

Storyboardって何だ？

~UIカタログのススメ~

自己紹介

名前：相川大吾（あいかわだいご）

やってること：フロントエンドエンジニア

よく触る技術：Vue / Nuxt、React / Next

趣味：筋トレ、スパルタンレース、野球観戦



 **HR forecaster** by **doda**

 **Salaries** by **doda**

PERSOL MIRAI Z



パーソルキャリアについて

**GIVE PEOPLE
THE POWER
TO OWN THEIR
WORK-LIFE.**

人々に「はたらく」を自分のものにする力を

パーソルキャリアについて

創業年



1989年

従業員数



5,756名

※有期社員含む
グループ会社は内中のみを数え、2023年3月1日時点

プロダクトづくりに関わるスペシャリスト



741名

※特選職域、2023年4月1日時点

グループ売上高(億円)の推移

2018年 9,258

2021年 10,608

2019年 9,705

2022年

2020年 9,507

12,239

0 → 1
フェーズ



新規事業開発
プログラム



実践型能力開発研修
プログラム

1 → 10
フェーズ

Salaries

HiPRO Direct

HR Spanner
by doda

HR forecaster
by doda

Hranalyst
by doda

エクスチーム

10 → 100
フェーズ

doda

doda X

HiPRO Biz

HiPRO Tech

BPS

CREDENCE
by doda

本日の流れ

1. 当社紹介
2. Storybookってなんだ？
 - a. コンポーネント指向の開発
 - b. Storybookで何ができる？
 - c. Storybookのメリットデメリット
3. 動かしてみよう（React.js）
4. まとめ

本日の資料について

- 本日のスライドと、ハンズオンの完成品は以下のリポジトリに格納してあります
 - <https://github.com/DaigoAikawa/storybook-tutorial>

Storybookってなんだ？

Storybookってなんだ？

UIカタログを作ることができるツール

- UIコンポーネントをアプリケーションとは切り離して開発、検証が行える
- UIコンポーネントのドキュメンテーションも行える
- React.js、Vue.js、SvelteなどメジャーなJSフレームワーク、ライブラリに対応

[公式ページ](#)

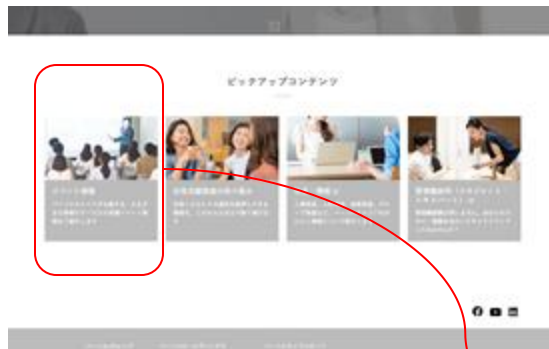


UIコンポーネント？ 🤔

UIコンポーネント

画面表示を特定の機能やパーツごとに分けたものをUIコンポーネントと呼ぶ

当社HPを例に考えてみる



```
<div>
  
  <div>
    <p>タイトル</p>
    <p>内容</p>
  </div>
</div>
```

EventCard
component

UIコンポーネント

使い回すUIはコンポーネント化することで記述量を減らせる

```
<div>
  <EventCard
    画像="img"
    タイトル="hoge"
    内容="fuga"
  />
  <EventCard
    画像="img"
    タイトル="hoge"
    内容="fuga"
  />
  <EventCard
    画像="img"
    タイトル="hoge"
    内容="fuga"
  />
  <EventCard
    画像="img"
    タイトル="hoge"
    内容="fuga"
  />
</div>
```



イベント情報

パーソルキャリアが主催する、さまざまな事業やサービスの各種イベント情報をご紹介します



女性活躍推進の取り組み

女性一人ひとりの置けるを後押しできる環境を、これからも全力で創り続けます



制度・環境

人事制度、人材育成、表彰制度、グループ推進など、パーソルキャリアめはたらく環境について紹介します



管理職採用（マネジメント・エキスパート）

管理職経験は問いません。あなたのスキル・経験を活かしてキャリアアップしてみませんか？

UIコンポーネント

- 使い回すUIはコンポーネント化することで記述量を減らせる
- 可読性も上がる



イベント情報

パートタイムキャリアが主眼する、さまざまな事業やサービスの各種イベント情報をご紹介します



女性活躍推進の取り組み

女性一人ひとりの活躍を支援できる環境も、これからは全力で創り続けます



制度・環境の

人事制度、人材育成、表彰制度、グループ推進など、パートタイムキャリアのほたらく環境について紹介します



管理職採用（マネジメント・エキスパート）の

管理職経験は問いません。あなたのスキル・経験を活かしてキャリアアップしてみませんか？

```
<div>
  <div>
    
    <div>
      <p>タイトル1</p>
      <p>内容1</p>
    </div>
  </div>
  <div>
    
    <div>
      <p>タイトル2</p>
      <p>内容2</p>
    </div>
  </div>
  <div>
    
    <div>
      <p>タイトル3</p>
      <p>内容3</p>
    </div>
  </div>
  <div>
    
    <div>
      <p>タイトル4</p>
      <p>内容4</p>
    </div>
  </div>
</div>
```

```
<div>
  <div>
    
    <div>
      <p>タイトル1</p>
      <p>内容1</p>
    </div>
  </div>
  <div>
    
    <div>
      <p>タイトル2</p>
      <p>内容2</p>
    </div>
  </div>
  <div>
    
    <div>
      <p>タイトル3</p>
      <p>内容3</p>
    </div>
  </div>
  <div>
    
    <div>
      <p>タイトル4</p>
      <p>内容4</p>
    </div>
  </div>
</div>
```

UIコンポーネント

メジャーなフロントエンドフレームワークでは、HTMLに加えて、スタイル（CSS）とロジック（JavaScript/TypeScript）をまとめて一つのコンポーネントとする

ex) コピーボタンコンポーネント

- `<button>`などからなるHTML
- スタイル
- クリップボードへコピーする処理

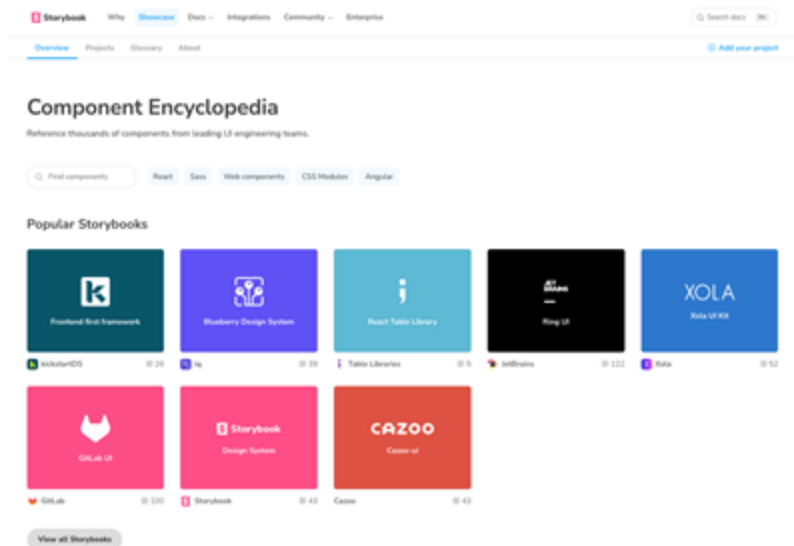


Storybookの話に戻ります 🙋

Storybookで何ができる？

Storybookの公式HPから公開されているStorybookをみることで
できるので試してみる

<https://storybook.js.org/showcase>



Storyboardで何ができる？

- 実際の画面に配置しなくてもコンポーネントのみで実装ができる
- アプリケーション上だと再現しにくいコンポーネントの状態も再現できる
- コンポーネントのドキュメントとして活用できる
- デザイナーが工数の低いデザインを考えることができる

Storybook導入のメリット

- 実際の画面に配置しなくてもコンポーネントのみで実装ができる
 - バックエンドの実装を待たずに開発ができる
 - 画面に依存した実装を防ぎやすい
 - 使い回しのしやすいコンポーネントを作成しやすい
- アプリケーション上だと再現しにくいコンポーネントの状態も再現できる
 - テキストが長くなった時の挙動は？
 - データがない時の挙動は？
- コンポーネントのドキュメントとして活用できる
 - どんなデータを渡せるのか、どんな動きをするのか確認しやすい
- デザイナーが工数の低いデザインを考えることができる
 - storybook上で実現できるUIは工数が低く実装できる

Storybook導入のデメリット

- Storybookでコンポーネントを表示するためには、コンポーネントの他にstoriesファイルを作成する必要がある
- storyと呼ばれる、コンポーネントの状態のプリセットを適切に考える必要がある
- 極めようとするすると学習コストがかかる

導入する場合、どこまでこだわるのか、何を求めて導入するのかをよく検討して導入するべき

Storybookを立ち上げてみよう 🧑🏻

ハンズオンにあたって

- 今回の例ではReact/Next.jsを用いてチュートリアルをします
- 時間の都合上React/Next.jsの記法については割愛します
- Storybookの画面を起動でき、サンプルコードが動作していることを確認します
- 時間があれば簡単なコンポーネントを作ってみます

準備(1)

1. エディタとコマンドを打てるツールを起動してください
 - a. エディタ: VS Code
 - b. コマンドツール: ターミナル、パワーシェル
2. Node.jsをインストールする
 - a. Node.jsが使えるかどうかコマンドツールで確認
 - i. `node --version`
 - ii. `npm --version`
 - b. つかえなかったらインストール
 - i. <https://nodejs.org/en>
3. npx のインストール
 - a. npx が使えるかどうか確認
 - i. `npx --version`
 - b. 使えなかったらインストール
 - i. `npm i -g npx@latest`

準備(2)

4. Reactアプリの雛形作成

- a. `npx create-next-app@latest storybook-tutorial`

5. 念の為、依存パッケージをインストール

- a. `npm install`

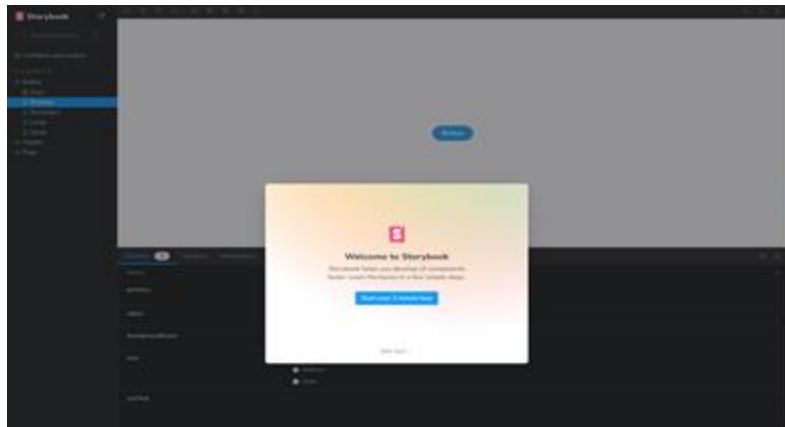
Next.jsの起動確認

1. アプリケーションを起動してみる
 - a. `npm run dev`
 - b. <http://localhost:3000/> (3000番ポートが使用されていた場合、別のポートの可能性あり)
2. 不要な初期設定の削除
 - a. `src/app/global.tsx`の4行目以降を削除
 - i. サンプルページ表示のためのスタイルが邪魔なため



Storybookを起動してみる

1. Storybook を起動
 - a. `npx sb init`
 - b. 何か聞かれたら全てyes
2. 起動完了 🎉
3. 次回以降のStorybookの起動方法
 - a. `npm run storybook`



コンポーネントを追加してみよう

- ここからは実装しながらお伝えします
- コードを確認したい方は、以下のリポジトリを参照してください
 - <https://github.com/DaigoAikawa/storybook-tutorial>

まとめ

- StorybookはUIカタログが作れる
- UIカタログがあると、いろんなメリット
 - アプリに組み込む前にコンポーネント単体で実装できる
 - 様々な状態を簡単にチェックできる
 - コンポーネントの単体テストの役割
 - デザイナーとの連携
 - etc
- storiesファイルを作成することでカタログ化可能

ありがとうございました🥰