**Framework CSS**

Conteúdo:

* [Resumo](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#summary)
  + [Questão](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#issue)
  + [Decisão](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#decision)
  + [Status](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#status)
* [Detalhes](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#details)
  + [Suposições](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#assumptions)
  + [Restrições](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#constraints)
  + [Posições](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#positions)
  + [Argumento](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#argument)
  + [Implicações](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#implications)
* [Relacionados](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#related)
  + [Decisões relacionadas](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#related-decisions)
  + [Requisitos relacionados](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#related-requirements)
  + [Artefatos relacionados](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#related-artifacts)
  + [Princípios relacionados](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#related-principles)
* [Notas](https://github.com/joelparkerhenderson/architecture_decision_record/blob/master/examples/css-framework.md#notes)

**Resumo**

**Questão**

Queremos usar uma estrutura CSS para criar nossos aplicativos da web:

* Queremos que a experiência do usuário seja rápida e confiável, em todos os navegadores e tamanhos de tela populares.
* Queremos uma iteração rápida no design, layout, UI / UX, etc.
* Queremos aplicativos responsivos, especialmente para telas menores, como em dispositivos móveis, telas maiores, como em widescreen 4K, e telas dinâmicas, como telas giratórias.

**Decisão**

Decidiu sobre Bulma.

**Status**

Decidiu sobre Bulma. Aberto a novas opções de estrutura CSS assim que chegarem.

**Detalhes**

**Suposições**

Queremos criar aplicativos da web que sejam modernos, rápidos, confiáveis, responsivos etc.

Os aplicativos da web modernos típicos estão reduzindo / eliminando o uso de jQuery por vários motivos:

* JavaScript moderno em fases de muitos recursos que jQuery forneceu, então jQuery é menos necessário, e há módulos melhores / mais rápidos / menores que fornecem implementações específicas
* A abordagem ampla da jQuery é fazer manipulação direta de DOM, que é um anti-padrão para estruturas JavaScript modernas (por exemplo, React, Vue, Svelte)
* jQuery interfere em si mesmo se for carregado duas vezes, etc.

**Restrições**

Se escolhermos um framework CSS que usa jQuery, ficaremos paralisados ​​importando jQuery. Por exemplo, Semantic UI usa jQuery, e Tachyons não.

Se escolhermos uma estrutura CSS mínima, renunciaremos aos componentes da estrutura que podemos desejar agora ou em breve. Por exemplo, Semantic UI fornece um carrossel de imagens, e o Tachyons não.

**Posições**

Consideramos o uso de nenhuma estrutura. Isso ainda parece viável, especialmente porque a grade CSS fornece muito do que precisamos para nosso projeto.

Consideramos muitos frameworks CSS usando uma triagem de lista curta rápida: Bootstrap, Bulma, Foundation, Materialize, Semantic UI, Tachyons, etc. Nossas duas seleções para uma revisão mais profunda são Semantic UI (porque tem a abordagem mais semântica) e Bulma (porque tem a abordagem mais leve que fornece os componentes que desejamos agora).

Nós consideramos a IU semântica. Isso fornece muitos componentes, incluindo aqueles que desejamos para o nosso projeto: guias, grades, botões, etc. Fizemos um piloto com a IU semântica de duas maneiras: usando arquivos CDN típicos e usando repositórios NPM. Obtivemos sucesso com Semantic UI em uma página HTML estática, mas não tivemos sucesso em nossa caixa de tempo para construir um SPA de JavaScript (principalmente devido a problemas de carregamento do jQuery). Descobrimos que outros programadores têm pedido aos desenvolvedores da Semantic UI para criar uma versão livre de jQuery, pelos mesmos motivos que nós. Outros programadores têm solicitado uma versão livre de jQuery por muitos anos, mas os desenvolvedores disseram não e declararam que qualquer versão livre de jQuery seria muito difícil de escrever, por exemplo ~ "o projeto Semantic UI tem mais de 22.000 pontos de contato que usam jQuery "

Exemplo com semântica:

< Div class = " top ui anexado menu de tabular " >

< a class =" item de " > Alpha </ a >

< a class =" item de " > Bravo </ a >

</ div >

Nós consideramos Bulma. Bulma tem muitos recursos semelhantes à UI Semântica, embora não tantos componentes sofisticados. Bulma é construído com técnicas modernas, como nenhum jQuery. Bulma tem alguns componentes de terceiros, alguns dos quais podemos querer usar.

Exemplo com Bulma:

< Div class = " tabs " >

< ul >

< li > < uma > Alpha </ a > </ li >

< li > < uma > Bravo </ a > </ li >

</ ul >

</ div >

**Argumento**

Como acima.

Especificamente, a IU semântica parece ter um sinalizador de cautela tanto em termos de tecnologia (ou seja, tantos pontos de contato do jQuery) e também em termos de liderança (ou seja, jQuery-free era um não difícil, em vez de tentar um roteiro, melhoria contínua ou doação arrecadação de fundos, etc.).

**Implicações**

Se encontrarmos uma boa estrutura CSS não jQuery, isso geralmente é útil e bom no geral.

**Relacionados**

**Decisões relacionadas**

A estrutura CSS que escolhemos pode afetar a testabilidade.

**Requisitos relacionados**

Queremos enviar um aplicativo puramente moderno rapidamente.

Não queremos perder tempo trabalhando em estruturas mais antigas (especialmente IU semântica) usando dependências mais antigas (especialmente jQuery).

**Artefatos relacionados**

Afeta todo o HTML típico que usará o CSS.

**Princípios relacionados**

Facilmente reversível.

Necessito de velocidade.

**Notas**

Quaisquer notas aqui.