

Nuxt.js移行プロジェクトの話

Taishi Inoue
note engineer meetup #1

Who am I

Taishi Inoue / @tic40

2018/06～ piece of cake, inc

[note.mu](#)の

フロントエンドリプレイスを担当



In progress



 **note** frontend system

Background

noteのフロントエンドをNuxt.jsへ刷新します



こんぴゅ

2018/07/25 15:01

note.mu/konpyu/n/n9b7bf4343514

Agenda

プロジェクト開始から今日までの取り組み/TIPSを紹介

- ・フロントエンドのキャッチアップ
- ・コードの秩序を保つ
- ・コンポーネント設計方針を決める
- ・SSR起因のエラーを解消する
- ・コンポーネントを管理する
- ・パフォーマンス向上への取り組み

> フロントエンドのキャッチアップ

- ・コードの秩序を保つ
- ・コンポーネント設計方針を決める
- ・SSR起因のエラーを解消する
- ・コンポーネントを管理する
- ・パフォーマンス向上への取り組み

チーム体制

- ・エンジニア3名(リモート2、オフィス1)
- ・UI周りの調整には都度デザイナーも加わる
- ・Vue.js、Nuxt.jsの社内知見は少ない。

フロントエンドキャッチアップの必要性

フロントエンドのキャッチアップ

- ・社内ハンズオンの開催

es2015復習-Vue.js入門-Nuxt.js入門ハンズオンを社内開催

- ・社外交流

社外から知見のある人物を招いて情報交換、レビュー

- ・知見の共有

得られた知見は社内wikiへ集約

- ・フロントエンドのキャッチアップ

> コードの秩序を保つ

- ・コンポーネント設計方針を決める
- ・SSR起因のエラーを解消する
- ・コンポーネントを管理する
- ・パフォーマンス向上への取り組み

コードの秩序を保つ

開始当初はVue.jsのスタイルガイドに沿っていないコードが散見されていた。

```
<template>
  <ul>
    <li v-for="item in items">
      {{ item }}
    </li>
  </ul>
</template>
```

← v-forの要素に対して v-bind:key
が指定されていない。

*ref: jp.vuejs.org/v2/style-guide/

コードの秩序を保つ

- ESLintに`vue/recommended`ルールを適用
- CIで自動化、Vue.jsスタイルガイド違反のコードを撲滅

.eslintrc.js

```
extends: [  
  'plugin:vue/recommended',  
],
```

```
4:7  error  Elements in iteration expect to have 'v-bind:key' directives  vue/require-v-for-key
```

```
* 1 problem (1 error, 0 warnings)
```

- ・フロントエンドのキャッチアップ
- ・コードの秩序を保つ

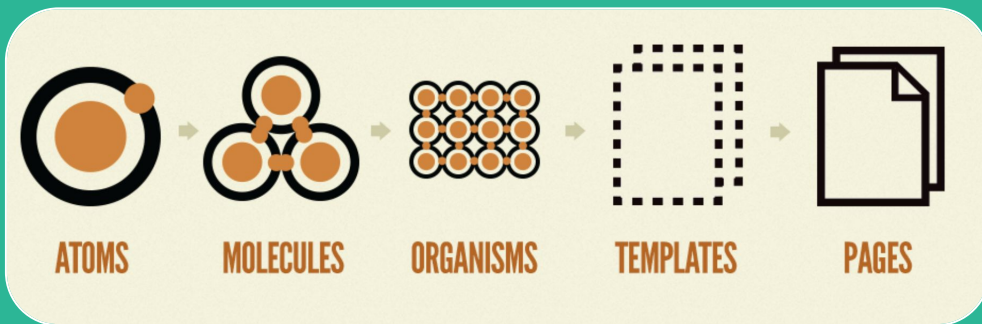
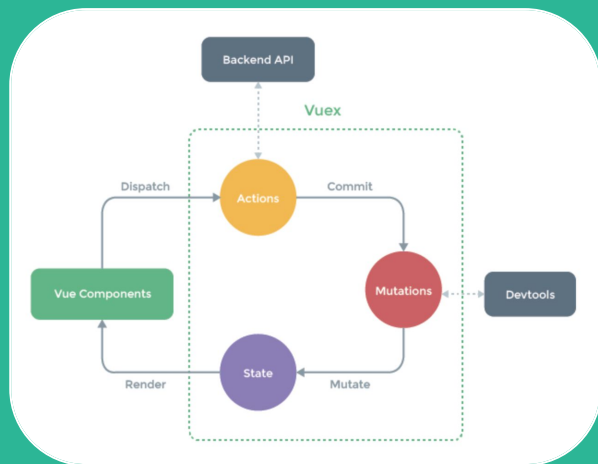
>コンポーネント設計方針を決める

- ・SSR起因のエラーを解消する
- ・コンポーネントを管理する
- ・パフォーマンス向上への取り組み

コンポーネント設計

状態管理にVuex

コンポーネントデザインにAtomic Designを採用



コンポーネント設計の揺らぎ

デザインパターンを取り入れたとはいえ、実装者によって設計に差があった。

- ・単一コンポーネントの再利用性と責務
- ・atom vs molecule、molecule vs organism
- ・状態管理(vuex state/コンポーネント内data/\$emit)使い分け

設計の揺らぎをなくす

揺らぎがある部分は明確にガイドライン化

- ・単一コンポーネントの再利用性と責務

再利用性のために責務を増やさない。責務が増える場合はコンポーネントを分割する

- ・atom vs molecule、molecule vs organism

atomは他のコンポーネントを含まない、stateless、vuexを参照しない...
等々

- ・フロントエンドのキャッチアップ
- ・コードの秩序を保つ
- ・コンポーネント設計方針を決める

> SSR起因のエラーを解消する

- ・コンポーネントを管理する
- ・パフォーマンス向上への取り組み

SSR起因のエラー

コードをそのまま移行するとSSR(server-side-rendering)起因のエラーが多発してしまった

- ・window is not defined

SSR時には、window関数をはじめクライアントサイドのリソースにはアクセスできない。

- ・cookieの参照

これも上記と同じくSSR時に参照できないので嵌った。

エラーログの収集

sentry-moduleプラグイン

github.com/nuxt-community/sentry-module

slack連携してエラーが起きたら通知。クライアントサイドで予想外なことが起こっていないかチェック



Sentry APP 4:02 PM

ReferenceError: location is not defined

Culprit

index.vue?5bcf in VueComponent.created

Project

note-front

- ・フロントエンドのキャッチアップ
- ・コードの秩序を保つ
- ・コンポーネント設計方針を決める
- ・SSR起因のエラーを解消する

> コンポーネントを管理する

- ・パフォーマンス向上への取り組み

コンポーネント把握できない問題

← 再利用可能なコンポーネントが増え、もはや把握ができなくなってしまった開発者



コンポーネントカタログの導入

Storybook: github.com/storybooks/storybook

- ・運用コストはかかるが、
コンポーネントが把握できなくなることによる弊害 > 運用コスト

*Nuxt v2で Storybook v3.xが動かなくなる問題があったが、現在はStorybook v4.0rc バージョンを使うことで回避

- ・フロントエンドのキャッチアップ
- ・コードの秩序を保つ
- ・コンポーネント設計方針を決める
- ・SSR起因のエラーを解消する
- ・コンポーネントを管理する

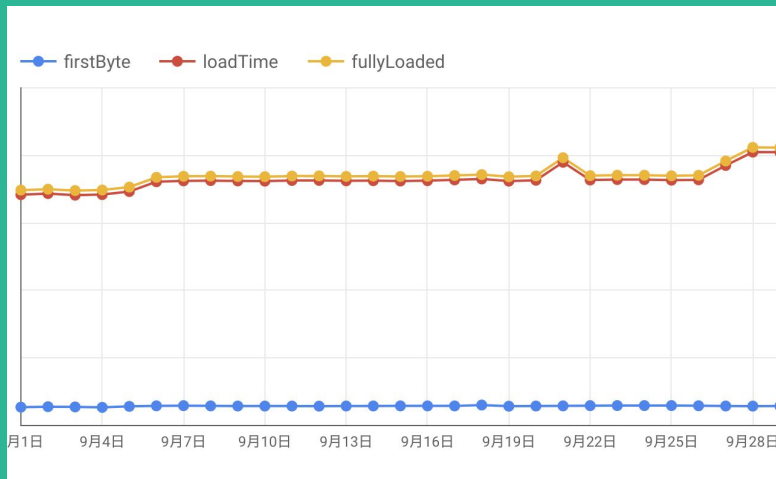
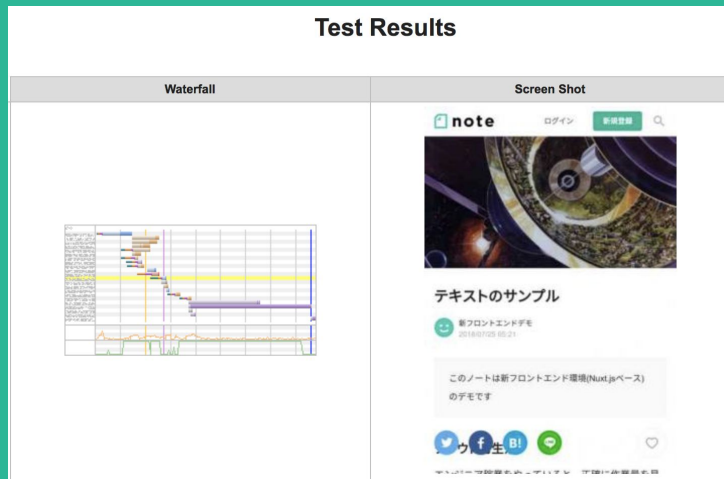
> パフォーマンス向上への取り組み

パフォーマンス計測

gas-webpagetest: github.com/uknmr/gas-webpagetest

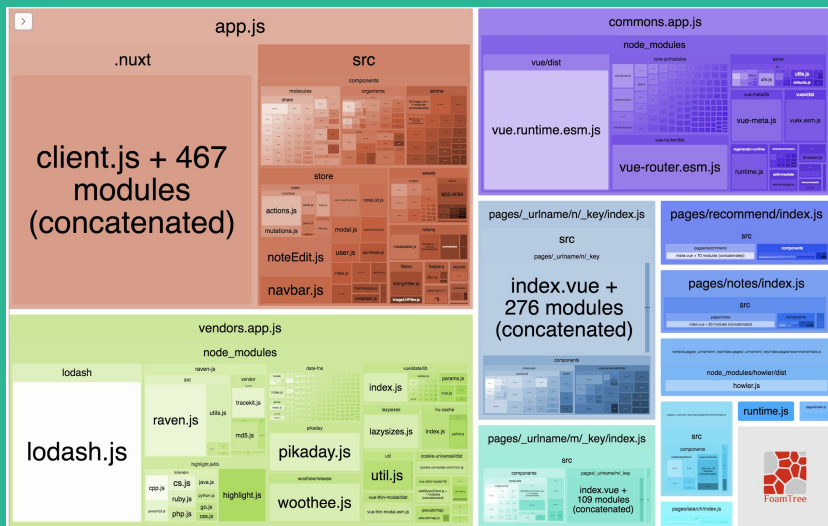
webpagetestで定期的に自動計測 > data studioでログの可視化

*SpeedCurveも検討(将来的には導入したい)



bundleファイル分析

- ・webpack-bundle-analyzerを活用
- ・モジュール単位のファイルサイズを可視化。ファイルサイズの大きいものから最適化



まだまだあります高速化施策

パフォーマンス向上は地道な取り組み

- ・画像サイズの最適化
- ・リソースの遅延ロード
- ・リクエスト数を減らす
- ・PWA対応
- ・APIパフォーマンスの向上

高速なnoteを目指して、

継続してチューニングしていきます

最後に

リリースノート公開中

リリースノート

Nuxt移行プロジェクト
2018年10月前半

ナビバーの追加実装など Nuxt移行プロジェクト リリースノート (2018年10月前半)

リリースノート

Nuxt移行プロジェクト
2018年9月後半

「ノート一覧」ページをテストリリース Nuxt移行プロジェクト リリースノート (2018年9月後半)

note.mu/noteeng/m/me7637ba82821

Vue Fes Japan@11/3



株式会社ピースオブケイク / エンジニア

福井 烈



青森県弘前市生まれの Web 系エンジニア。2児の父。

note のフロントエンドを Nuxt.js で再構築した話

弊社サービス note (<https://note.mu/>) は、Ruby on Rails と Angular.js 1系で構築された CSR のみで動作している SPA ですが、特に低スペックなスマートフォンや PC での初期表示が遅いことが大きな課題点の 1つでした。

<https://vuefes.jp/>

ありがとうございました