



Atomic Design

Co-op project
Tracy Chow

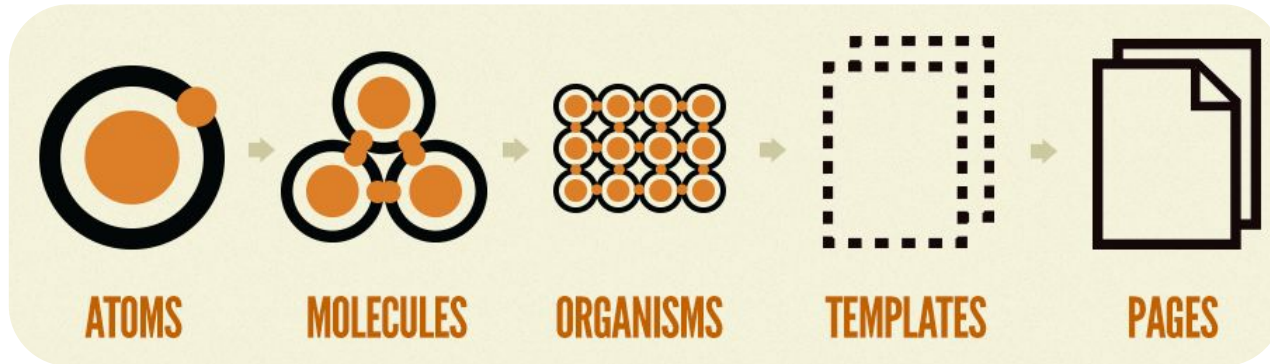


Basic Overview

- Atomic design is a way for developers to **break down interfaces and ui components into smaller units**
- It allows for a clearer mental model, and a more organized design library
- Built for easy reusability

Breakdown

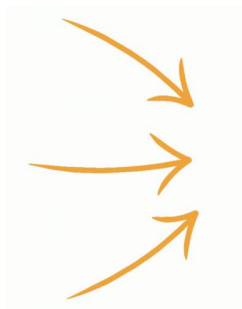
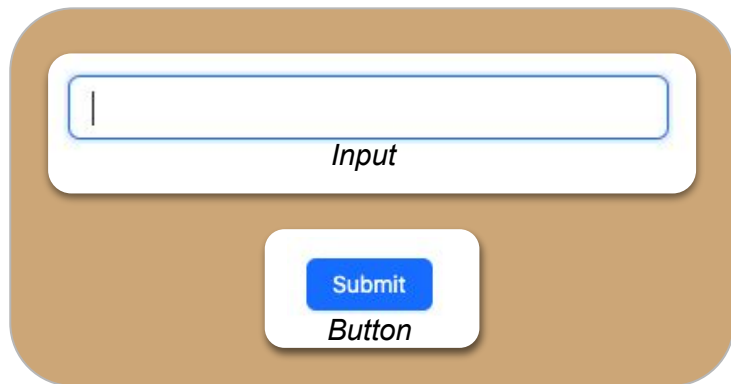
- Largely based off the natural world, it mimics how all organisms are built from the same set of molecular atoms
- In terms of the atomic design methodology, the main 'groups' considered are:



Atoms

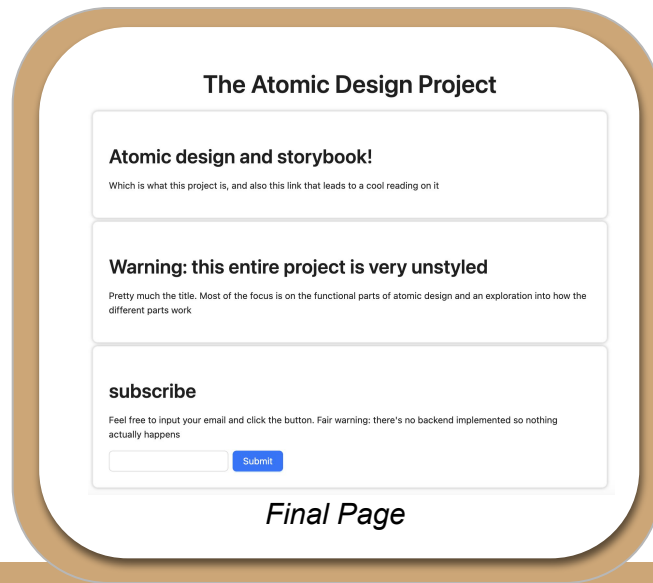
Definition

- The **smallest unit** of the mental model
- Can't be broken down anymore without losing functionality
- Usually basic HTML elements that are differentiated by unique properties
- Ex: inputs, buttons, text fields



Creating a blog page

- Let's say we're creating a blog page with posts and a subscribe form
- To follow along, feel free to pull the example repo:
<https://github.com/tc2780/atomic-design-project/tree/main>
- Atoms include: buttons, titles, inputs, etc...



Molecules

Definition

- Made by **combining atoms** together and usually need to be part of an *organism* to have value
- Atoms and molecules are fairly similar, and there isn't a clear line that defines them
 - But the important part is to keep the definitions consistent across the project

Blog post example

- An example molecule would be a submit field made by using input and button atoms



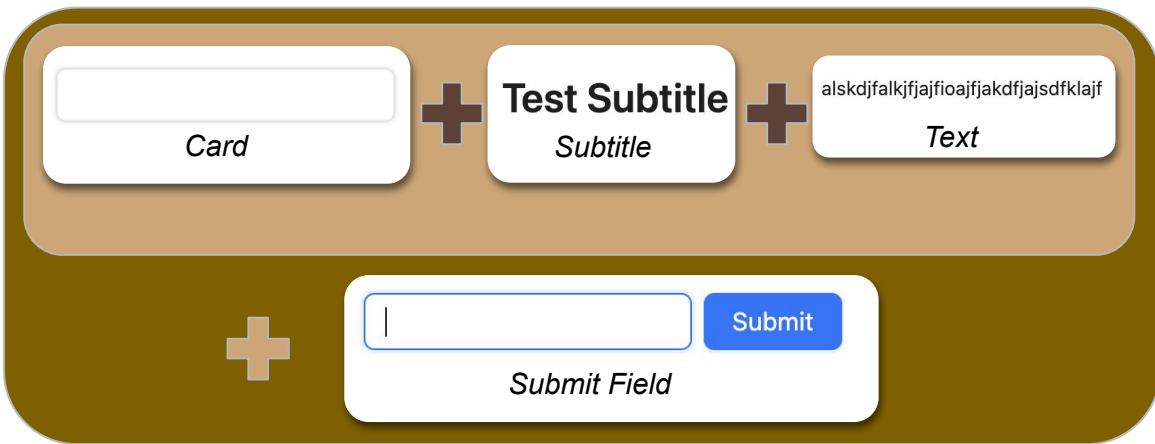
Organisms

Definition

- Organisms are usually usable components and are defined as **sections of an interface**
- Can be made from multiple molecules, atoms or even other organisms
- Ex: a navbar, forms, or popups

Blog Post Organisms

- For our example, let's define 2 organisms
- First, a post organism that consists of a title and content
- And a subscribe form organism that has a title, text and a submit field for users to submit their email



Templates

Definition

- Components coming together with *placeholders* in place of content
- Provide a **wireframe** for easy development and reusability
- Useful if there's constant reuse of a page layout
- Ex: an empty web page – no actual content, just placeholders



A Template for all Blog Pages

- The template for our ongoing example is the full homepage for the blog
- Complete with a title, template posts and a subscribe form at the bottom
- But with all template content



Pages

Definition

- Pages are **instance of templates**
- Placeholders are replaced with content that is seen by users
- Allows for all pages to be consistent in design

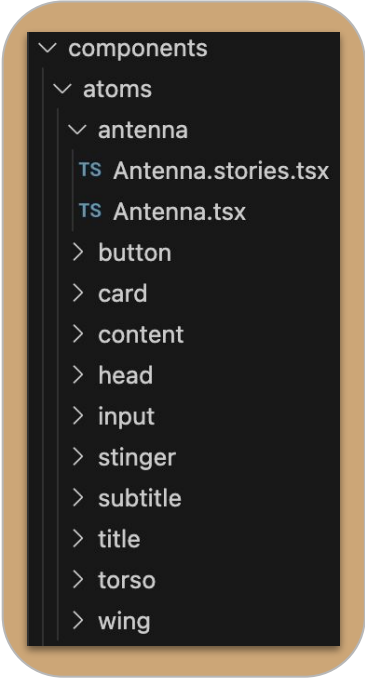
Creating a custom blog

- You'll see in the example that i've quickly filled out a blog just by providing *titles* and *paragraphs*

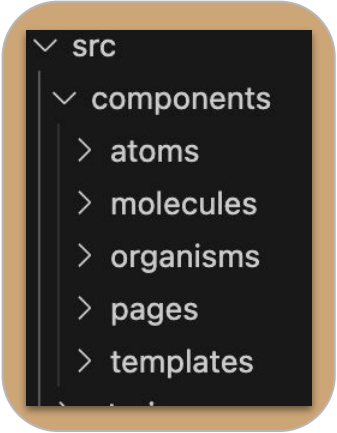


Folder structure

- There's no strict way to organize the different components in your project
- Only recommendation: **keep it consistent** across projects
- The example repo is one way of organizing components and their files
 - Folders that reflect the atomic design methodology
 - Files for components in their own folders



```
▼ components
  ▼ atoms
    ▼ antenna
      TS Antenna.stories.tsx
      TS Antenna.tsx
      > button
      > card
      > content
      > head
      > input
      > stinger
      > subtitle
      > title
      > torso
      > wing
```



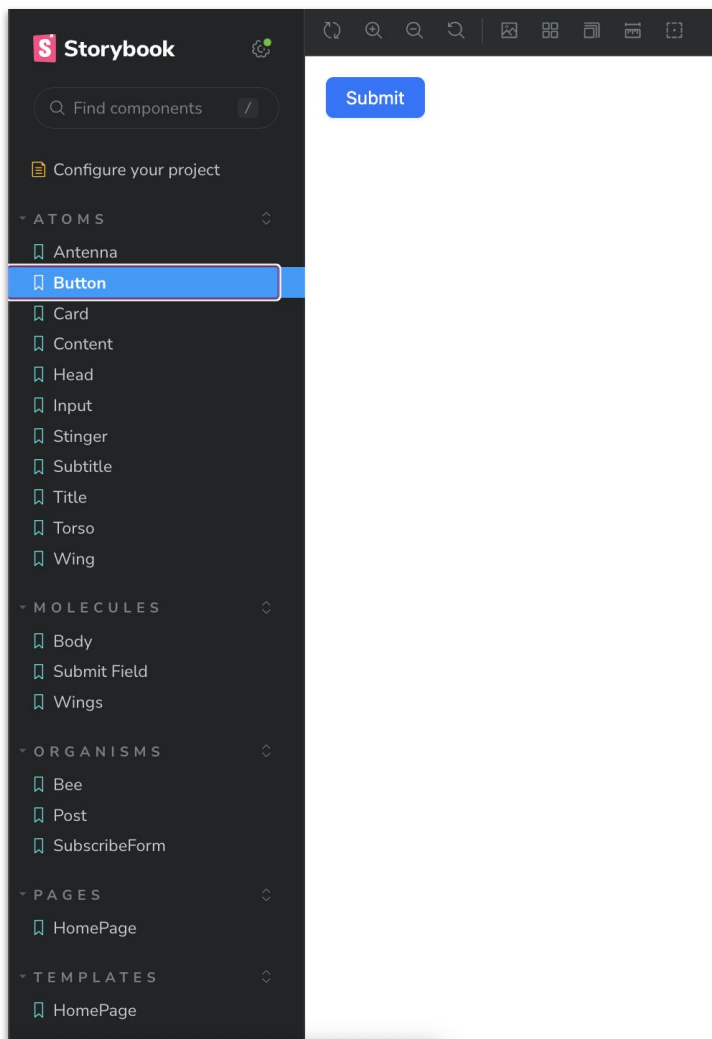
```
▼ src
  ▼ components
    > atoms
    > molecules
    > organisms
    > pages
    > templates
```

Storybook & atomic design

- This is more specific to the example repo and how it was built
- A short reading on this, with more information: <https://bradfrost.com/blog/post/atomic-design-and-storybook/>
- In short: stories are useful and a good way to **visualize** the components in relation to the atomic design methodology

```

└─ components
  └─ atoms
    └─ antenna
      ├── Antenna.stories.tsx
      ├── Antenna.tsx
      ├── button
      └── card
```



A slightly different example!

For a slightly different example less focused on providing value to users, I've included a bee example in the repo

- The different parts of the bee are broken down into atoms and molecules as needed, and the final bee is defined as an organism

Example repo:

- <https://github.com/tc2780/atomic-design-project>



Links

- <https://bradfrost.com/blog/post/atomic-web-design/>
- <https://bradfrost.com/blog/post/extending-atomic-design/#:~:text=Atomic%20design%20is%20a%20methodology,That's%20the%20extent%20of%20it>
- <https://bradfrost.com/blog/post/atomic-design-and-storybook/>
- <https://atomicdesign.bradfrost.com/chapter-2/>
- <https://medium.com/galaxy-ux-studio/principles-of-atomic-design-7b03a30c3cb6>
- <https://blog.logrocket.com/applying-atomic-design-next-js-project/>
- <https://blog.logrocket.com/atomic-design-react-native/>
- <https://blog.kamathrohan.com/atomic-design-methodology-for-building-design-systems-f912cf714f53>