Build a Design System in SwiftUI with the Atomic Design Methodology

Ibrahima Ciss

APPDEVCON 2024

About Me

🇸🇳

From
Senegal

👨‍💻

Indie
iOS Engineer

🐦

@bionik6
on Twitter

📺

Youtube channel
@iosmasterydev

🙏 Acknowledgments

📝 Agenda

- What is a design system?
- Components of a design system
- What is the Atomic Design Methodology?
- Case study
- Atomic Design Pros & Cons
- Resources

🎨 What is a Design System?

- A design system is a comprehensive guide that includes the design principles, visual language, components, and other standards that guide the creation of digital products within an organization.

- It works as a single source of truth for designers and developers, ensuring consistency and efficiency.

🧩 Components of a Design System

- Style guide: It includes information about typography, colors, spacing, imagery and the overall visual treatment of the product.
- Component library: This is a set of reusable components that can be combined to create UIs. Each of these components is defined with specific props and rules to cover every possible scenario or state.
- Pattern library: A collection of design patterns that offer solutions to common interface problems. For example, how to display notifications, how form inputs should behave, and more.
- Documentation: Detailed explanations and instructions on how to use the components and patterns. It can include code snippets, design tokens, usage guidelines, best practices, and anything that helps teams understand and use the design system correctly.

🧬 What is the Atomic Design Methodology?

Created by Brad Frost, Atomic Design Methodology is a principle for creating design systems. It borrows its concept from chemistry, breaking down interfaces into their atomic elements and working its way up to complex structures.

- Instead of designing pages, Atomic design focuses on creating and assembling various interface elements.

🧩 Components of Atomic Design

- Atoms: The most basic building block of an interface (like a label, input or a button). By themselves, atoms tend to be quite abstract but meaningful when used in context.
- Molecules: Simple groups of UI elements functioning together as a unit. For example, a search form molecule might include a label atom, an input field atom and a button atom.
- Organisms: Relatively complex UI components composed of groups of molecules (and possibly atoms). For example, a product card in an ecommerce store.
- Templates: Page-level objects where you place your organisms and molecules into a layout and see the design taking shape.
- Pages: Specific instances of templates that show what a UI looks like with real representative content in place.

👨‍💻 Case Study

Building a Movies app

🦾 Atomic Design Pros

- Scalability
- Reusability
- Consistency
- Efficiency

😩 Atomic Design Cons

- Overhead
- Rigidity
- Complexity
- Time-consuming

🤔 Should you use Atomic Design in your project?

It depends 🤷‍♂️

📚 Resources

Atomic Design by Brad Frost Talk: youtube.com

Atomic Design by Brad Frost blog post: bradgrost.com

😉 One last thing

👀 I'm looking for a new role

📮 Contacts

Follow me on twitter.com/bionik6

Linkedin: linkedin.com/in/bionik6

Youtube: youtube.com/@iosmasterydev

🙏 Thank you

🎤 Questions