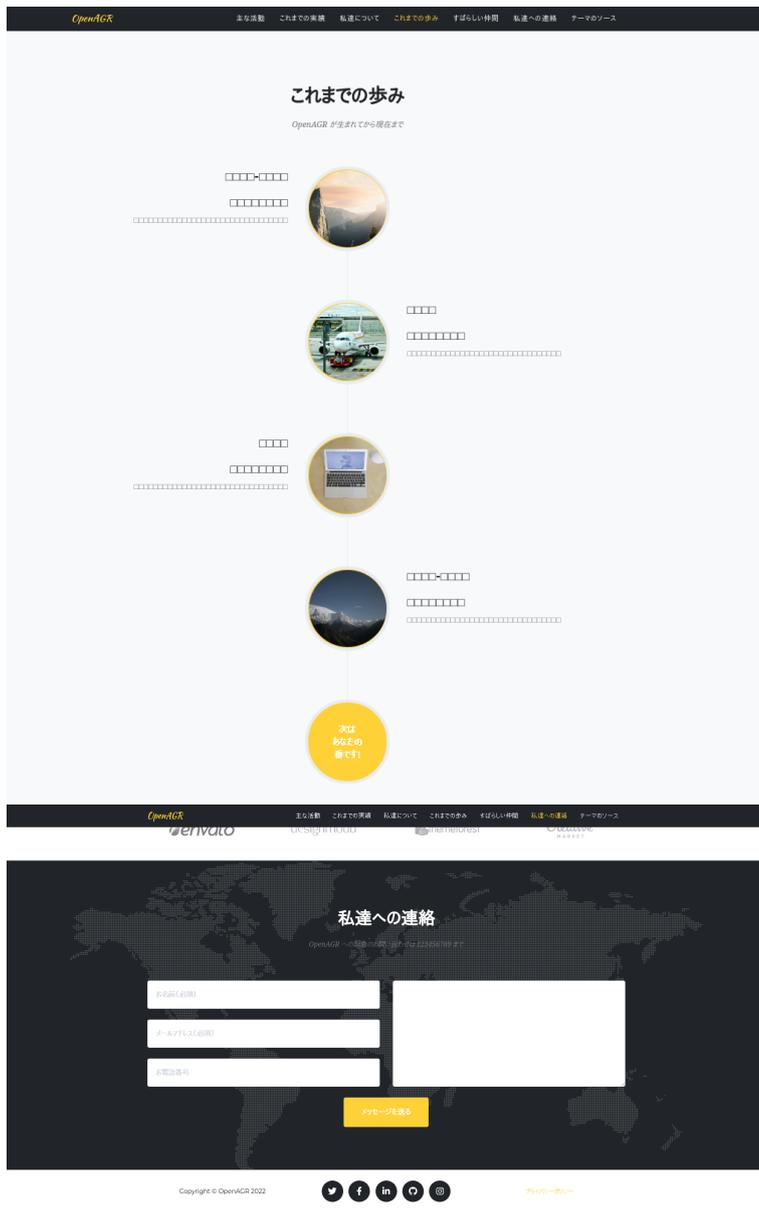
```
kiri@smtp:~/projects/jekyll[1067]% scp -r agency-jekyll-theme guest@t:~/
```

2) テストサイトを立ち上げる

```
guest@t:~/agency-jekyll-theme % bundle exec jekyll serve -Ht.truefc.org -P8080
Configuration file: /home/guest/agency-jekyll-theme/_config.yml
Theme Config file: /home/guest/agency-jekyll-theme/_config.yml
      Source: /home/guest/agency-jekyll-theme
      Destination: /home/guest/agency-jekyll-theme/_site
Incremental build: disabled. Enable with --incremental
Generating...
      done in 0.188 seconds.

Please add the following to your Gemfile to avoid polling for changes:
require 'rbconfig'
if RbConfig::CONFIG['target_os'] =~ /(?!i-mx:bsd|dragonfly)/
  gem 'rb-kqueue', '>= 0.2'
end
Auto-regeneration: enabled for '/home/guest/agency-jekyll-theme'
Server address: http://t.truefc.org:8080/
Server running... press ctrl-c to stop.
[2022-08-01 09:43:20] ERROR '/favicon.ico' not found.
%
```





5.2.2 Gem ベーステーマを元にしたカスタムページ

(1) Gem ベーステーマを作る

Gem パッケージから Gem ベーステーマの FreeBSD パッケージを作る

1) Gem パッケージを DISTFILES へコピー

```
# cp ~/guest/agency-jekyll-theme/jekyll-theme-agency-1.1.0.gem /var/ports/distfiles/
#
```

2) ベースとなるスケルトン (textproc/rubygem-minima) をコピー

```
root@t:/home/guest # cd /var/ports/jdtpkxb/textproc/
root@t:/var/ports/jdtpkxb/textproc # cpdup /usr/ports/textproc/rubygem-minima rubygem-
```

3) rubygem-jekyll-theme-agency 用に変更

- Makefile の変更点は以下の 4 つ

- ▷ PORTNAME ⇒ jekyll-theme-agency
- ▷ PORTVERSION ⇒ 1.1.0 (*.gemspec のバージョンに合わせる)
- ▷ COMMENT ⇒ "Bootstrap Agency ported to Jekyll" (*.gemspec のサマリーから拝借)
- ▷ LICENSE ⇒ MIT (*.gemspec のライセンスに合わせる)

- distinfo を作り直す

```
root@t:/var/ports/jdtpkxb/textproc/rubygem-jekyll-theme-agency # rm distinfo
root@t:/var/ports/jdtpkxb/textproc/rubygem-jekyll-theme-agency # make makesum
```

4) Gem パッケージを作成&インストール

```
# make package install
#
```

- Gem パッケージは /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/* にインストールされる

```
root@t:/var/ports/jdtpkxb/textproc/rubygem-jekyll-theme-agency # pkg info -l rub
rubygem-jekyll-theme-agency-1.1.0:
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/404.html
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/LICENSE.txt
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/README.md
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_data/naviga
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_data/sitete
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_data/style.
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/ab
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/cl
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/co
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/fo
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/he
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/mo
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/na
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/na
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/po
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/se
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/te
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_includes/ti
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_layouts/def
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_layouts/hom
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_layouts/pag
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/base/_
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/base/_
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/base/_
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/compon
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/compon
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/compon
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/layout
  /usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/_sass/layout
```



```

/usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/index.md
/usr/local/lib/ruby/gems/2.7/gems/jekyll-theme-agency-1.1.0/legal.md
/usr/local/lib/ruby/gems/2.7/specifications/jekyll-theme-agency-1.1.0.gemspec
/usr/local/share/licenses/rubygem-jekyll-theme-agency-1.1.0/LICENSE
/usr/local/share/licenses/rubygem-jekyll-theme-agency-1.1.0/MIT
/usr/local/share/licenses/rubygem-jekyll-theme-agency-1.1.0/catalog.mk
root@t:/var/ports/jdtpkxb/textproc/rubygem-jekyll-theme-agency #

```

(2) カスタムページを作る

1) 新たにカスタムディレクトリを掘る

```

kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1078]% mkdir -p ../custompages/openagr
kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1079]%

```

2) 変更部分だけをカスタムディレクトリにコピー

- *index.{html,md} は変更の有無にかかわらず必ずコピー
- その他は基本的に gem パッケージに無いものは全てコピーする

```

kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1079]% cp _index.md _config.yml
kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1079]% cp -R _data _portfolio ..
kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1079]% mkdir -p ../custompages/openagr
kiri@smtp:~/projects/jekyll/agency-jekyll-theme[1079]% cp assets/img/openagr.png

```

3) Gem ベーステーマを参照するように設定

- _config.yml の theme キーに Gem ベーステーマ名をセット

```

kiri@smtp:~/projects/jekyll/custompages/openagr.org[1080]% diff -u ../../agency-jekyll-theme/_config.yml 2022-07-28 09:35:10.049498000 +0900
--- ../../agency-jekyll-theme/_config.yml
+++ _config.yml 2022-08-01 16:13:57.395972000 +0900
@@ -11,7 +11,7 @@
 # You can create any custom variable you would like, and they will be accessible in the templates via {{ site.myvariable }}.

```

```

-theme: jekyll-agency
+theme: jekyll-theme-agency

```

```

url      : "" # the base hostname & protocol for your site, e.g. http://example.com
baseurl: "" # the subpath of your site, e.g. /blog
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1081]%

```

- Gemfile でテーマ gem をセット

```

kiri@smtp:~/projects/jekyll/custompages/openagr.org[1081]% diff -u ../../agency-jekyll-theme/Gemfile 2022-07-27 16:31:59.734716000 +0900
--- ../../agency-jekyll-theme/Gemfile
+++ Gemfile 2022-08-01 16:20:33.592458000 +0900
@@ -1,7 +1 @@
-#after any changes to the Gemfile, execute bundle update!
-source "https://rubygems.org"
-gemspec
-

```

```

-# Delete the following lines if not on Windows:
-# Performance-booster for watching directories on Windows
-gem "wdm", ">= 0.1.0" if Gem.win_platform?
+gem "jekyll-theme-agency"
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1082]%

```

- 4) テストサイトにカスタムページをコピー

```
kiri@smtp:~/projects/jekyll/custompages[1083]% scp -r openagr.org guest@t:~/
```

- 5) カスタムサイトを立ち上げる

```

guest@t:~/openagr.org % cd $HOME/openagr.org
guest@t:~/openagr.org % bundle exec jekyll serve -Ht.truefc.org -P8080
Configuration file: /home/guest/openagr.org/_config.yml
      Source: /home/guest/openagr.org
      Destination: /home/guest/openagr.org/_site
Incremental build: disabled. Enable with --incremental
      Generating...
                done in 0.256 seconds.
Please add the following to your Gemfile to avoid polling for changes:
      require 'rbconfig'
      if RbConfig::CONFIG['target_os'] =~ /(?!i-mx:bsd|dragonfly)/
        gem 'rb-kqueue', '>= 0.2'
      end
Auto-regeneration: enabled for '/home/guest/openagr.org'
      Server address: http://t.truefc.org:8080/
      Server running... press ctrl-c to stop.
%

```

5.2.3 GitHub Pages で立ち上げる (GitHub へのデプロイ)

- (1) Gem ベースリポジトリを作る

- 1) オリジナルの Gem パッケージを展開

```

% cd /jekyll_root_dir
kiri@smtp:~/projects/jekyll[1133]% gem unpack agency-jekyll-theme/jekyll-theme-agency-1.1.0
/usr/local/lib/ruby/site_ruby/2.7/rubygems/package.rb:509: warning: Using the last
Unpacked gem: '/home/kiri/projects/jekyll/jekyll-theme-agency-1.1.0'
kiri@smtp:~/projects/jekyll[1134]%

```

- 2) .gitignore でコミットしないファイルを設定

```

% cd jekyll-theme-agency-1.1.0
% cat .gitignore
*.gem
*~
.bundle
.jekyll-cache

```

```
.sass-cache
_site
Gemfile.lock
%
```

3) GitHub 上でリポジトリを作成

- README.md も含めてコンテンツは何も作らなくて良い
- ここでは kmsq2c/jekyll-theme-agency

4) コミットし GitHub へプッシュ

```
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1085]% git init
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1089]% git add -A
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1090]% git commit -m "first commit"
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1091]% git branch -M main
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1092]% git remote add origin https://github.com:kmsq2c/jekyll-theme-agency
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1093]% git push -u origin main
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1094]%
```

(2) カスタムリポジトリを作る

1) README.md を書き換える

- 必ず README.md は、そのサイト用に書き換えておきましょう
- いずれにせよ README.md が無いとコミットできません

2) Gem ベーステーマのあるサイトをリモートテーマに設定

- remote_theme に Gem ベーステーマのリポジトリパスを設定

```
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1104]% diff -u _config.yml~
--- _config.yml~      2022-08-01 16:13:57.395972000 +0900
+++ _config.yml 2022-08-04 09:51:24.662300000 +0900
@@ -11,7 +11,8 @@
 # You can create any custom variable you would like, and they will be accessible
 # in the templates via {{ site.myvariable }}.
```

```
-theme: jekyll-theme-agency
+#theme: jekyll-theme-agency
+remote_theme: raviriley/agency-jekyll-theme
```

```
url      : "" # the base hostname & protocol for your site, e.g. http://example.com
baseurl: "" # the subpath of your site, e.g. /blog
```

```
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1105]%
```

- .gitignore でコミットしないファイルを設定

```
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1108]% cat .gitignore
*.gem
*~
.bundle
.jekyll-cache
.sass-cache
```

```
_site
Gemfile.lock
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1109]%
```

- GitHub 上でリポジトリを作成
 - ▷ README.md も含めてコンテンツは何も作らなくて良い
 - ▷ ここでは kmsq2c/openagr
- コミットし GitHub へプッシュ

```
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1085]% git init
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1089]% git add -A
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1090]% git commit -m "first
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1091]% git branch -M main
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1092]% git remote add origin
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1093]% git push -u origin ma
kiri@smtp:~/projects/jekyll/custompages/openagr.org[1094]%
```

- GitPages を作る
 - ① Settings ⇒ Pages
 - ② Source ⇒ "Deploy from branch"
 - ③ Branch ⇒ main/root ⇒ Save
 - ④ サイトの URL が表示されれば成功!

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is live at <https://kmsq2c.github.io/openagr/>

Last deployed by  github-pages 4 minutes ago

[Visit site](#)

...

Build and deployment

Source

Deploy from a branch ▾

Branch

Your GitHub Pages site is currently being built from the main branch. [Learn more.](#)

 main ▾

 / (root) ▾

Save

5.3 一般的に良くあるテーマ

テーマ選択の条件 (5.1 節) の (3), (4) を満たしていないテーマとして thedevslot/WhatATheme[9] があります。デモページは [8] です。

5.3.1 ロケール部分を変数化する

- メニューや内容の埋め込み部分を全て変数にして _config.yml に記述

```
--- ./_config.yml~ 2022-08-04 16:21:54.200032000 +0900
```

```
+++ ./_config.yml 2022-05-17 14:57:40.253974000 +0900
```

```
@@ -6,6 +6,27 @@
```

```
# Base URL of the Site (i.e., Name of the Repository in which the Site is hosted)
```

```
baseurl:
```

```

+# Menu bars and buttons
+home: HOME
+about: ABOUT
+contact: CONTACT
+blog: BLOG
+more: MORE
+contact_me: Contact Me
+read_more: Read More
+ping_me_on_messenger: Ping me on Messenger
+send_me_an_email: Send me an Email
+tweet_me_on_twitter: Tweet me on Twitter
+back_to_top: BACK TO TOP
+
+# Site's Descriptions
+have_a_good_read: Have a Good Read
+search_here: Search Here
+you_can_find_me_here: You Can find me here
+more_links: More Links
+subscribe_via_rss: Subscribe via RSS
+recent_posts: Recent Posts
+
# Title of the Site
title: What A Theme
# Description of the Site
--- ./_includes/dropdown.html~ 2022-08-04 15:38:36.262535000 +0900
+++ ./_includes/dropdown.html 2022-05-17 14:18:00.699154000 +0900
@@ -1,10 +1,10 @@
<div class="navbar-item has-dropdown is-hoverable">
  <a class="navbar-link">
-     MORE
+     {{site.more}}
  </a>
  <div class="navbar-dropdown has-background-black is-left">
    {% for sitepage in site.html_pages %}
-     {% unless sitepage.url == '/' or sitepage.url == '/404' or sitepage.url == '/t
+     {% unless sitepage.url == '/' or sitepage.url == '/404.html' or sitepage.url =
    <a href="{{sitepage.url | absolute_url}}"
      class="navbar-item has-text-grey-light {% if page.url == sitepage.url | abs
      {{sitepage.title}}
--- ./_includes/footer.html~ 2022-08-04 15:37:40.241081000 +0900
+++ ./_includes/footer.html 2022-05-10 14:42:31.569714000 +0900
@@ -1,7 +1,7 @@
<footer id="footer">
  <!--Footer Button-->
  <div class="container has-text-centered has-background-grey-darker" id="backtotop">
-     <a class="has-text-white" onclick="window.scroll(0,0)">BACK TO TOP</a>

```

```

+         <a class="has-text-white" onclick="window.scroll(0,0)">{{site.back_to_top}}</a>
    </div>
    <!--Footer Main Section-->
    <div class="has-background-grey-darker">
@@ -27,7 +27,7 @@

        <!--Link Section-->
        <div class="column has-text-white">
-            <h3>More Links</h3>
+            <h3>{{site.more_links}}</h3>
            {% for sitepage in site.html_pages offset: 4 %}
            {% unless sitepage.url == '/' limit: 2 %}
            <li>
@@ -36,13 +36,13 @@
                {% endunless %}
                {% endfor %}
            <li>
-                <a target="_blank" href="{{site.url}}{{site.baseurl}}/feed.xml">Sub
+                <a target="_blank" href="{{site.url}}{{site.baseurl}}/feed.xml">{{s
            </li>
        </div>

        <!--Blog-post Section-->
        <div class="column has-text-white">
-            <h3>Recent Posts</h3>
+            <h3>{{site.recent_posts}}</h3>
            {% for post in site.posts offset: 0 limit:3 %}
            <li>
                <a href="{{post.url | absolute_url}}">{{post.title}}</a>
--- ./_includes/showcase.html~ 2022-08-04 15:38:57.680995000 +0900
+++ ./_includes/showcase.html 2022-05-09 16:35:12.105073000 +0900
@@ -9,7 +9,7 @@
        </h3>
        <hr>
        <a href="{{site.url}}{{site.baseurl}}/#about"
-            class="button is-white is-rounded is-outlined is-uppercase has-text-weight-normal"
+            class="button is-white is-rounded is-outlined is-uppercase has-text-weight-normal"
        </a>
    </div>
</div>
--- ./_includes/blogpage-heading.html~ 2022-08-04 15:38:13.910293000 +0900
+++ ./_includes/blogpage-heading.html 2022-05-09 16:30:03.230245000 +0900
@@ -2,7 +2,7 @@
    <nav class="level has-text-centered">
        <!-- Left side -->
        <div class="level-left">

```

```

-         <p class="subtitle is-uppercase has-text-weight-medium has-text-grey">Have a Go
+         <p class="subtitle is-uppercase has-text-weight-medium has-text-grey">{{site.ha
</div>
<!-- Right side -->
<div class="level-right">
--- ./_includes/search.html~ 2022-08-04 15:38:49.812704000 +0900
+++ ./_includes/search.html 2022-05-09 16:32:47.563692000 +0900
@@ -1,7 +1,7 @@
<div class="field has-addons has-text-weight-medium has-text-grey" id="search">
  <p class="control">
    <input id="search-input" class="input is-shadowless is-uppercase has-text-grey
-      placeholder="Search Here">
+      placeholder="{{site.search_here}}">
  </p>
  <p class="control">
    <button class="button">
--- ./_includes/blog-card.html~ 2022-08-04 15:38:06.222160000 +0900
+++ ./_includes/blog-card.html 2022-05-10 14:32:05.050642000 +0900
@@ -2,7 +2,7 @@
  <div class="column is-marginless is-paddingless is-one-third-desktop is-one-third-f
    <div class="image is-16by9" style="background-image: url('{{post.post-image}}');
  </div>
-  <a href="{{post.url | absolute_url}}">
+  <a href="{{post.url}}">
    <div class="column has-text-left-desktop has-text-left-tablet">
      <h1 class="title is-size-4-touch">{{post.title}}</h1>
      <div class="content has-text-grey">
--- ./_includes/navbar.html~ 2022-08-04 15:37:12.623991000 +0900
+++ ./_includes/navbar.html 2022-05-09 17:11:46.668299000 +0900
@@ -22,11 +22,11 @@
  <!-- navbar items | right side -->
  <div class="navbar-end">
    <a class="navbar-item {% if page.layout == 'default' %}is-active{% endif %}
-      href="{% if page.layout == 'default' %}{{site.url}}{{site.baseurl}}
-      <a class="navbar-item" href="{{site.url}}{{site.baseurl}}/#about">ABOUT
-      <a class="navbar-item" href="{{site.url}}{{site.baseurl}}/#contact">CON
+      href="{% if page.layout == 'default' %}{{site.url}}{{site.baseurl}}
+      <a class="navbar-item" href="{{site.url}}{{site.baseurl}}/#about">{{sit
+      <a class="navbar-item" href="{{site.url}}{{site.baseurl}}/#contact">{{s
    <a class="navbar-item {% if page.layout == 'blog' %}is-active{% endif %}
-      href="{{site.url}}{{site.baseurl}}/blog">BLOG</a>
+      href="{{site.url}}{{site.baseurl}}/blog">{{site.blog}}</a>
    {% include dropdown.html %}
  </div>
--- ./_includes/about.html~ 2022-08-04 15:37:51.783010000 +0900

```

+++ ./_includes/about.html 2022-05-10 15:36:08.745301000 +0900

@@ -9,17 +9,17 @@

```

    </div>
    <div class="column has-text-left-desktop has-text-left-tablet has-text-
      <p>{{site.author-about}}</p>
+
      <br>
      <div class="container columns is-centered">
        <div>
          <a href="{{site.url}}{{site.baseurl}}/#contact"
-             class="button is-rounded is-uppercase has-text-weight-n
-             Me</a>
+             class="button is-rounded is-uppercase has-text-weight-n
          </a>
        </div>
        <div>
          <a href="{{site.url}}{{site.baseurl}}/blog"
-             class="button is-rounded is-uppercase has-text-weight-n
+             class="button is-rounded is-uppercase has-text-weight-n
          </a>
        </div>
      </div>

```

--- ./_includes/contact.html~ 2022-08-04 15:38:27.540554000 +0900

+++ ./_includes/contact.html 2022-05-17 14:21:22.098524000 +0900

@@ -1,21 +1,21 @@

```

<section class="hero is-fullheight has-text-centered" id="contact">
  <div class="hero-body">
    <div class="container">
-     {% unless site.facebook_username == %}
+     {% unless site.facebook_username == "" %}
      <a id="contact-button" href="https://m.me/{{site.facebook_username}}"
        class="button is-rounded is-uppercase has-text-weight-normal is-black i
        target="_blank">
        <i class="fab fa-facebook-messenger"></i>{{site.ping_me_on_messenger}}
      </a>
      {% endunless %}
-     {% unless site.email == %}
+     {% unless site.email == "" %}
      <a id="contact-button" href="mailto:{{site.email}}"
        class="button is-rounded is-uppercase has-text-weight-normal is-black i
        target="_blank">
        <i class="fas fa-envelope"></i>{{site.send_me_an_email}}
      </a>
      {% endunless %}
-     {% unless site.twitter_username == %}
+     {% unless site.twitter_username == "" %}

```

```

    <a id="contact-button" href="https://www.twitter.com/intent/tweet?screen_na
      class="button is-rounded is-uppercase has-text-weight-normal is-black i
      target="_blank">
@@ -27,34 +27,34 @@
      {{site.you_can_find_me_here}}
    </h3>
    <div class="column is-full is-size-3">
-      {% unless site.facebook_username == %}
+      {% unless site.facebook_username == "" %}
      <a href="http://www.facebook.com/{{site.facebook_username}}" target="_b
      {% endunless %}
-      {% unless site.twitter_username == %}
+      {% unless site.twitter_username == "" %}
      <a href="http://www.twitter.com/{{site.twitter_username}}" target="_bla
      {% endunless %}
-      {% unless site.instagram_username == %}
+      {% unless site.instagram_username == "" %}
      <a href="http://www.instagram.com/{{site.instagram_username}}" target="
      {% endunless %}
-      {% unless site.linkedin_username == %}
+      {% unless site.linkedin_username == "" %}
      <a href="http://www.linkedin.com/in/{{site.linkedin_username}}" target=
      {% endunless %}
-      {% unless site.github_username == %}
+      {% unless site.github_username == "" %}
      <a href="http://www.github.com/{{site.github_username}}" target="_blank
      {% endunless %}
-      {% unless site.youtube_channel_id == %}
+      {% unless site.youtube_channel_id == "" %}
      <a href="http://www.youtube.com/channel/{{site.youtube_channel_id}}" ta
      {% endunless %}
-      {% unless site.reddit_username == %}
+      {% unless site.reddit_username == "" %}
      <a href="http://www.reddit.com/u/{{site.reddit_username}}" target="_bla
      {% endunless %}
-      {% unless site.behance_username == %}
+      {% unless site.behance_username == "" %}
      <a href="http://www.behance.com/{{site.behance_username}}" target="_bla
      {% endunless %}
-      {% unless site.dribbble_username == %}
+      {% unless site.dribbble_username == "" %}
      <a href="http://www.dribbble.com/{{site.dribbble_username}}" target="_b
      {% endunless %}
-      {% unless site.spotify_username == %}
+      {% unless site.spotify_username == "" %}
      <a href="http://open.spotify.com/user/{{site.spotify_username}}" target

```

```

        {% endunless %}
    </div>
--- ./_layouts/project.html~ 2022-08-04 15:41:03.651349000 +0900
+++ ./_layouts/project.html 2022-05-09 16:29:06.388160000 +0900
@@ -17,7 +17,7 @@
    <div class="hero-body">
      <div class="container has-text-centered">
        <div class="section">
-         <p class="subtitle is-uppercase has-text-weight-medium has-text-gre
+         <p class="subtitle is-uppercase has-text-weight-medium has-text-gre
        </div>
        {% include project-card.html %}
      </div>

```

5.3.2 Gem ベーステーマの gem を作る

(1) Gem ベーステーマのテンプレートを作成

```

kiri@smtp:~/work/jekyll[1198]% jekyll new-theme jekyll-theme-whatatime
create /home/kiri/work/jekyll/jekyll-theme-whatatime/assets
create /home/kiri/work/jekyll/jekyll-theme-whatatime/_layouts
create /home/kiri/work/jekyll/jekyll-theme-whatatime/_includes
create /home/kiri/work/jekyll/jekyll-theme-whatatime/_sass
create /home/kiri/work/jekyll/jekyll-theme-whatatime/_layouts/page.html
create /home/kiri/work/jekyll/jekyll-theme-whatatime/_layouts/post.html
create /home/kiri/work/jekyll/jekyll-theme-whatatime/_layouts/default.html
create /home/kiri/work/jekyll/jekyll-theme-whatatime/Gemfile
create /home/kiri/work/jekyll/jekyll-theme-whatatime/jekyll-theme-whatatime
create /home/kiri/work/jekyll/jekyll-theme-whatatime/README.md
create /home/kiri/work/jekyll/jekyll-theme-whatatime/LICENSE.txt
initialize /home/kiri/work/jekyll/jekyll-theme-whatatime/.git
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
create /home/kiri/work/jekyll/jekyll-theme-whatatime/.gitignore
Your new Jekyll theme, jekyll-theme-whatatime, is ready for you in /home/kiri/work/jekyll
For help getting started, read /home/kiri/work/jekyll/jekyll-theme-whatatime/README.md.
kiri@smtp:~/work/jekyll[1199]%

```

(2) *.gemspec を修正

```

kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1201]% diff -u ~/work/jekyll/jekyll-
--- /home/kiri/work/jekyll/jekyll-theme-whatatime/jekyll-theme-whatatime.gemspec
+++ jekyll-theme-whatatheme.gemspec      2022-05-09 10:18:34.026688000 +0900
@@ -1,14 +1,15 @@
 # frozen_string_literal: true

Gem::Specification.new do |spec|
- spec.name          = "jekyll-theme-whatatime"
+ spec.name          = "jekyll-theme-whatatheme"
  spec.version       = "0.1.0"
  spec.authors       = ["Kazuhiko Kiriyama"]
- spec.email         = ["kiri@kx.truefc.org"]
+ spec.email         = ["kiri@truefc.org"]

- spec.summary       = "TODO: Write a short summary, because Rubygems requires one."
- spec.homepage      = "TODO: Put your gem's website or public repo URL here."
- spec.license       = "MIT"
+ spec.summary       = "Jekyll Portfolio theme (thedevsloot@gmail.com)"
+ spec.description   = "A customizable Jekyll Portfolio theme which supports blogging w
+ spec.homepage      = "https://github.com/thedevsloot/WhatATheme/"
+ spec.license       = "GPL-2.0-only"

  spec.files         = `git ls-files -z`.split("\0").select { |f| f.match(%r!^(assets|
kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1202]%

```

(3) 必要なファイルをコピー

(4) gem を作成

```

kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1426]% rm -rf .git
kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1427]% git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/kiri/projects/jekyll/jekyll-theme-whatatheme/.
kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1428]% git add -A
kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1429]% git commit -m "Init commit"
[master (root-commit) e0600cb] Init commit
45 files changed, 2235 insertions(+)

```

```
create mode 100644 .gitignore
create mode 100644 404.md
create mode 100644 Gemfile
create mode 100644 LICENSE
create mode 100644 LICENSE.txt
create mode 100644 README.md
create mode 100644 _config.yml
create mode 100644 _data/projects.yml
create mode 100644 _includes/about.html
create mode 100644 _includes/blog-card.html
create mode 100644 _includes/blogpage-heading.html
create mode 100644 _includes/contact.html
create mode 100644 _includes/dropdown.html
create mode 100644 _includes/footer.html
create mode 100644 _includes/head.html
create mode 100644 _includes/meta.html
create mode 100644 _includes/navbar.html
create mode 100644 _includes/project-card.html
create mode 100644 _includes/search.html
create mode 100644 _includes/showcase.html
create mode 100644 _layouts/404.html
create mode 100644 _layouts/blog.html
create mode 100644 _layouts/compress.html
create mode 100644 _layouts/default.html
create mode 100644 _layouts/page.html
create mode 100644 _layouts/post.html
create mode 100644 _layouts/project.html
create mode 100644 _posts/2020-04-21-what-is-jekyll-how-to-use-it.md
create mode 100644 _posts/2020-04-22-how-to-install-whatatheme.md
create mode 100644 _posts/2020-04-24-sample-post.md
create mode 100644 _sass/main.scss
create mode 100644 assets/css/style.scss
create mode 100644 assets/images/1280x720 Placeholder.png
create mode 100644 assets/images/How to install and use WhatATheme.png
create mode 100644 assets/images/SamplePost.png
create mode 100644 assets/images/What is Jekyll and How to use it.png
create mode 100644 assets/images/logo.png
create mode 100644 assets/images/meta.jpg
create mode 100644 assets/js/simple-jekyll-search.js
create mode 100644 blog.md
create mode 100644 index.md
create mode 100644 jekyll-theme-whatatheme.gemspec
create mode 100644 project.md
create mode 100644 search.json
create mode 100644 test-page.md
kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1432]% gem build jekyll-theme-whatat
```

```
Successfully built RubyGem
Name: jekyll-theme-whatatheme
Version: 0.1.0
File: jekyll-theme-whatatheme-0.1.0.gem
kiri@smtp:~/projects/jekyll/jekyll-theme-whatatheme[1433]%
```

5.3.3 Gem ベーステーマの作成から GitHub Pages へのデブ ロイまで

5.2.2 節～5.2.3 節と同様

6 最後に

- 単一のページしか立ち上げない場合には Gem ベースリポジトリとカスタムリポジトリは分ける必要があるのか？
 - ⇒ 分ける必要はありません ;-)
- Gem ベースリポジトリとカスタムリポジトリは分けて何か嬉しいことは？
 - ▷ 同種のページを複数立ち上げたい場合
 - ▷ エンドユーザーが怪しい場合⇒カスタムリポジトリだけいじらせる
- ロケール等がうまく切り分けられているテーマであれば非常に便利

参考文献

- [1] agency-jekyll-theme. <https://raviriley.github.io/agency-jekyll-theme-starter/>.
- [2] agency-jekyll-theme (GitHub). <https://github.com/raviriley/agency-jekyll-theme>.
- [3] Free Jekyll Themes. <https://jekyll-themes.com/free/>.
- [4] Gem ベーステーマを理解する. <http://jekyllrb-ja.github.io/docs/themes/>.
- [5] Jamstack Themes. <https://jamstackthemes.dev/theme/>.
- [6] Liquid. <https://shopify.github.io/liquid/>.
- [7] Step by Step チュートリアル. <http://jekyllrb-ja.github.io/docs/step-by-step/01-setup/>.
- [8] WhatATheme. <https://thedevsloot.com/WhatATheme/>.
- [9] WhatATheme (GitHub). <https://github.com/thedevsloot/WhatATheme>.
- [10] YAML Ain't Markup Language (YAML) version 1.2. <https://yaml.org/spec/1.2.2/>.