

# ポートフォリオサイトのレベルを1mmだけ上げる講座

- デザイン
- 内容 (魅せ方)
- コンテンツ管理
- ブランディング
- UX

“デザインについてはここでは扱いません!!

どう魅せるかについてはあくまでも自分の体験的な部分で話します



amiokzk ...

Message

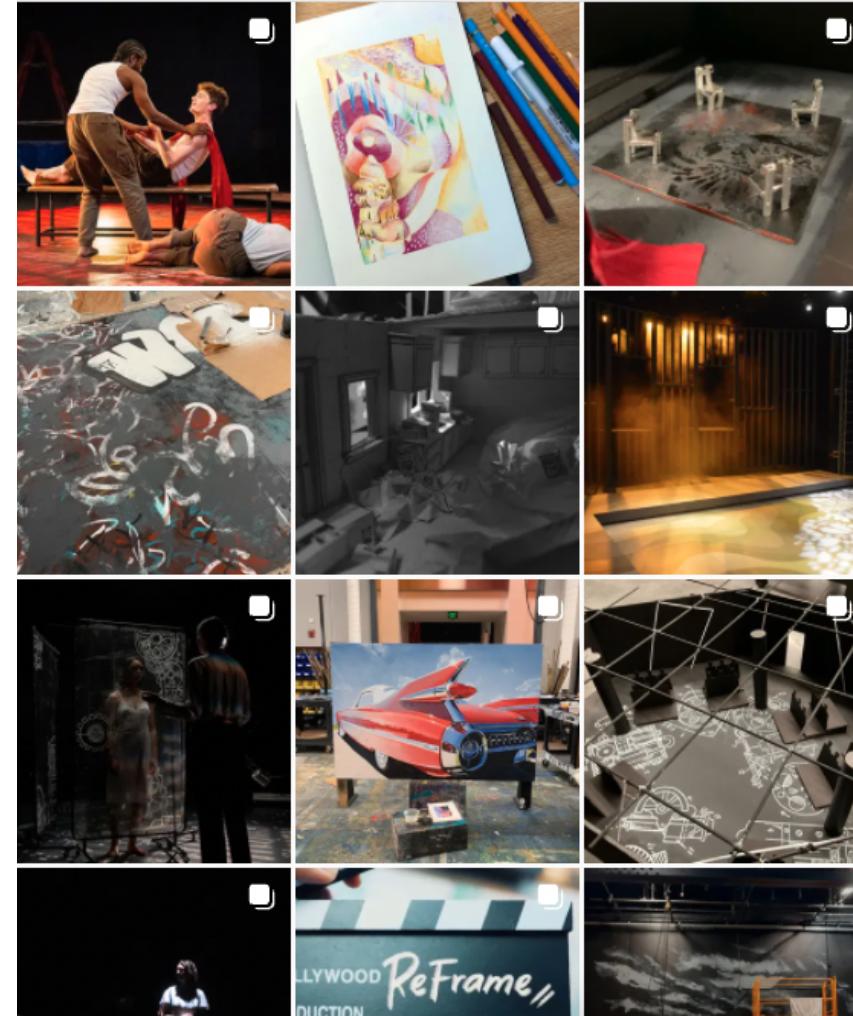


Ami Okazaki

Designer

岡崎アミ | 21 | scenic design | tokyo // boston

@butheatre | personal @amiokazakii\_



# ポートフォリオとは

ポートフォリオとは、

本来、図面や地図など綴じられていないペーパーを  
ひとまとめにして入れるケースのこと

である。

イラストレーターなどが、自分の作品の一部を営業のツー  
ルとしてまとめたものをポートフォリオと呼んでいる。

^1

# エンジニアのポートフォリオ

エンジニアの業界で言うポートフォリオとは、自分の作品の一部を営業のツールとして見せることは一緒だが、見せるものとして、

- フロントエンドエンジニア: デザイン、UX
- バックエンドエンジニア: 処理の速さ、きれいさ
- インフラエンジニア: どのクラウドについての知識があるか

などに加え、使える技術スタックをアピールする場であることが異なる。

# データサイエンティストのポートフォリオ

エンジニア的技術スタックではなく、

- どのような分析技術が使えて
- 今までどのようなデータに対してインサイトを出してきたのか

などを掲載すべき...? (データサイエンティストとして就活したわけではないでしらん)

## ただ、君たち新卒でしょ？

新卒の場合、技術スタックなどの技術力を完全に重視しているわけではなく、どちらかというとポテンシャル(やる気)が重視される傾向にある

(大企業など、レベルの高い企業はまた別だと思いますが...)

→ MUDS生として技術スタックと同時にやる気、将来の目標などを見せることが重要

# ポートフォリオは営業のパンフレット

いかにヤバそうなやつに見せられるかが勝負

306.jp

News Works Biography Blogs Contact ENGLISH



**Featured Work: アプリ**

**> cd .. /**

新型コロナウイルスが未だ流行している中、大学では徐々に日常が戻りつつあります。そんなニューノーマルが求められる時代に、少しでも元の日常に近付くために開発したアプリです。

武藏野大学では、対面授業が一部（武藏野大学データサイエンス学部: MUDSにおいては未来創造プロジェクトのみ）解禁されたと同時に、体温測定がキャンパス入構の条件となりました。そんな中毎日、体温を測定し、紙やスプレッドシートに記録していくのは非常に煩わしく感じました。そこで、体温管理などをより手軽にできるアプリを開発することにしました。

- ✓ Django / Slack APIについて学ぶことができた
- ✓ パスワードレス認証の実装を行った
- ✓ パスワードレス認証やファイアウォールの設置、厳しいアクセス制限をかけるなど、セキュリティに最大限配慮した

LEARN MORE

何を、どういう背景を持ってやって、何を達成して、どのようなものが出来上がったのかを1スクリーンで見せる

# ポートフォリオは営業のパンフレット

できれば、自分で1から、身の回りにあるちょっとした課題を解決するためのアプリを作つてみる

未来創造プロジェクトの内容でも十分

## ポートフォリオは営業のパンフレット

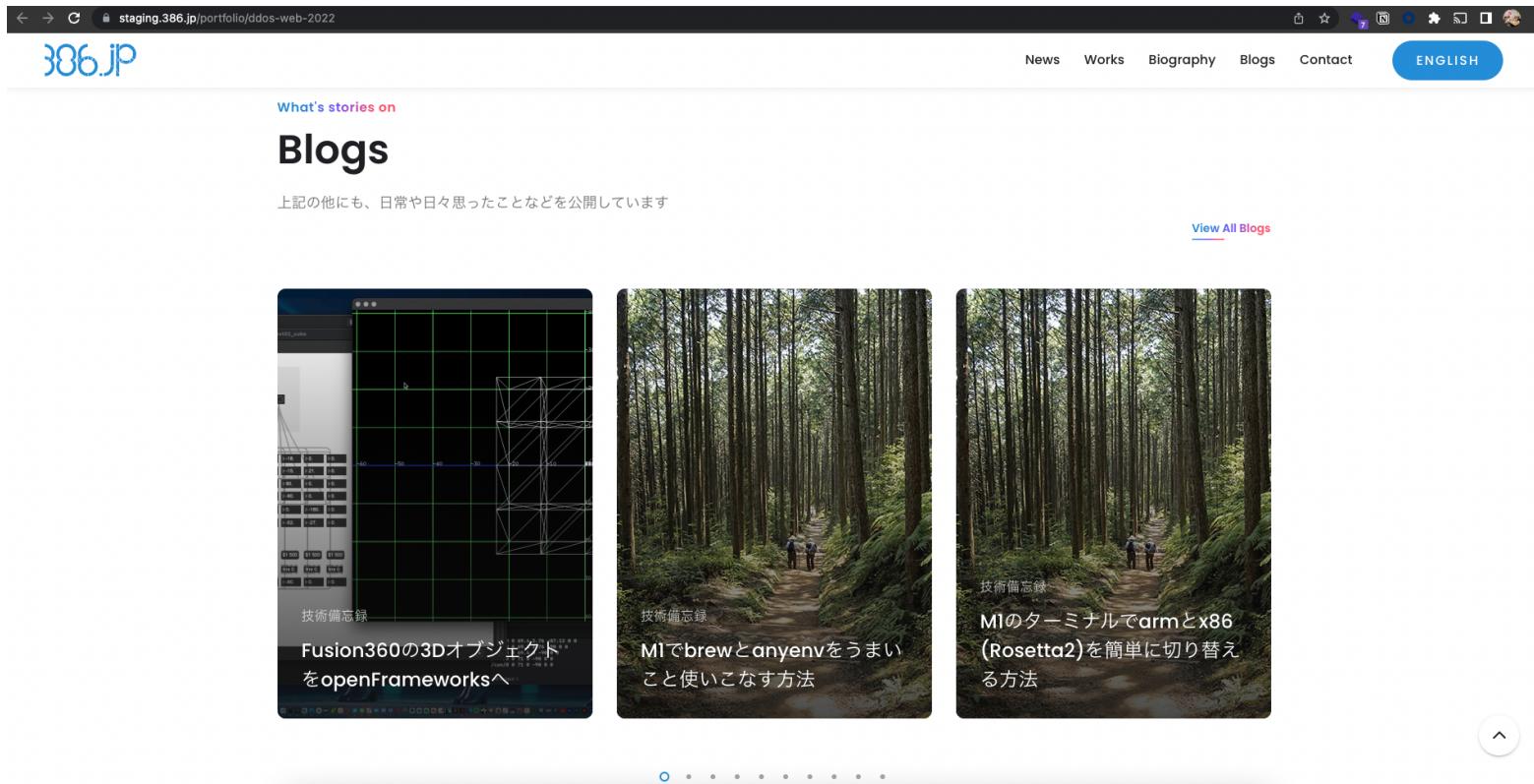
結局、コーディング力はコーディングテストで測られることが多い

ポートフォリオは思いを伝える場所

# ブログもポートフォリオの1つ

「Works」にならないけど日々勉強しててー、っていう内容はブログとして積極的に発信しよう

QiitaとかZennを使ってもいいけど、ポートフォリオサイトとして1画面にまとめよう



# 基本的に採用担当者は書類を見ない

思ってるほどじっくりポートフォリオを見ない、もしくは全く見ない会社とかもあります

1ページにとりあえずまとめて、「詳細はこちら」の先にくっそ熱い思いを語る

## 熱い思いの内容

- どのような背景・問題意識で作ったのか
- どのくらいの期間かかったのか
- どのような技術スタックなのか
- どれくらいの人数、開発手法で開発したのか
- どこにこだわった? / どういうトラブルがあった?
- どれくらいの人数を使ってもらえたのか、その影響は?
- ユーザーの感想は?
- ここから学んだこと

## ただ、見てもらえない=手を抜いていい、にはならない

内容がなかったり、ありきたりなテンプレを使ってたりしていると、選考に進んでポートフォリオを見られたときにマイナス要素に...?

面接前に見ていて、面接中の話題の1つになったり...?

# 出る杭になる

- 学生なのにサービス作ってる (使ってもらえてる)
- 研究頑張ってる
- インターンめっちゃ行って成果出してる
- 論文頑張ってる (賞もとってる)
- ポートフォリオのデザインがやべえ

## まずはデザインを頑張ってみよう

1からHTML, CSS, JavaScriptで書くのもいいんだけど、多忙を極めてるMUDS生は1mmだけテンプレのお世話になろう

サイトをすべてテンプレで作るのではなく、テンプレを切り貼りしてカスタマイズしまくって自分の色を出す

できればレスポンシブ対応など、制作力も示す

今日のメインの話題はそこではない

# ポートフォリオサイトの更新は闇のサイクル

ポートフォリオの制作はもちろん、コンテンツの更新もまあまあめんどくさい作業ですよね

**もしHTMLとCSS直書きのポートフォリオサイトだったら:**

1. ポートフォリオサイトに載せられそうな成果が出る
2. 喜ぶ😊
3. ポートフォリオサイトも更新しなきゃってなる
4. けどHTML直書きだからどこをどうコピペして追加するのか探らなくてはいけない
5. 萎える
6. けどやんなきゃいけない (様々なタスクに押しつぶされて忘れる
7. なにかきっかけがあってポートフォリオの修正を思い出す
8. けど修正箇所探るのがめんどくさくてなかなかやる気が出ない
9. 無限列車編突入

## ポートフォリオサイトの更新は闇のサイクル

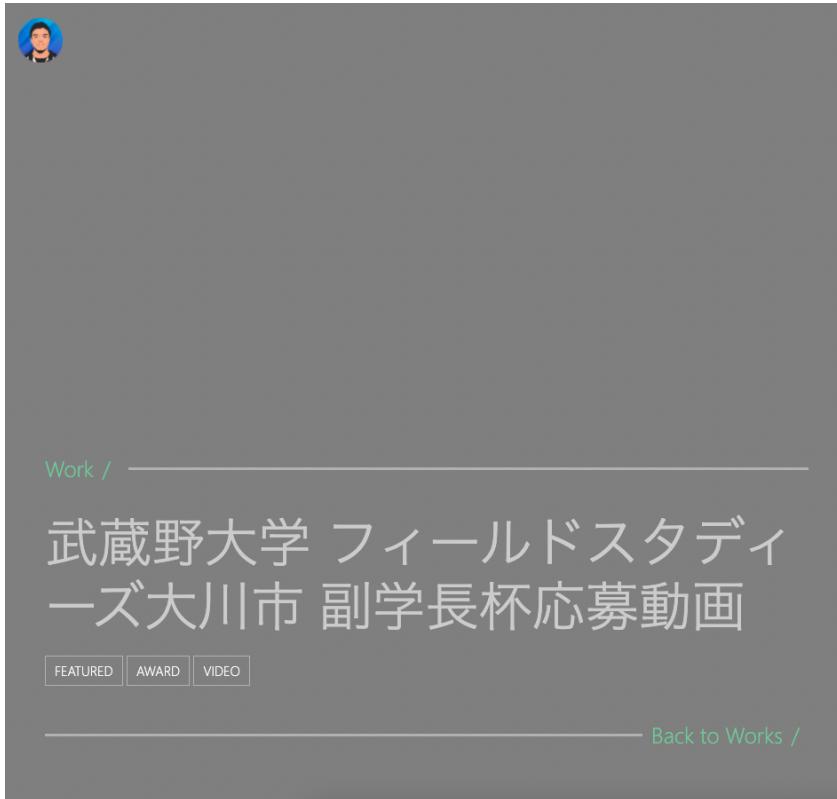
大学1, 2年の頃のポートフォリオサイトはこのような感じでした

せっかく色々カスタマイズして作っても更新がめんどくさくて意味がなくなる

それって作ってる意味ないですよね

# ポートフォリオサイトの更新は闇のサイクル

闇のサイクルに陥ってた頃のポートフォリオ、その1



Work /

## 武蔵野大学 フィールドスタディーズ大川市 副学長杯応募動画

FEATURED AWARD VIDEO

Back to Works /

武蔵野大学 公式ホームページの記事

※スピーチを当日行ったのが法学部政治学科 高橋希歩さんのため、彼女の名前のみ、代表者として記事内に記されています。

27 Sep 2019

Description

武蔵野大学の一般教養プログラム「武蔵野BASIS」の一環の、フィールドスタディーズ(FS)で、「福岡県大川市モノ作りの魅力発信プロジェクト」というプログラムに参加し、その成果物として、2分間の動画を作りました。その時開催された第2回副学長杯「武蔵野FS動画・プレゼンコンテスト」に応募し、最優秀賞を獲得しました。

Materials

- Adobe Premiere Pro CC
- Adobe Audition CC
- Adobe AfterEffects CC

Credits

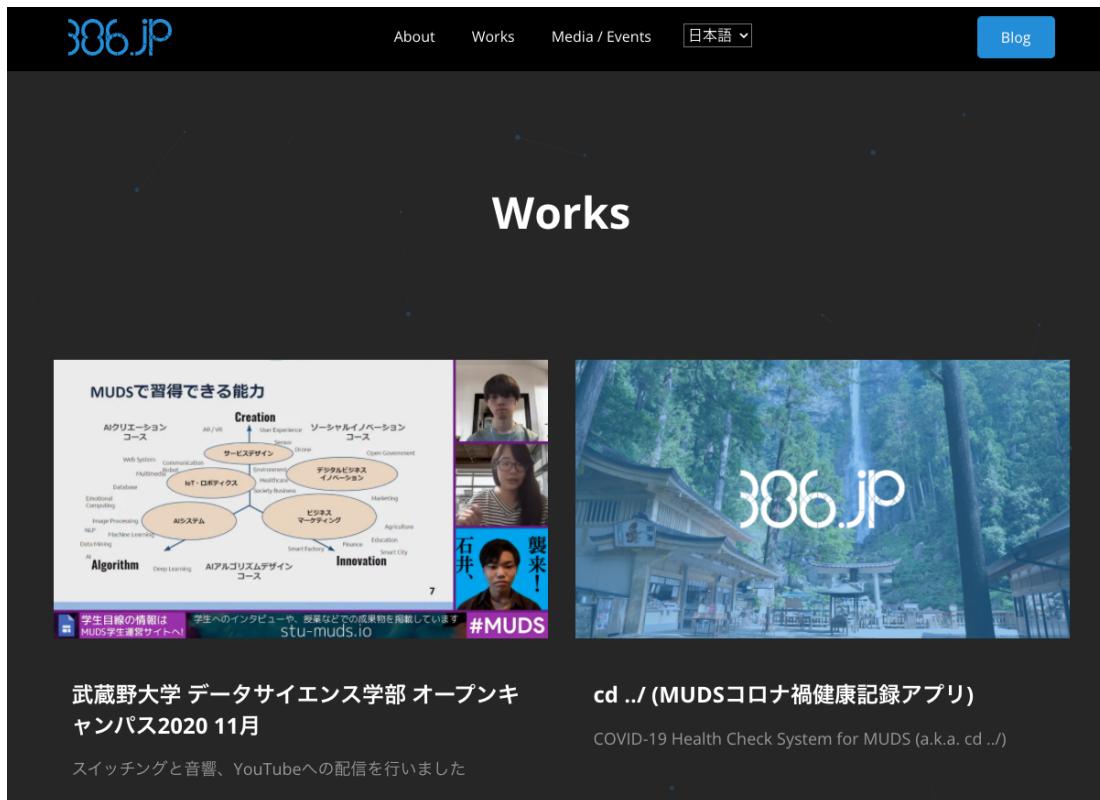
- Copyright 福岡県大川市モノ作りの魅力発信プロジェクト2019.
- Narrator: Kurumi Miyahara
- Editor / Production Planner: Kyosuke Miyamura
- Photos / Videos: Jyoichiro Azuchi, Kiho Takahashi, Kurumi Miyahara, Kyosuke Miyamura, Miyako Kawahara, Takamasa Minami
- この動画は副学長杯に応募した動画のため、武蔵野大学も広告媒体へのアップロード等の権利を有しています。

Share it:

[Twitter](#) [Facebook](#)

# ポートフォリオサイトの更新は闇のサイクル

闇のサイクルに陥ってた頃のポートフォリオ、その2



## とりあえず財力で叩いてみる

お金がある人はWordPressとかWixとかSquarespaceを使ってもいいと思います

## とりあえず財力で叩いてみる

WordPressとは、

PHPで書かれているオープンソースのブログシステム、CMSのことである。

WordPressサーバーを建ててそこにWorksのコンテンツを載せる

# WordPressを使ったポートフォリオの例

- [Takafumi Nakanishi](#)
- [Perfume Global Lab \(旧Perfume Global Site Project\)](#)
- [Satoshi Horii](#)

# なぜWordPressを選ばないのか

- 高い
- メンテナンスがめんどくさい
- 攻撃者から狙われる

→ お金のあまりかからない方法でなんとかしたいよね、もしできれば無料できたらいいよね

## 少しでも更新作業を楽にする

まずはHTML, CSS直書きからステップアップする

SSG (Static Site Generation / 静的サイト生成)

# SSGとは

SSG (Static Site Generation / 静的サイト生成)とは、

| ソフトウェアの力を使って、1つ1つの要素からサイトを組み立ててくれるもの  
である。

eg: ヘッダー、Works一覧、Workページの中の要素 (技術スタックの一覧、目的について語ってるゾーン)、Worksのデータなどをバラバラに作っておいて、それを組み立て・組み合わせて1つのサイトにしてくれる

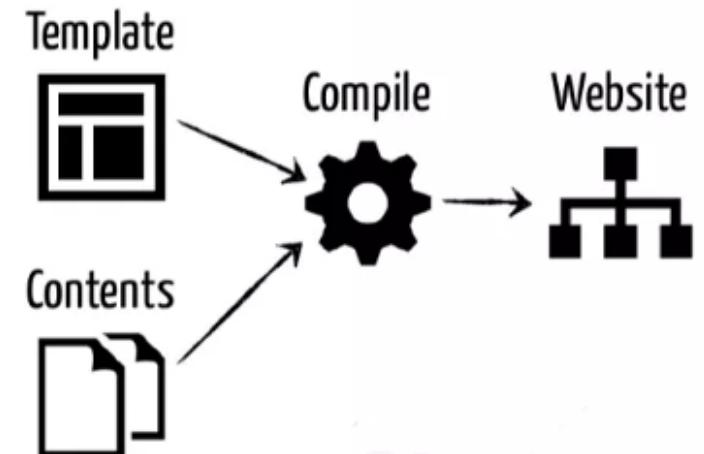
# SSGとは

HTML, CSS直書きだとただの「SS」

その「SS」の状態のサイトを分解・再構成して管理し  
やすくなった状態

SSGができるフレームワークの1つ: Jekyll

## Static Site Generator



# Jekyllとは

JekyllはSSGができるフレームワークの1つ

気軽に環境構築できるし、環境構築しないでもテンプレとサイトの内容が書いてある  
markdownファイルさえあれば、それをGitHubにアップしていい感じにすればサイトを構築  
できる



このサイトはJekyllで組み立てたもの

# Jekyllのメリット

- テンプレが豊富
- GitHub Pagesを使えばローカルの環境構築なしでもサイトを構築できる

# [Activity] JekyllとMarkdown、GitHubで作る秒サイト

いっちょ、作ってみますか!

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

---

**Repository template**

Start your repository with a template repository's contents.

No template ▾

---

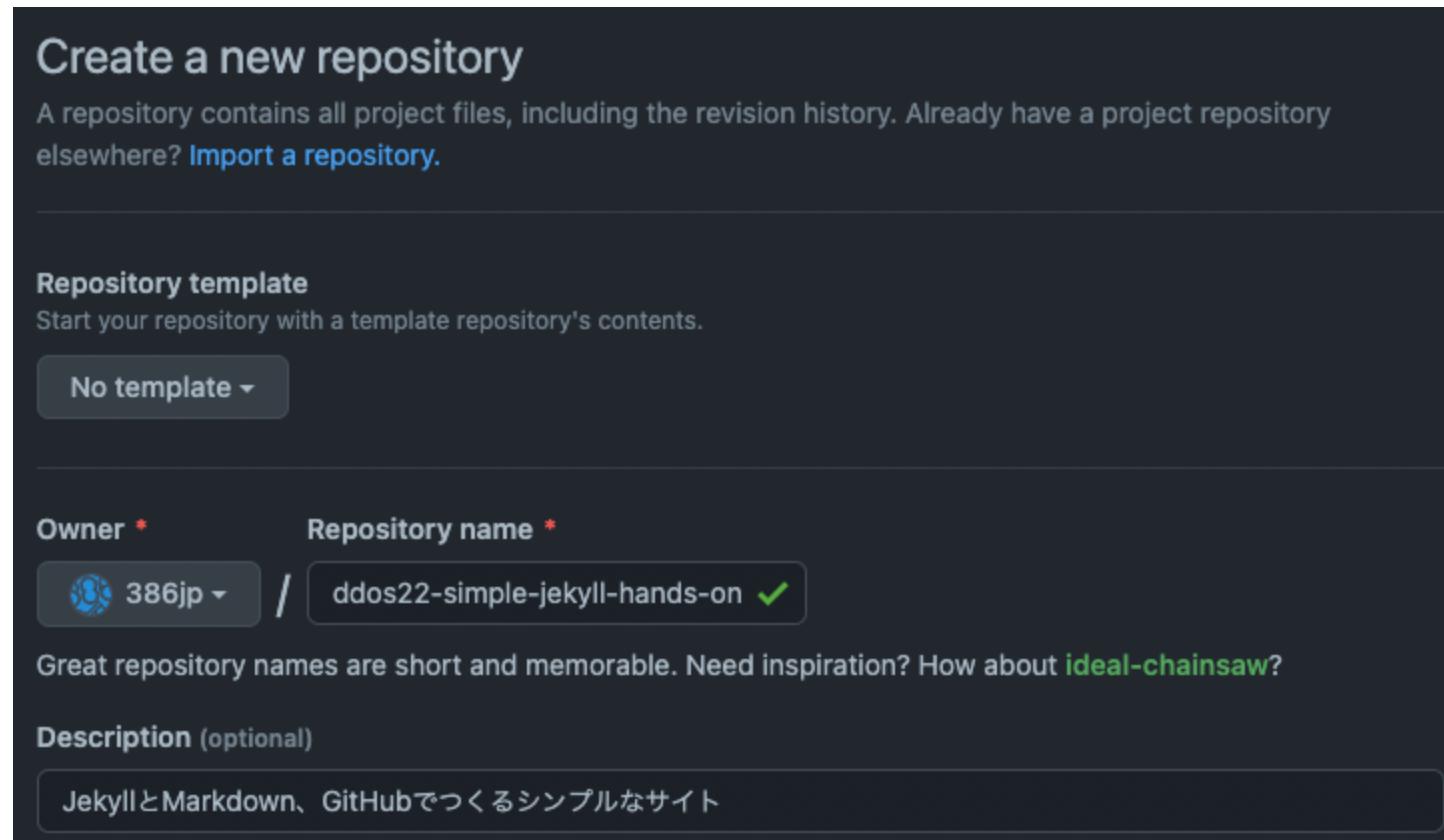
**Owner \***

386jp / ddos22-simple-jekyll-hands-on ✓

Great repository names are short and memorable. Need inspiration? How about [ideal-chainsaw](#)?

**Description (optional)**

JekyllとMarkdown、GitHubでつくるシンプルなサイト



なにはともあれ、まずはレポジトリ作成

# [Activity] JekyllとMarkdown、GitHubで作る秒サイト

レポジトリができたら、 README.md という名前でドキュメントを作成



The screenshot shows a GitHub repository page for 'ddos22-simple-jekyll-hands-on'. The 'README.md' tab is selected, indicating the file is being previewed. The content of the file is displayed as follows:

```
1 # こんにちはみなさん  
2  
3 DDoS、ポートフォリオ講座、いかがお過ごしでしょうか?  
4  
5 markdownファイルだけでポートフォリオが作れたら、サイトの管理楽になりそうじゃないですか?
```

内容はMarkdown記法に従っていればなんでもOK

# [Activity] JekyllとMarkdown、GitHubで作る秒サイト

ファイルを保存したら、レポジトリの設定 → 「Pages」へ行く

「Sources」をこんな感じに設定して保存を押す

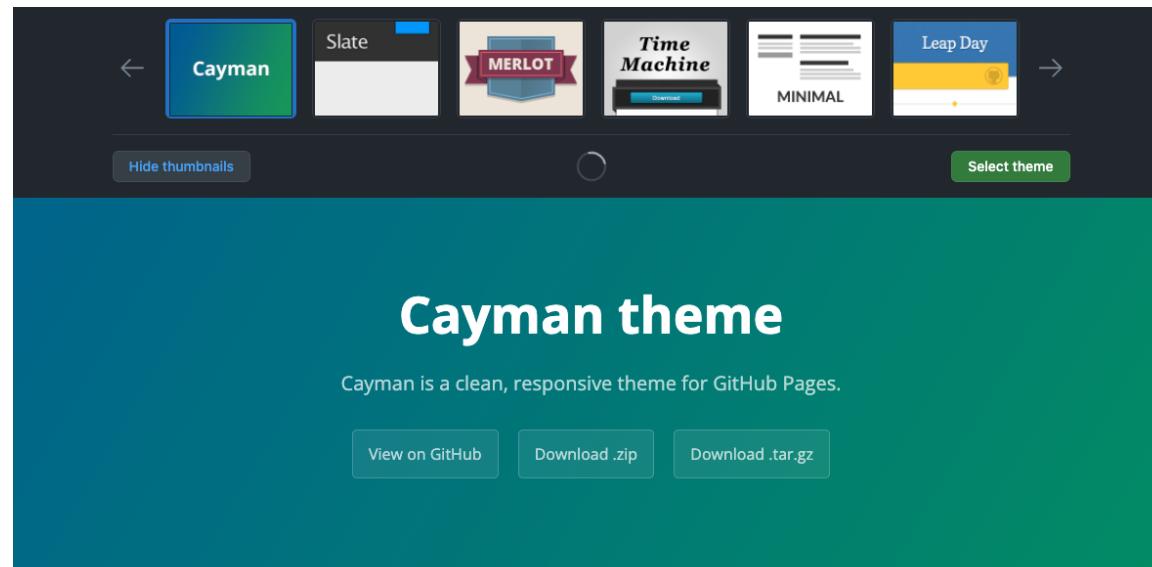
The screenshot shows the GitHub repository settings page for the repository "386jp/ddos22-simple-jekyll-hands-on". The "Settings" tab is selected. On the left, there is a sidebar with various repository settings: General, Access, Collaborators, Code and automation, Branches, Tags, Actions, Webhooks, and Pages. The "Pages" option is highlighted. The main content area is titled "GitHub Pages" and contains the following information:

- A message stating: "GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository."
- An info message: "Your site is ready to be published at <https://386jp.github.io/ddos22-simple-jekyll-hands-on/>"
- A warning message: "Caution: This repository is private but the published site will be public."
- A "Source" section indicating the site is built from the "main" branch.
- Buttons for "Branch: main", "Path: / (root)", and "Save".
- A "Theme Chooser" section with a "Choose a theme" button.

# [Activity] JekyllとMarkdown、GitHubで作る秒サイト

「Theme Chooser」の「Choose a Theme」を押してテーマを選択

お好きなのをどうぞ



いいのが見つかったら「Select Theme」で保存

# [Activity] JekyllとMarkdown、GitHubで作る秒サイト

設定ページに書いてあるURLにアクセスすればサイトの完成!



こんにちはみなさん

DDoS、ポートフォリオ講座、いかがお過ごしでしょうか?

markdownファイルだけでポートフォリオが作れたら、サイトの管理楽になりそうじゃないですか?

---

[ddos22-simple-jekyll-hands-on](#) is maintained by [386jp](#).

This page was generated by [GitHub Pages](#).

別途、独自ドメインを設定することもできるし、カスタムテーマを入れることもできるから、可能性は無限大!

# JAM vs LAMP

今まで見てきたJekyllはJAMstackと呼ばれる構成

## JAMstack

- J: JavaScript
- A: API
- M: Markup

一方で、WordPressのようなブログサイトはLAMPstack

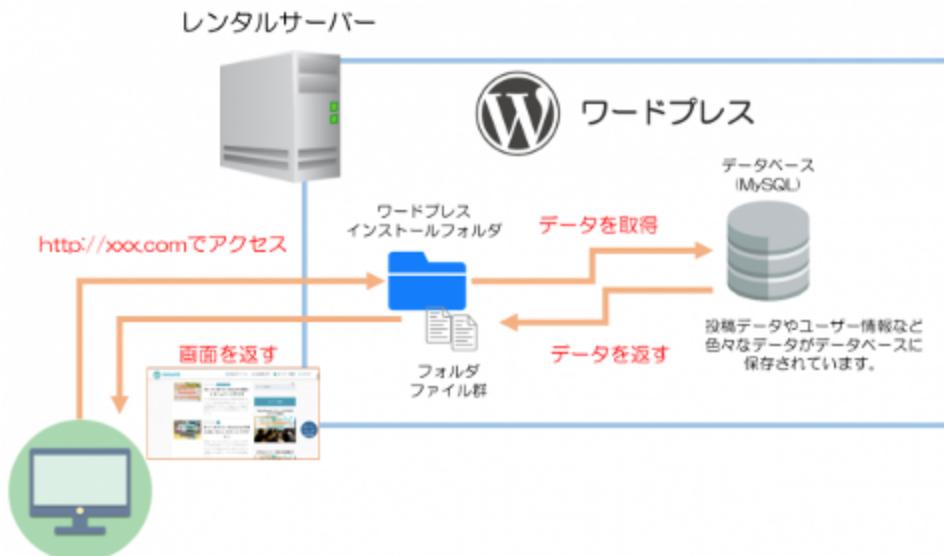
## LAMPstack

- L: Linux
- A: Apache (Web Server)
- M: MariaDB / MySQL
- P: PHP / Perl / Python

# JAM vs LAMP

今まで見てきたJekyllはJAMstackと呼ばれる構成

LAMPstackは毎回コンテンツをDBに取得しに行ってるので遅い、コンテンツと表示部を分離できなくて自由度が下がる、などのデメリットが...



## JAM vs LAMP

JAMstackが最近の主流になってきている (Web概論のAPIの話)

JekyllはJAMstackに分類できる

じゃあ、Jekyllでいいじゃん

そうなんです...

ただ、もう少しレベルアップしませんか？

# 世の中にあるふれるSSGフレームワークの輪

- Jekyll
- Hugo
- Gatsby
- Nuxt.js
- Next.js
- (React.js)

# Hugoとは

HugoはJekyllと近い操作感

GitHub Pagesでホストできないので、Jekyllより少し難易度高め

複数言語対応がデフォルトでされていたり、複数の投稿の種類（ブログとWorksみたいな）に対応しているのでポートフォリオ向き

# Nuxt.js

Vue.js というJavaScriptフレームワークで書かれているフレームワーク

日本の企業のサイトをよく見るとNuxtが使われていることが多い気がする

SSGの機能として使っているところはあまり見ないけど、参考までに掲載

# React.js, Next.js, Gatsby

React.jsというフレームワークで書かれているフレームワークがNext.jsとGatsby

## React

React自体人気、Vueに比べて大規模向き

ReactだけでSSGしてる例はほとんど見ない

## Next

Nextは海外では増えてきてるが日本だといまいち

Nuxt.jsと比較したときにちょっと遅いらしい、ただGoogleのサイトスコア的には上  
SSG対応はいまいち (どちらかというとNuxt.jsのほうがつよい)

Vercelというホスティングサービスと組み合わせると超最強

## Gatsby

NextよりSSG強い、SSG特化型

# 世の中にあるSSGフレームワークの輪

- Jekyll
- Hugo
- Gatsby
- Nuxt.js
- Next.js
- (React.js)

個人的にNext.js (with Vercel)かGatsbyを将来性や機能性を含めて推します!

自分の学んでみたい技術スタックやこれからの説明を聞いて、組み合わせを考えてみよう

# [余談] SSGサイトを手軽にホスティングできるサービス

## GitHub Pages

- Jekyll専用

## Netlify

- なんでもいける
- CMSとか、フォーム管理機能、Slack通知機能付き

## Vercel

- Next.jsとの組み合わせ最高
- CloudFlareとの相性が悪い

## [余談] SSGサイトを手軽にホスティングできるサービス

### Forestry

- CMS付き

### CloudFlare Pages

- 帯域無制限
- 一部SSR非対応、フレームワーク非対応
- 新参者

# [余談] SSGサイトを手軽にホスティングできるサービス

## Firebase

- なんでもいける (多分)  
ドメイン対応がいまいち (SSL)

## AWS

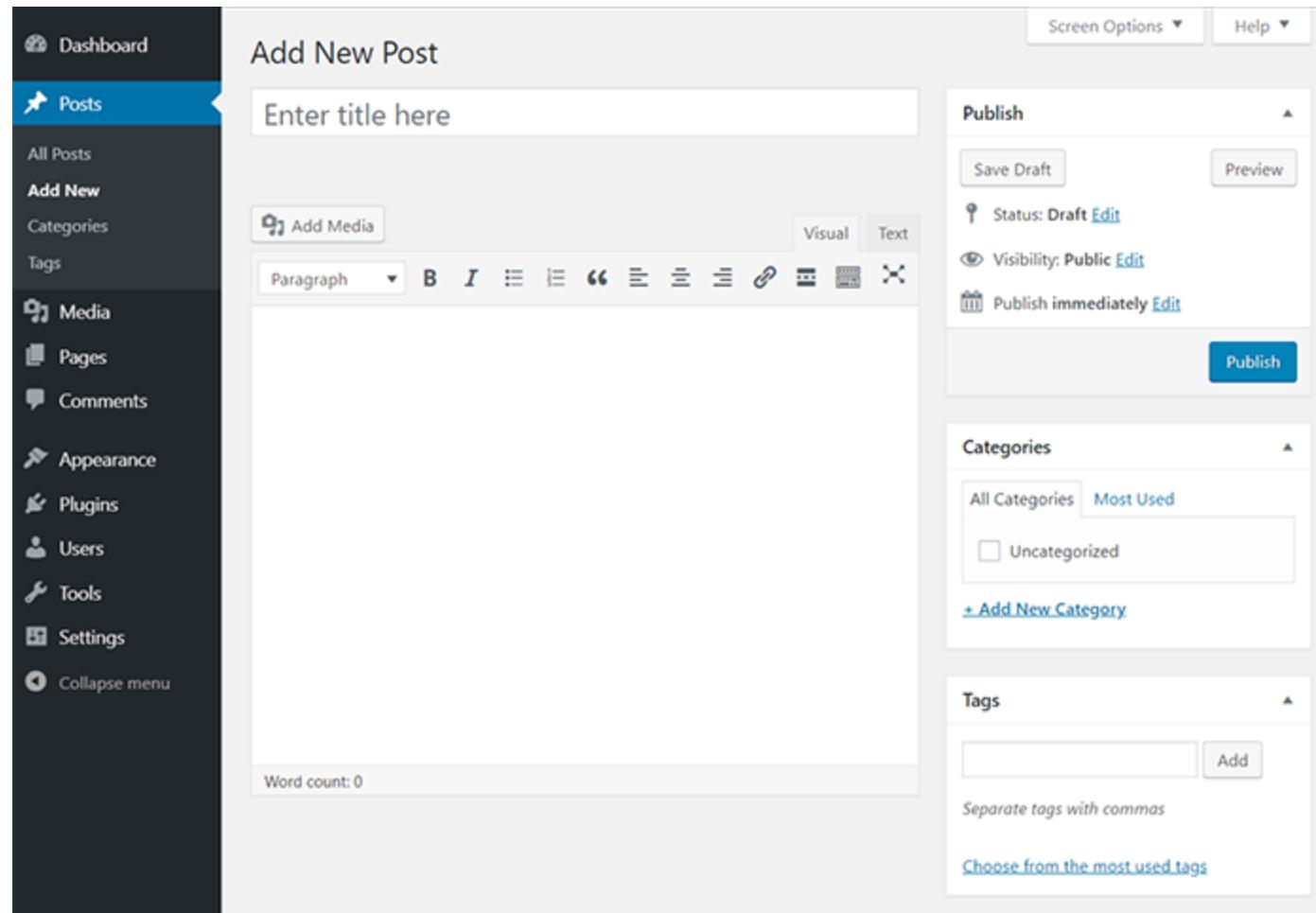
- 管理がめんどくさいけど柔軟性抜群

## コンテンツの管理をもっと楽にしたい

Jekyllでmarkdownを書くだけでサイトが作れるようになったおかげでHTML直書き時代よりはコンテンツ管理が楽になった

けど、もっと楽にしたいよね

# コンテンツの管理をもっと楽にしたい



WordPressのコンテンツ管理画面

# コンテンツの管理をもっと楽にしたい

コンテンツ管理を楽にするためのシステム、 CMS (Content Management System)

CMSでコンテンツをかんたんに管理して、サイトに表示させる

# CMSの例

386.jp の管理画面

The screenshot shows the 386.jp CMS management interface. At the top, there's a header with a logo, the text "[Works] 案件 きやりーぱみゅぱみゅの類似歌詞検索 - Japanese (Japan) ↗ 8", and navigation tabs: "Editor" (which is active), "References" (with a count of 1), and "Tags".

The main content area has several sections:

- タグ (required)**: A list of tags with a "PUBLISHED" status and an ellipsis button. It includes "[Works] タグ" and "データ分析".
- [Works] タグ**: Another list of tags with a "PUBLISHED" status and an ellipsis button. It includes "[Works] タグ" and "アプリ".
- A large dashed-line box contains a button labeled "↗ Add existing content".
- 出版物かどうか**: A section with radio buttons for "Yes" (unchecked), "No" (checked), and "Clear".
- 使用したツール**: A list of tools with a "PUBLISHED" status and an ellipsis button. It includes "[Works] ツール" and "spaCy".
- [Works] ツール**: Another list of tools with a "PUBLISHED" status and an ellipsis button. It includes "[Works] ツール" and "FastAPI".
- [Works] ツール**: A third list of tools with a "PUBLISHED" status and an ellipsis button.

# CMSの例

386.jp

News Works Biography Blogs Contact

ENGLISH

## Nearest Lyrics for KPP

きゃりーぱみゅぱみゅの楽曲を対象に、入力したキーワードに類似した歌詞を提示するアプリです。

楽曲の歌詞をフレーズごとに取得し、spaCy及びGinzaを用いて300次元のベクトルデータにした上で、コサイン類似度を用いて類似歌詞を取得しています。ユーザーがキーワードを入力し、それをベクトル化し、コサイン類似度で比較することによって、瞬時にユーザーのキーワードと最も近い歌詞を取得することができます。

Project Date

July 22, 2021

Project Area

データ分析, アプリ

### 目的・目標など

本プロジェクトは武蔵野大学 データサイエンス学部 3年次の授業、「専門コース演習II（人工知能アルゴリズムデザイン）」の成果物として制作されました。

日本の「カワイイ」文化や「原宿」文化を発信し続けてきたきゃりーぱみゅぱみゅ。彼女の独特的な世界観の歌詞と中田ヤスタカのサウンドで世界中で人気なアーティストである一方、よく意味のわからない歌詞があるのも事実。そんな不思議な歌詞を自然言語処理を使い分析することで、本来の意味を少しでも理解できるのではないかと思い、このアプリを制作しました。

### 使用したツール:

✓ spaCy  
データ分析

✓ PostgreSQL  
データ分析

✓ FastAPI  
CLI / バックエンドAPI

✓ Docker  
ITインフラ

# CMSの例

- WordPress
- Jekyllのmarkdown方式

それ以外にも:

- Contentful
- microCMS
- strapi
- Craft CMS
- Prismic CMS

# Headless CMSのなかたち

- Contentful
- microCMS
- strapi
- Craft CMS
- Prismic CMS

# Headless CMSとは

従来型のCMS (WordPressなど)はコンテンツの入稿画面・データベースに加えて表示面がセットになっていますが、それに対してヘッドレスCMSはヘッド(表示面)がなく、APIによってコンテンツを配信するのが特徴です。<sup>^2</sup>

# Headless CMSの利点

同じコンテンツを様々な場所に展開できる

- Worksページ
- ポートフォリオ用シングルページ

サイトのデザインを変更しやすい

# **SSG x Headless CMS = さいきょう**

ぼくのさいきょうのポートフォリオサイトができあがるね!!

# [余談] 386.jpの構成

v1

- Jekyll
- カスタムテーマを魔改造
- markdownでコンテンツ管理

v2

- Hugo
- 複数言語対応など、柔軟に改造
- markdownでコンテンツ管理

# [余談] 386.jpの構成

v3.0

- React.js (今回の話から外れるけど動的なサイト、SSRです)
- Headless CMSであるContentfulでコンテンツ管理
- 複数言語対応
- ポートフォリオ用シングルページもパーツとAPIの取得部分を書くだけなので短時間で構築

v3.1 (現在進行系)

- Next.js with Vercel (今度は完全にSSG)

ここまで、様々なサイトの構築方法をご紹介してきました。  
けど、もっと見た目を気にしようよ

# 独自ドメインを取得・導入しよう

独自ドメインとは

386.jp , google.com , amazon.com , musashino-u.ac.jp

# 独自ドメインの重要性

- 覚えてもらえる
- アクセスしやすい
- ブランディング (宮村 → 386)

# 独自ドメインの取得方法

レジストラーと呼ばれる会社から購入する(買い切りではなく、サブスク型)

- [お名前.com](#) (安いけどサポートが悪いので覚悟した上で使う)
- GoDaddy
- Namecheap
- CloudFlare Registerer

# [Activity] ドメインを取得してみよう

今回はGoDaddyを使います

[godaddy.com](https://www.godaddy.com)



386.jp

Search Domain

いい感じの名前を検索してみよう

## [余談] 覚えてもらえやすいドメイン

- 短い
- 名前の中の一部
- あだ名
- 特徴的ななにか

# [Activity] ドメインを取得してみよう

The screenshot shows the GoDaddy website interface. At the top, there is a search bar with the query '386.jp'. Below the search bar, a message says 'Domains include free Privacy Protection forever.' A 'Continue to Cart' button is visible. The main content area displays two domain options: '386.jp' (marked as 'DOMAIN TAKEN') and '386.life' (marked as 'GREAT ALTERNATIVE'). Both options show a price of ¥243 for the first year. Below these, sections for 'All endings' and 'More names' list various premium domain options with their prices and renewal details.

Category	Domain Name	Description	Price	Renewal Price
All endings	386.co	PREMIUM Call (480) 366-3343 for buying assistance Estimated Value	¥38,828 <sup>①</sup>	¥4,539/yr when you renew
	386.ltd	PREMIUM Call (480) 366-3343 for buying assistance Estimated Value	¥29,679 <sup>①</sup>	¥29,679/yr when you renew
	386.shop	PREMIUM Call (480) 366-3343 for buying assistance Estimated Value	¥317,343 <sup>①</sup>	¥6,102/yr when you renew
More names	486.org	PREMIUM Call (480) 366-3343 for buying assistance Estimated Value	+ ¥2,399/yr ¥1,220,432 <sup>①</sup>	
	the386.jp	Restrictions apply <sup>②</sup>	¥4,027 <sup>①</sup>	Same price next year
	386s.jp	Restrictions apply <sup>②</sup>	¥4,027 <sup>①</sup>	Same price next year

たくさんのドメインがありますね。おっと、386.jpは既に取られていたようですね。

## [Activity] ドメインを取得してみよう



386.world

¥5,125 **¥243<sup>?</sup>**

for the first year



お気に入りのドメインが見つかったらカートに追加しよう

1 item

Subtotal: **¥17,938**

## [Activity] ドメインを取得してみよう

カートに追加したら、購入に進みましょう。

Domain Protectionなどのオプションはなしで大丈夫。

クレジットカードかPayPalで支払えます

**386.world**

.WORLD Domain Registration

5 Years ▾

**¥20,743**

¥25,626

19% off

Renews March 2027 for ¥427/mo



(¥25,626 total)

170,000 times a year - That's how often  
criminals attempt to steal domains.

Protect your domain.

**Full Domain Protection**

**¥799/yr per domain**

**Get It**

Looks Good, Keep Going

**Continue Shopping**

っていう冗談はさておき...

## ドメインを取得してみよう

このような形で年間数千円でドメインが持てて、ブランディングとしての役割を果たしてくれるし、持ってて損はないものだと思うので、購入してみてはいかがでしょうか？

# じゃあ、ドメインを取得したとして、そのあとどうする？

ドメインを取得したら、まずはポートフォリオを独自ドメインにしてみよう！

の前に...

# DNSって大事

DNSとは、IPアドレスとドメインを変換する電話帳のようなものでしたよね？

ドメインを利用する場合は、DNSサーバーに接続さえできていれば変換ができますが、運営する場合は自分で「どうドメインとIPアドレスを変換するか」を定義してあげる必要がある。  
(DNSレコード)

レジストラーが運営しているDNSサーバーはDNSレコードが世界中に浸透するまで時間がかかるので、CloudFlare DNSを使うことをおすすめします。 (CloudFlareだとレコードの浸透が早い)

# CloudFlareへの登録はかんたん

ドメイン購入後、ICANNの認証(メール認証)などを済ませて、CloudFlareのアカウントを作成

サイトを追加から**無料**でドメインをCloudFlareと紐付けることができます

## Cloudflare で Web サイトをスピードアップして強固に保護

自分のサイトを入力します (example.com):

サイトを追加

---

複数のサイトを追加しますか? [追加する方法.](#)

## CloudFlareへの登録はかんたん

レジストラーの管理画面からドメインの「ネームサーバー」をCloudFlareから指定されたものに変更することでCloudFlareとの紐付けは完了!

DNSレコードの浸透が早いことで、もし操作を誤って異なるDNSレコードを登録したとしても、復旧が早い

## まずは、GitHub Pagesでホストしているサイトを独自ドメインに

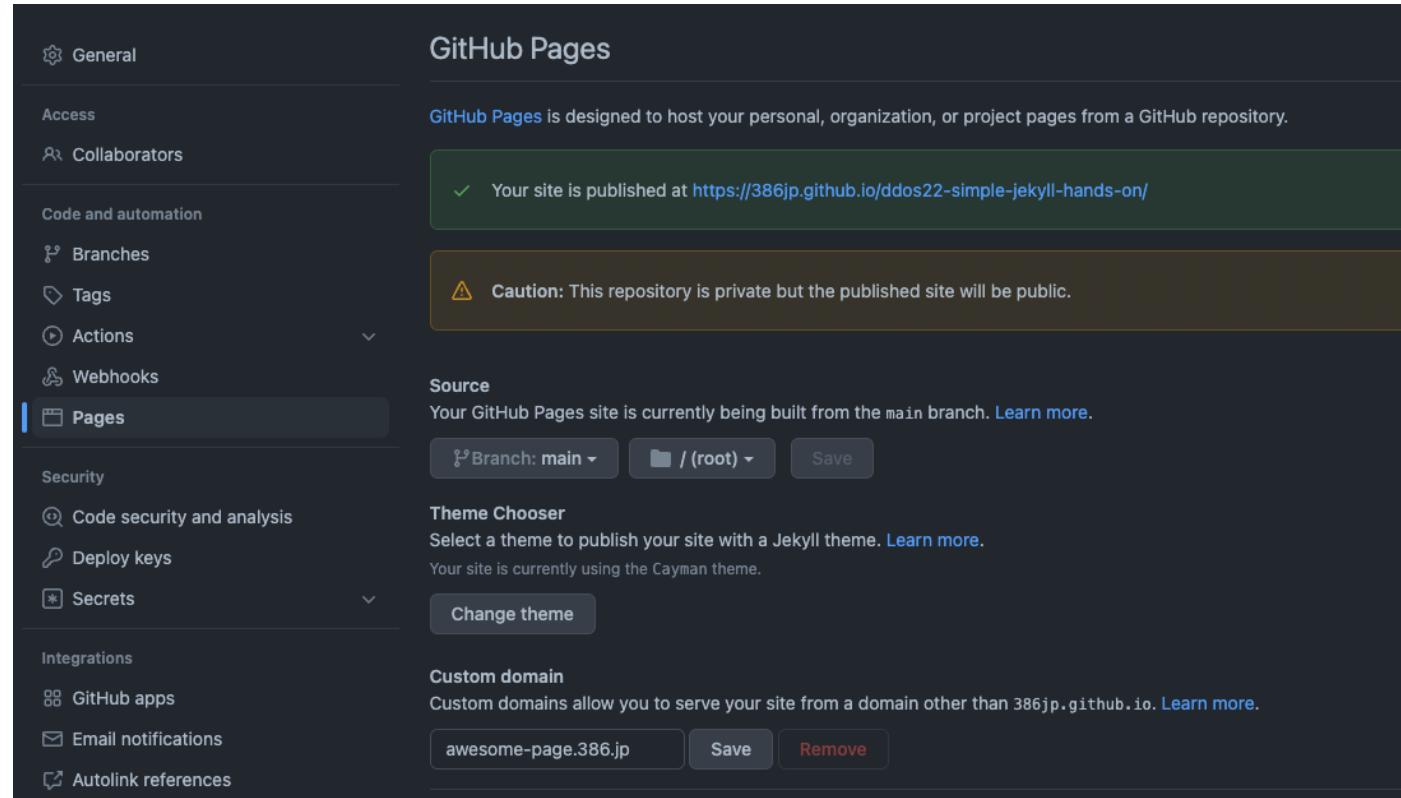
みなさんの多くがGitHub Pagesでポートフォリオサイトをホストしていると思います  
それを独自ドメインにしてみませんか？

# GitHub Pagesでホストしているサイトを独自ドメインに

まずは、ポートフォリオサイトを運用しているGitHubレポジトリにアクセス

設定 → Pagesから、GitHub Pagesの設定画面を出す

Custom domainの部分に自分のドメイン・サブドメインを入力、保存



## [余談] ドメイン・サブドメインとは

### ドメイン

- 386.jp

### サブドメイン

- www.386.jp
- muds-covid-health.386.jp
- awesome-page.386.jp

# GitHub Pagesでホストしているサイトを独自ドメインに

CloudFlareで CNAME レコードを登録

CNAMEとは、\*\* 「ドメインのエイリアス (別名)」 \*\*を設定するためのレコード

386.jp の DNS 管理

DNS レコードを検索 Advanced + レコードを追加

awesome-page.386.jp は 386jp.github.io のエイリアスであり、Cloudflare を介してプロキシしたトラフィックがあります。

タイプ	名前 (必須)	ターゲット (必須)	プロキシステータス	TTL
CNA... ▾	awesome-page	386jp.github.io	<input checked="" type="checkbox"/> プロキシ済み	自動

ルートには @ を使用します

キャンセル 保存



# GitHub Pagesでホストしているサイトを独自ドメインに

確認してみよう!



## こんにちはみなさん

DDoS、ポートフォリオ講座、いかがお過ごしでしょうか?

markdownファイルだけでポートフォリオが作れたら、サイトの管理楽になりそうじゃないですか?

---

[ddos22-simple-jekyll-hands-on](#) is maintained by [386jp](#).

This page was generated by [GitHub Pages](#).

ぼくのさいきょうのぽーとふおりおさいとのできあがり！

## もっと見てもらえるようにするために

もっとポートフォリオを手軽に見てもらえるようにするためにどうすればいいのか

→ ページ読み込み速度の高速化

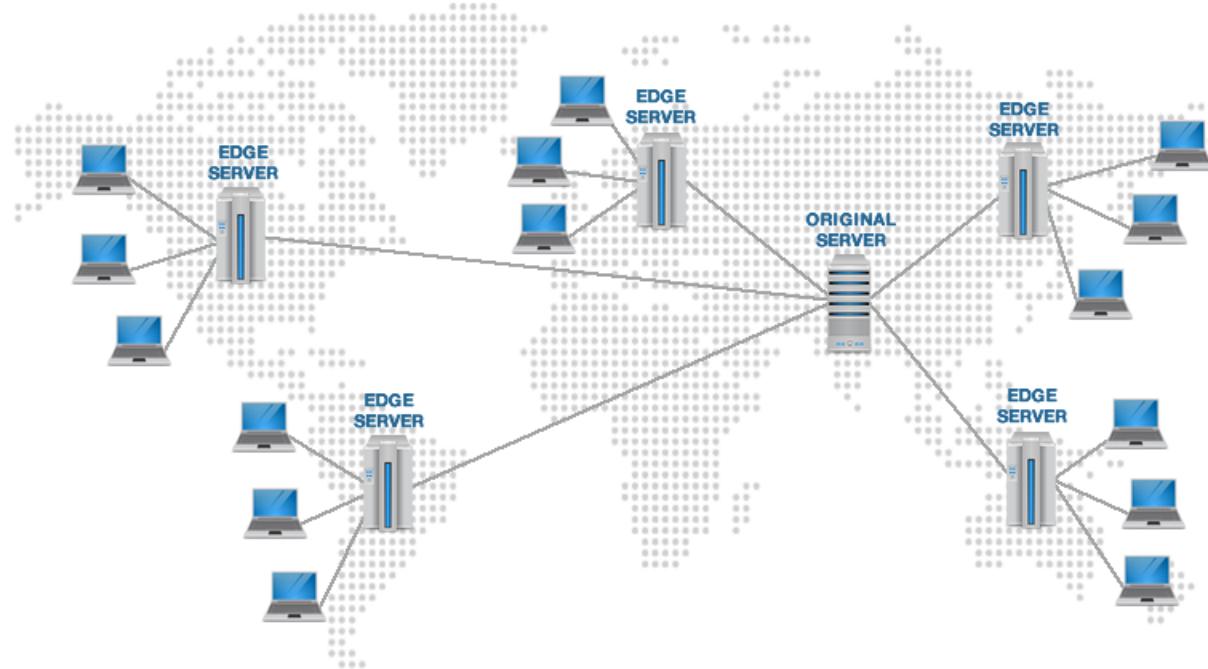
# ページ読み込み速度を高速化するには

キャッシュ (Cache)が重要

CDNってなんだ?

# CDNとは

CDN (Content Delivery Network)と呼ばれる技術で最速のルーティングを実現



オリジナルサーバー(運営が建てたサーバー)のデータの一部を世界中の様々な場所のサーバーにコピーしておき、クライアントから一番近いサーバーから、コピーされたデータを配信する→高速にデータを配信できる

## CDNとは

CDN (Content Delivery Network)と呼ばれる技術で最速のルーティングを実現

CloudFlareでこれが実現できちゃう!!

# CDNを使ってみる

CDN (Content Delivery Network)と呼ばれる技術で最速のルーティングを実現

いや、もうすでにCDN、使ってますよ

# CDNを使ってみる

386.jp の DNS 管理

DNS レコードを検索 Advanced レコードを追加

awesome-page.386.jp は 386jp.github.io のエイリアスであり、Cloudflare を介してプロキシしたトラフィックがあります。

タイプ	名前 (必須)	ターゲット (必須)	プロキシステータス	TTL
CNAME	awesome-page	386jp.github.io	プロキシ済み	自動

ルートには @ を使用します

キャンセル 保存

## CDNを使ってみる

GitHub Pages自体、GitHubのCDNが噛んでいるので高速だが、CloudFlareのCDNによって  
高速化 & さらなる機能が使える

# CloudFlareの機能

- HSTS (HTTP Strict Transport Security): ドメインがSSLを使っていることをブラウザ側に記憶させる技術が標準装備
- パブリックIPを持っていなくてもArgo Tunnelでアプリを気軽に公開
- 無料で独自ドメインのメールアドレスを持つ (受信のみ)
- サイトへのアクセス制限を気軽にかけられる
- ページのリダイレクト

などなど

CloudFlareの営業ではありませんので!!

けどほんとにいい製品なんで...

# ページ読み込み速度を高速化するには

根本的なサイトの作り方が重要

特にSSGフレームワークなどを使っていると、重要になってくる

けど客観的に「いいサイト」ってどう評価できるの?

→ Google PageSpeed Insightsを活用しよう

# Google PageSpeed Insightsとは

Google PageSpeed Insightsとは、

モバイル端末やパソコン向けのページの実際のパフォーマンスに関するレポートと、  
うしたページの改善方法を確認できる無料サービスのこと

である。

[pagespeed.web.dev](https://pagespeed.web.dev)

# Google PageSpeed Insightsを使ってみる

ちなみに、386.jpは突貫工事なのでめっちゃ悪いです笑

This URL

https://www.386.jp/

42

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. [See calculator.](#)

▲ 0–49   ■ 50–89   ● 90–100

METRICS

First Contentful Paint  
1.1 s

Speed Index  
3.1 s

Largest Contentful Paint  
3.9 s

Time to Interactive  
3.5 s

Total Blocking Time  
640 ms

Cumulative Layout Shift  
0.001

Captured at Mar 28, 2022, 1:41 PM GMT+9

Emulated Desktop with Lighthouse 9.3.0

Single page load

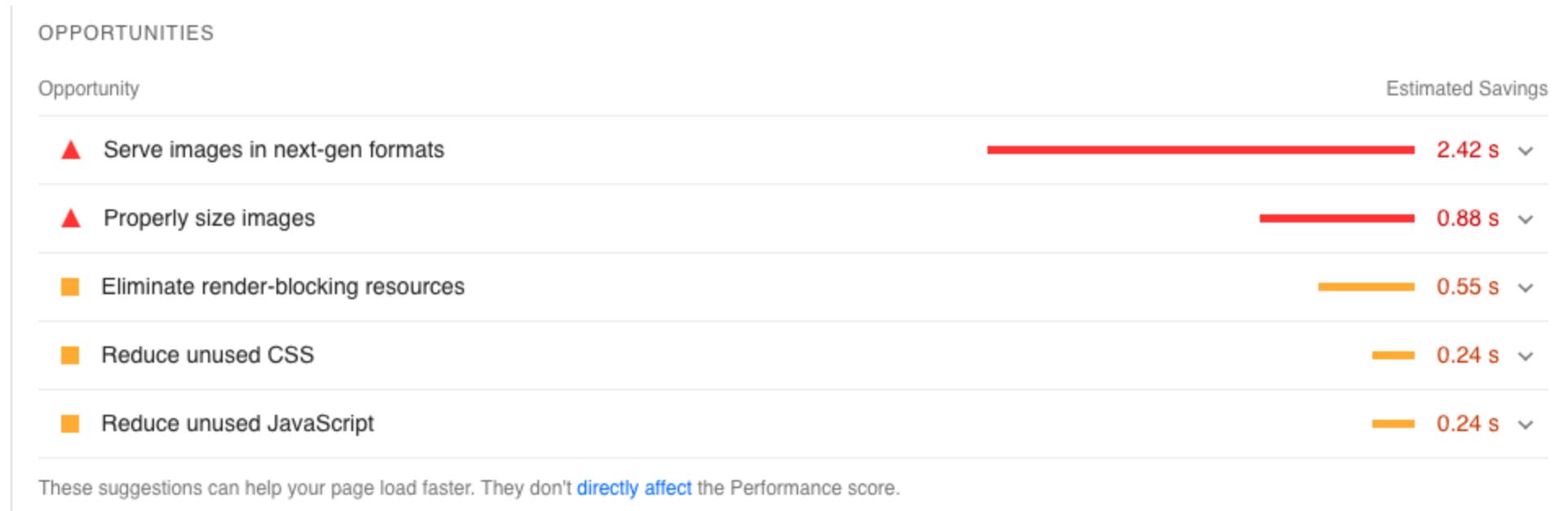
Initial page load

Custom throttling

Using HeadlessChromium 98.0.4758.102 with lr

# Google PageSpeed Insightsを使ってみる

どこがどのように悪いのか提案してくれます



# Google PageSpeed Insightsを使ってみる

このような感じで、自分のサイトの改善点を速度的な観点から教えてくれるので参考にしながら、ポートフォリオサイトをよりよくしましょう

## 人間にとての良いサイトにするには

けど、いくらページの読み込みが早くても人間が見ていいサイトじゃないと意味ないよね  
アクセス状況を分析しよう

→ Google Analyticsを活用しよう

# Google Analyticsとは

Google Analyticsとは、

Googleが提供する、Webサイトのアクセス状況を分析できるツールで、ユーザーの行動や属性、訪問したページなどを分析することができるもの

である。

専用のコードをサイトに埋め込むことで利用できる。

どれくらいの人数に見てもらえているのかも重要だが、「離脱率」と「直帰率」に注目

## 離脱率と直帰率<sup>^3</sup>

- **離脱率:** 個々のページのすべてのページビューで、そのページがセッションの最後のページになった割合
- **直帰率:** そのページから始まったすべてのセッションで、そのページがセッションに存在する唯一のページだった割合

# その他にも検索での順位を見るには...

検索における順位やパフォーマンスを見るために、Google Search Consoleを活用しよう

Webサイトの認証 (HTMLファイルの設置, TXTレコードによる認証など)だけで利用可能

サイトの概要  
過去 28 日間

Google 検索からお客様のサイトへの  
クリック数が 33 クリックに到達しました  
(+3%)

最も多く検索されたクエリ ▾

1 rocketchat nginx	平均掲載順位: 3.2	3回のクリック
2 anyenv m1	平均掲載順位: 5.7	2回のクリック
3 tkinter wsl2	平均掲載順位: 5	2回のクリック
4 rocketchat websocket	平均掲載順位: 8.1	1回のクリック
5 anyenv brew	平均掲載順位: 8.4	1回のクリック

< 1~5/7 >



The screenshot shows a summary of search queries for a website over the past 28 days. It highlights a 3% increase in clicks to 33. Below, a list of top queries is shown with their average search rank and click-through rates:

Query	Avg Rank	Clicks
rocketchat nginx	3.2	3
anyenv m1	5.7	2
tkinter wsl2	5	2
rocketchat websocket	8.1	1
anyenv brew	8.4	1

検索されているキーワードなども見ることができるので、コンテンツの参考に