

# 情報デザイン応用演習II 10. ポートフォリオ作成 I

# 目次

- 初めに
- ポートフォリオサイト
- Github Pagesとは?
  - Git
  - Github
  - Github Pages
- HP制作 ワークフロー
  - ワークフロー
  - 環境
  - やってみよう

# 初めに

## 前回の内容

- flexbox
- css grid

についておさらいしました。

## 今後の予定

### ポートフォリオ作成

を最終課題として、それに対して今回入れて6回課題に取り組んでいこうと思います。

企画から公開までを全てこなす課題となります。

# ポートフォリオサイト

## 前提条件

実力差が激しいので、

1. 静的ページで作成する
  - Github Pagesとして公開すること
  - レスポンシブ対応であること
2. Wordpressを利用する
  - FSE対応テーマでなければ小テーマを利用すること
3. ポートフォリオ作成ツールを利用する

のどれでも良いことにします。(本当は3は避けて欲しいけど...)

ツールについて

## **Git? Github? Github Pages?**

この3つの単語がよくわからないと、全くわかりませんね。

## Git

**バージョン管理システム**と呼ばれるものです。

開発においては、バグを修正したり、新機能を追加したり、ということをよくします。

これを、フォルダなどで日付管理することもできますが、システムで管理させるといいよね、というのがバージョン管理システムと呼ばれるものです。

プロジェクトを管理する場所のことを**リポジトリ**と呼んだりします。

特に、自分のコンピュータ上の管理する場所を

ローカルリポジトリ

と呼びます。



## Github

一人ではなく、共同で作業すると

- 誰が
- いつ
- どんな修正を加えたか

について管理する必要があります。

ってことは、サーバが必要ってことです。

有名なサービスだと

- Github

というのがあります。これは、自分のコンピュータ上でないので、

リモートリポジトリ

と呼ばれます

## Github Pages

サーバ上にプロジェクトを管理できるGithubですが、  
Webサーバを使えるようにすれば、Webページを作成できますよね。

これが、Githubの

Github Pages

という機能になります。

## Git/Github 紹介動画

- Gitとは？GitHubとは？10分で仕組み・使い方を解説！【超入門編】(11:18)
- エンジニア向けと言ってるけど、全員使えていいと思っています。
- 主にはテキストファイルで使うべきで、動画などは容量が大きいためあまり向いてません。

## Github Pages 紹介動画

- 【GitHubの使い方入門】 GitHub Pagesを使って世界中にサイトを公開しよう！アウトプットこそ学びだ！(10:30)

真似をして、最低限のHTMLでいいので、リモートリポジトリで公開してみましょう。

## Github vscode 連携 紹介動画

Githubでプロジェクト作成して、それをVScodeで**クローン**(ローカルに引っ張ってくる)のが一番簡単だとおもうので、その方法、及び更新方法について説明します。

- [GIT with VSCode | Clone, Commit & Push | Git Commit & Push with VSCode | GIT Clone, Commit & Push\(6:55\)](#)

git configの件は、個人を特定するだけなので、ユーザ名とか適当でいいです。メアドは有効な方がエラーなど報告があるので、使えるメアドにしましょう。(設定は一回だけです)

## 仕組みの紹介動画

- 【基本知識】 GitHubとは？できることや基礎用語を解説！ (12:19)

## Gitを使いこなしてみよう

- Githubにリポジトリ作成(まだtestとかにしておこう)
- 簡単なHTMLをアップロードする
- Github Pagesで公開
- VSCodeでローカルにクローン
- ファイルの修正・ファイルの新規追加
- コミット 同期などして、リモートリポジトリに修正を加え、最新バージョンが更新されていることを確認しよう。

## その他...

これ、HTMLなどのコーディングだけでなく、文書作成でも利用できると思います。

実際、自分はこのサイト・スライドを作るときに使いまくってます。

詳しくはメディア表現VIでも扱います。

Git,Github,Github pagesに関する動画はたくさん上がっているので、わからなくなったら確認してみよう。

このページもよくまとまっています。

- **【超簡単】GitHub Pagesとは？手軽にサイト公開してみよう！**



# HP制作 ワークフロー

## ワークフロー

通常以下のような流れとなります。

1. ヒアリング
2. 企画立案・サイト設計
3. コンテンツ制作
4. Webサイトデザイン
5. コーディング
6. 校正・品質チェック
7. 確認・チューニング
8. サイト公開

## 環境

通常では

1. 開発・テスト環境
2. ステージング環境
3. プロダクション(本番)環境

と使い分けます。

1. 個人のPC
2. 関係者がアクセスできるサーバ
3. 不特定多数がアクセスできるサーバ

ですが、今回2番目は使いません。

2のところで、テストなどを行います。

## ヒアリング

クライアントがいる場合、要望などを聞きます。

また、この場合、現状サイトや競合サイトの分析も行います。

今回は、ポートフォリオサイトとしてどんなものがあるか、分析調査してみましょう。

## 企画立案・サイト設計

分析調査を踏まえたうえで、ユーザに効果的な

- Webサイトの企画
- サイトマップ

を制作します。

## ポートフォリオサイトに必要な事項

- 自己紹介
- 作品紹介(制作の意図・プロセス・担当範囲)
- スキル
- 実績
- 連絡先

## コンテンツ制作

コンテンツに必要な

- コピー
- 文章
- 画像

などを制作します。

今回は、ポートフォリオサイトのため、これまでに制作し、掲載したいコンテンツをWebで掲載できるように準備しましょう。

例えば、psd, ai, movファイルでは掲載できません。PDF,PNG,mp4形式に変換しましょう。

## IFRAME

HTMLではIFRAMEというタグがあります。

Webページ内に別のWebページや画像、動画などのコンテンツを読み込んで表示することができます。

ってことは、前半にやったJavaScriptの作品もこれで公開できるはずです。

- [iframeとは？基本的な仕組みと使い方・使用例を解説](#)



## Webサイトデザイン

情報を整理し、ユーザビリティを考慮したWebページをデザインします。

一から制作するのは大変ですので、XD,Figmaのテンプレートを利用してもOKです。レスポンシブ対応なことにも留意しましょう。

- [FREE Responsive Portfolio Template -- Behance](#)

こんなの使うと作りやすいかもです。自分でも探してみましょう。

- トップページ
- 各作品ページ

の二つがあれば、最低限足りると思います。(他に自己紹介ページかな...)

後述する**モバイルファースト**も意識しましょう。

## コーディング

最近では**モバイルファースト**でレスポンス対応することが多いようです。(Googleさんの意向も大きいようですが)

- [モバイルファーストデザインとはスマホ版から作ればOK？手順やCSS解説つき](#)

```
@media (min-width:768px){  
  /*タブレット用として画面幅を768pxまでに設定*/  
}  
@media (min-width:1024px){  
  /*PC用として画面幅を1024pxまでに設定*/  
}
```

で追記していきましょう。最悪、モバイルとPC版でOKとします。

- [参考：レスポンス対応のレイアウトを実装する最新テクニックを解説、モバイルファーストとデスクトップファーストの現状](#)

## ハンバーガーメニュー

レスポンス対応時にほとんどは、レイアウト変更で対応できるかと思いますが、それに対応できないのがハンバーガーメニューとなります。

この辺参考にしてみましょう。

- **【コピペで簡単】ハンバーガーメニューをCSSだけで作る方法**

## 校正・品質チェック

様々な環境で表示エラーや動作の不具合がないかチェックしましょう。

- ResponsiveViewer(Chrome機能拡張)

等をうまく使っていきましょう。

## 確認・チューニング

クライアントにチェックをうけ、直していきます。

今回は、なくてもいいですかね。

## サイト公開

Githubで公開できるようにして終了です。

## 就活で利用する場合

ソースを見れば、採用担当者はHTMLやJavaScriptの技術がどのくらいするか判断できることを意味します。

```
<!--  
    HTMLのコメントアウト  
-->  
// JavaScriptの1行のコメントアウト  
/*  
    JavaScriptの複数行のコメントアウト  
*/
```

等も気にすると、評価する人にアピールもできますね。

## ワークフロー(再掲)

1. ヒアリング
2. 企画立案・サイト設計
3. コンテンツ制作
4. Webサイトデザイン
5. コーディング
6. 校正・品質チェック
7. 確認・チューニング
8. サイト公開



## やってみよう

それでは、まず1-3あたりをやっていきましょう。

- 掲載する作品集め
- ページ構成

とよく考えましょう。

今日を入れて、残り6回となります。

トラブったらどんどん質問してください。