Atomic Design As A Migration Strategy

Harrison Harnisch (@hjharnis)
Harrison Harnisch

Senior Software Engineer @ Buffer

Twitter: @hjharnis
Email: harrison@buffer.com
You Should Never Throw Away Your Codebase And Start Over
Things You Should Never Do, Part I
Never rewrite gets less true over time
The Mythical Rewrite
There's a *third* option
Atomic Design
Build complexity by stitching simple things together
Atom » Molecule » Organism
Molecule

Search Component

```html
<div>
  <div>
    <label for="text-input">Search This Site</label>
  </div>
  <input type="text" placeholder="Enter Keyword" id="text-input"/>
  <button>Search</button>
</div>
```

Search This Site

Enter Keyword  Search
<table>
<thead>
<tr>
<th>Organism</th>
</tr>
</thead>
<tbody>
<tr>
<td>[List of organisms]</td>
</tr>
<tr>
<td>[Additional information]</td>
</tr>
</tbody>
</table>
Buffer Migration Process
Create Atom Library
Create Molecule Library
Migrate Atom In Existing App

- Pick something small like Text
- Build Text in the Atom Library
- Replace one Text element with the new Atom

`<Text bold>Some Bold Text</Text>`

Some Bold Text
Build A New Feature As A Molecule
What is a Product Designer? An awesome story by @jgadapee over on Medium! http://buff.ly/1LTbUqv
Some Assembly Required
Approach 1: IFrames
Approach 2: Inline Styles + Unset CSS
Build New Application Shell
Benefits And Lessons Learned
Level Up Developers
Inline Styles
Work Partitioning
Set Milestones
What does development look like?
Build in the most productive environment
Build For A Changing Future

- Should be easy to rebuild something if needed (modular)
- Rewrite while continuously shipping features
- Dev environment should be most productive environment