

Demystifying design system

Summary

What we will see during this presentation

- What is Design system ?
- Why do we need Design system ?
- Atomic design
- Questions to ask

**To better implement a new approach or technology,
it's important to clearly understand the usefulness
and benefits**

What is Design system ?

- A design system is a collection (Library) of reusable components that can be built together to create any digital product.
- When you have multiple applications (Web sites, Mobile apps)
- Is related to all technologies, **not only iOS**

Design System Benefits

Cost Efficiency & Reusability

Design system benefits

- Providing pre-designed and reusable components
- Developers can quickly assemble screens, reducing the time and effort required to build the app.
- Designers also benefit from reusable design assets, which streamline their workflow and improve productivity.

Cost Efficiency & Reusability

Design system benefits

- Like Russian nesting dolls, little components are included in bigger components which are included in bigger components which eventually create entire screens.



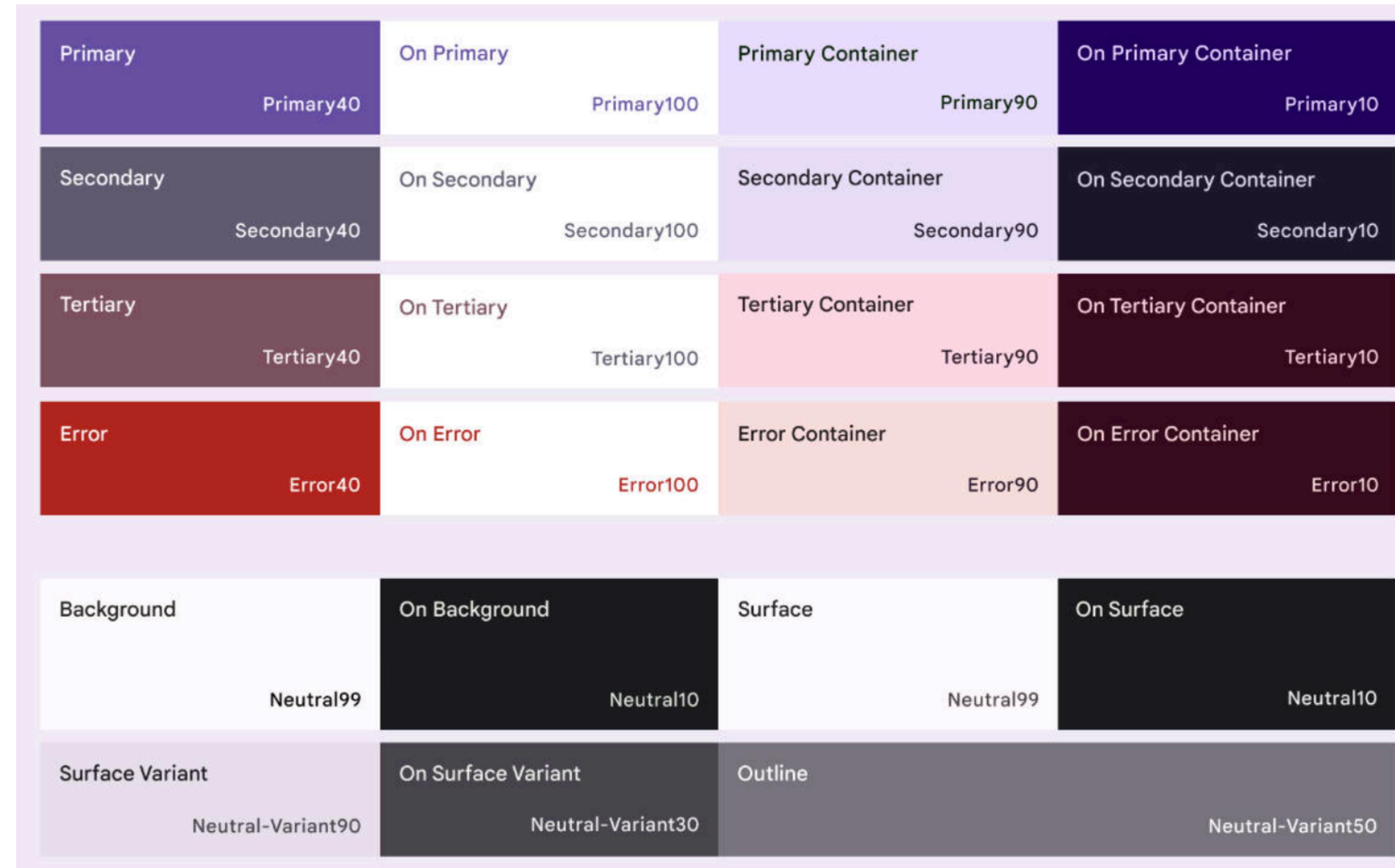
Collaboration

Design system benefits

- A design system facilitates collaboration between designers, developers, and stakeholders.
- Establishes a common language and understanding, enabling better communication and reducing the potential for misinterpretation.
- Designers can create and document design guidelines, while developers can refer to these guidelines for implementation.

Collaboration

Design guidelines examples



Collaboration

Design guidelines examples

heading-01

Goldman Sans Medium
40px

heading-02

Goldman Sans Medium
32px

heading-03

Goldman Sans Medium
24px

heading-04

Goldman Sans Medium
20px

heading-05

Goldman Sans Medium
16px

heading-06

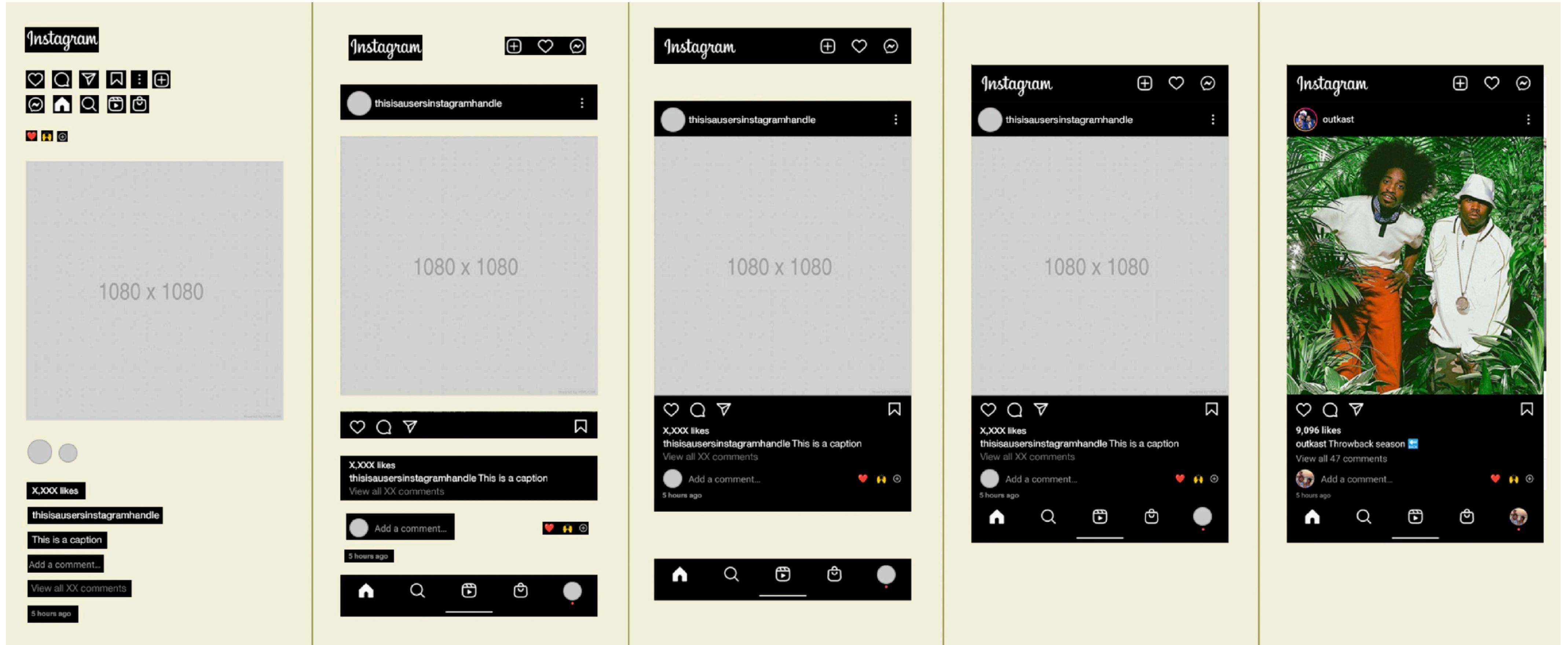
Goldman Sans Medium
14px

heading-07

Goldman Sans Medium
12px

Cost Efficiency & Reusability

Design system benefits



Branding & Better User Experience

Design system benefits

- Ensures visual and interaction **consistency** throughout the mobile app.
- Helps maintain brand consistency and strengthens the **app's identity**.
- Improved usability, and **reduced learning curves**.
- **Reduces confusion**, and builds trust with users.

Accessibility

Design system benefits

- Design systems can **incorporate accessibility guidelines** and best practices, ensuring that the mobile app is inclusive and usable by a diverse range of users
- By integrating accessibility features into the design system, developers and designers can **build accessibility directly into the app's foundation.**

Atomic Design

Atomic Design

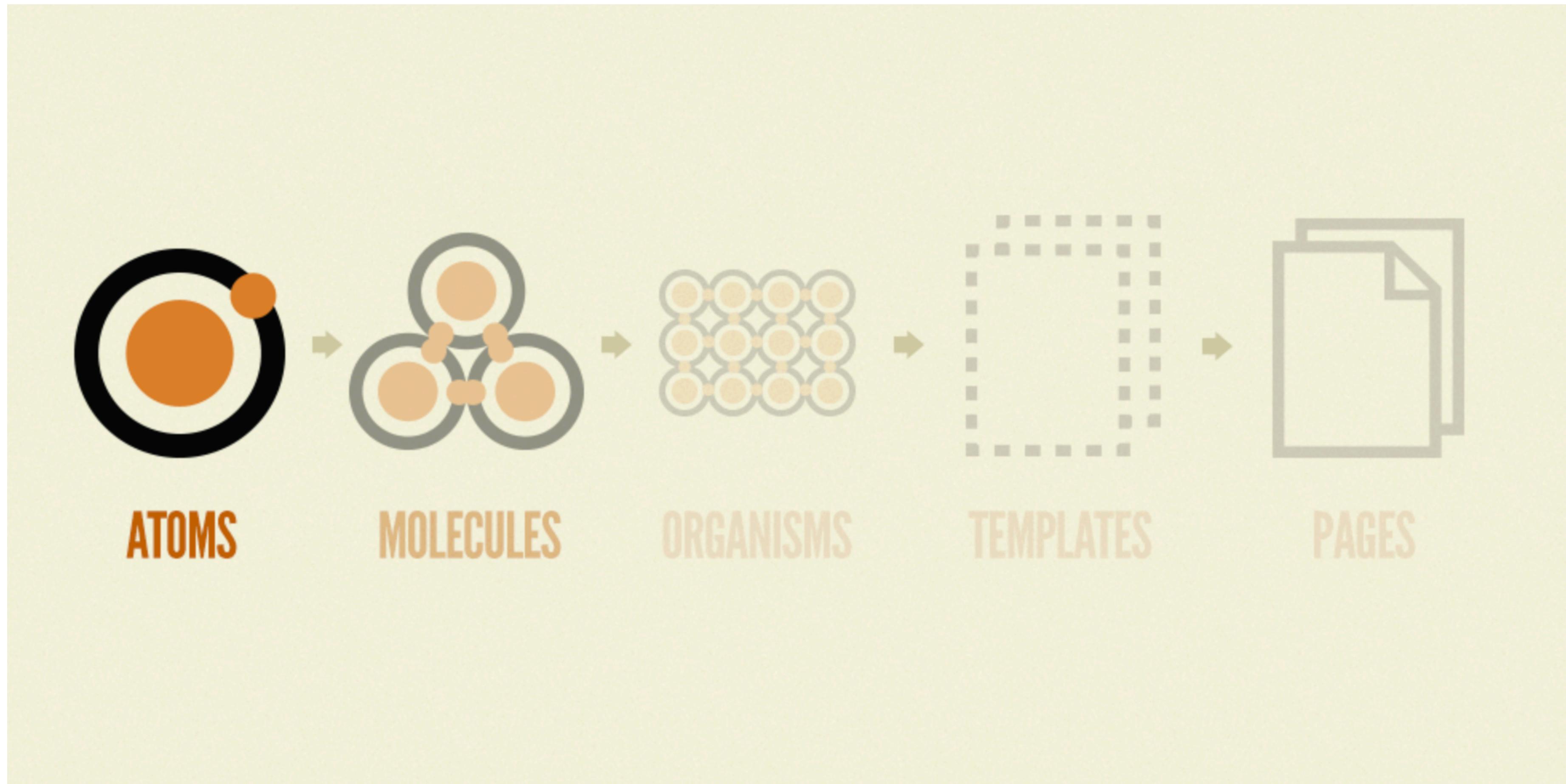
- All matter in the universe can be broken down into a finite set of atomic elements.

Periodic Table of the Elements																									
html																		col	table						
head	span																pre	meter	select	body	h1	section	colgroup	tr	
title	a																pre	meter	select	aside	h2	header	caption	td	
meta	rt	dfn	em	i	small	ins	s	br	p	blockquote	legend	optgroup	address	h3	nav	menu	th	pre	meter	select	body	h1	section	colgroup	tr
base	rp	abbr	time	b	strong	del	kbd	hr	ol	dl	label	option	datalist	h4	article	command	tbody	pre	meter	select	body	h1	section	colgroup	tr
link	noscript	q	var	sub	mark	bdi	wbr	figcaption	ul	dt	input	output	keygen	h5	footer	summary	thead	pre	meter	select	body	h1	section	colgroup	tr
style	script	cite	samp	sup	ruby	bdo	code	figure	li	dd	textarea	button	progress	h6	hgroup	details	tfoot	pre	meter	select	body	h1	section	colgroup	tr
								img	area	map	embed	object	param	source	iframe	canvas	track*	audio	video	device*					

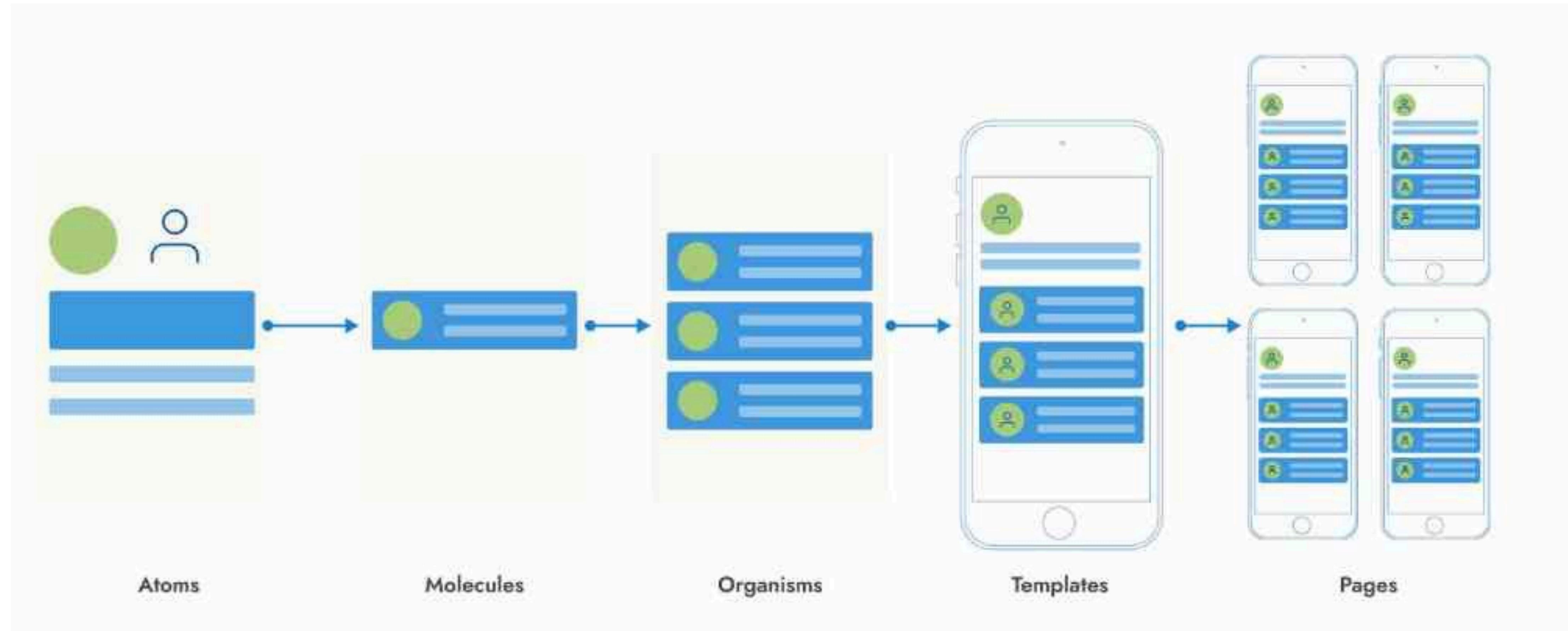
Atomic Design

- **Atoms** are the basic building blocks of all matter.
- **Molecules** are groups of two or more atoms held together by chemical bonds.
- **Organisms** are assemblies of molecules functioning together as a unit.

Atomic Design



Atomic Design



Atomic Design

Example of Atomic design implementation

- **Each atomic design components level is created in a separate folder**

```
graph TD; Root["> Utils"] --- Classes["<img alt='Folder icon' style='vertical-align: middle;' data-bbox='658 264 688 304' data-label='Image'] Classes --- Accessibility["> Accessibility"]; Root --- Atoms["> Atoms"]; Root --- Commun["> Commun"]; Root --- Extensions["> Extensions"]; Root --- Molecules["> Molecules"]; Root --- Organisms["> Organisms"]; Root --- Haptic["> Haptic"]; Root --- Resources["> Resources"];
```

The diagram illustrates a file structure for implementing Atomic Design. It features a purple rectangular background with a white list of folder names. Each folder name is preceded by a dark gray folder icon and a right-pointing arrow. The structure is organized into levels: 'Utils' is at the top level; 'Classes' is a sub-level under 'Utils', indicated by a downward-pointing arrow; and the remaining seven categories—'Accessibility', 'Atoms', 'Commun', 'Extensions', 'Molecules', 'Organisms', 'Haptic', and 'Resources'—are at the third level.

- >  Utils
- ▼  Classes
 - >  Accessibility
 - >  Atoms
 - >  Commun
 - >  Extensions
 - >  Molecules
 - >  Organisms
 - >  Haptic
 - >  Resources

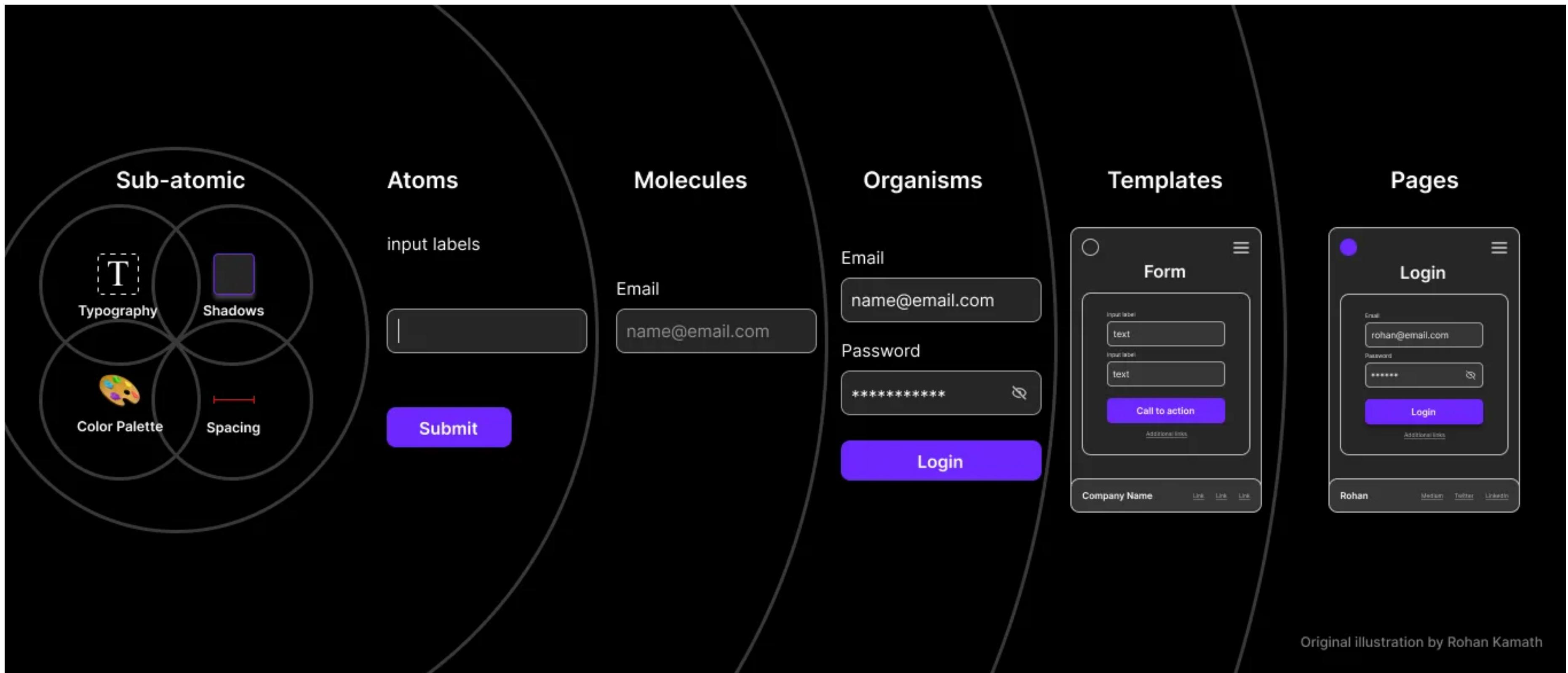
Questions to answer

Colors are atoms ?

- Atomic design on mobile has multiple implementations and way of doing
- One common difference is **whether we use colors & fonts as Atoms or as separate components**

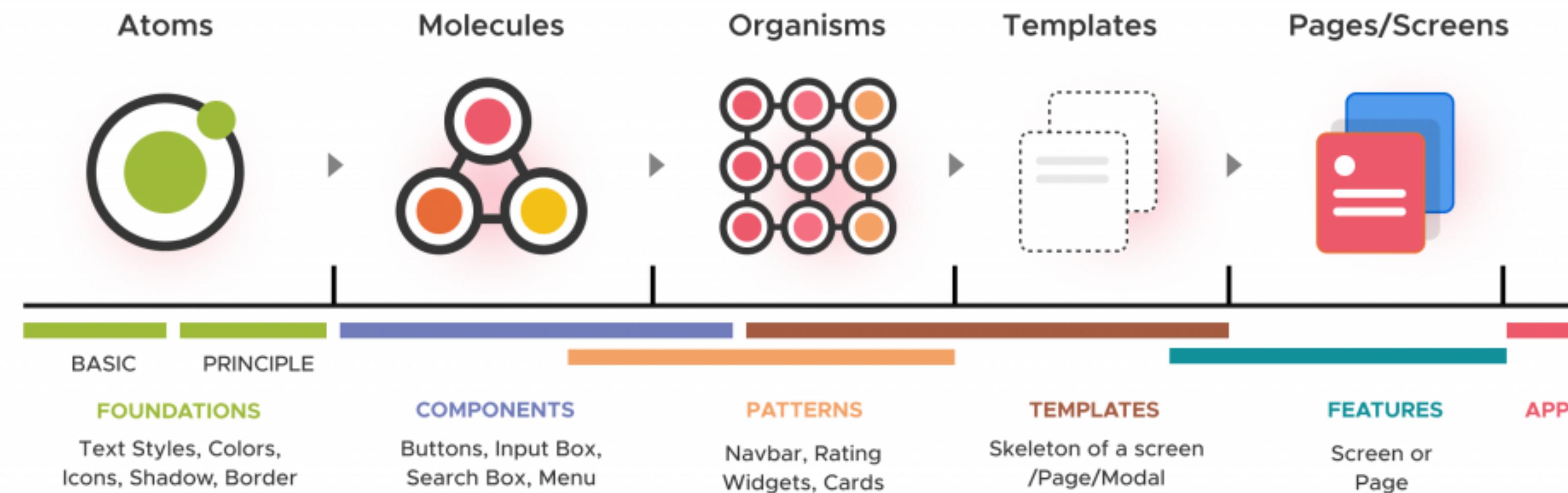
Colors are atoms ?

Colors as SubAtomic



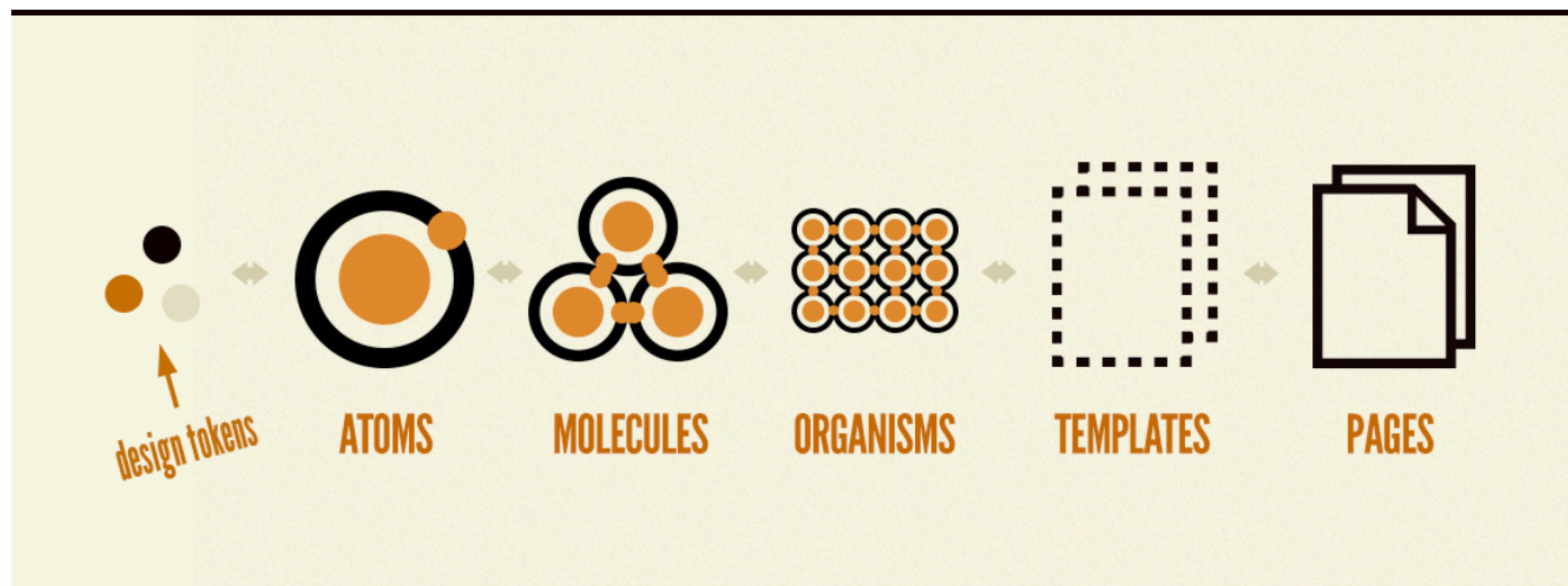
Colors are atoms ?

Colors as Atoms



Colors are atoms ?

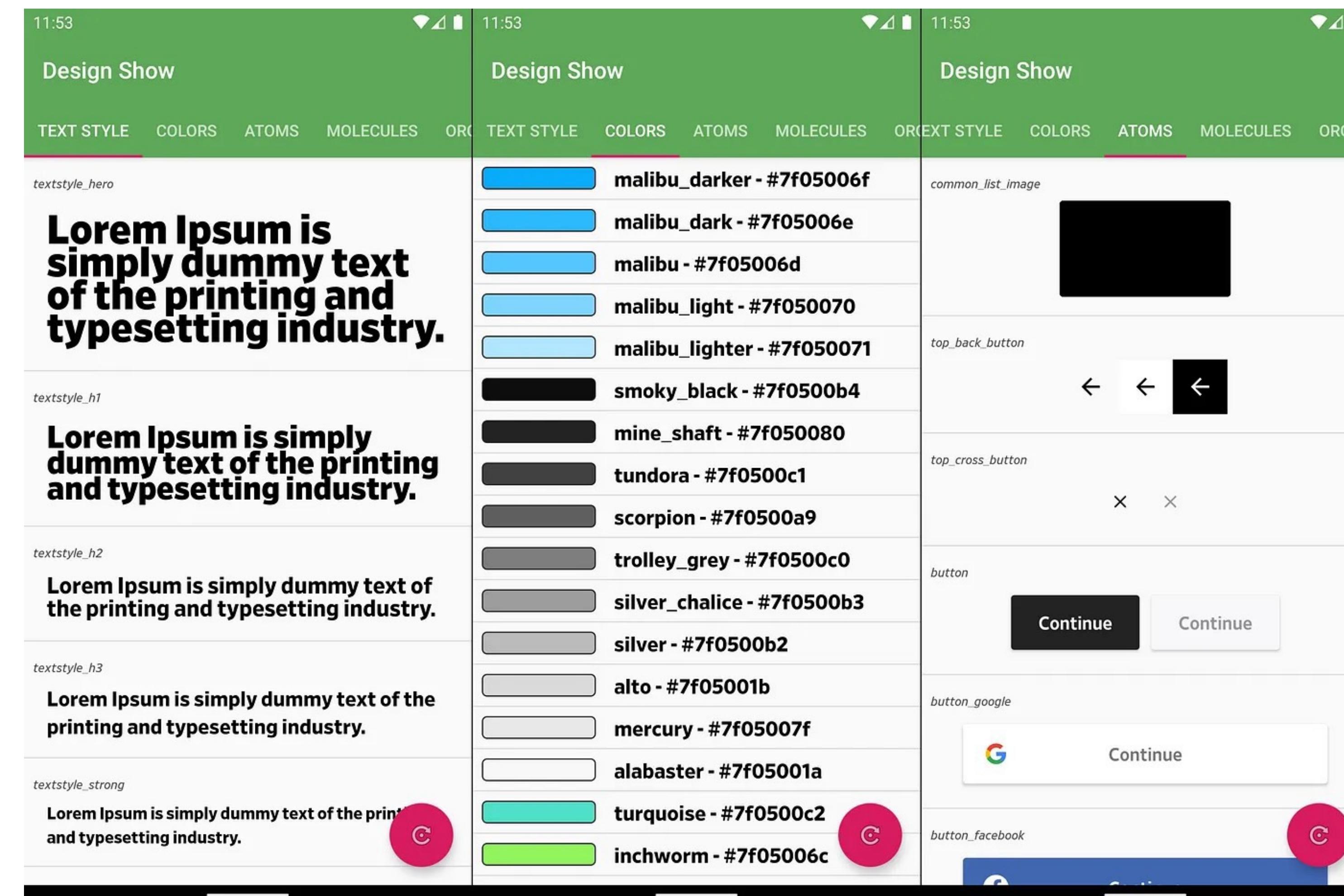
Colors as Token



Demo App

- A common practice is to create a separate demo app which shows all available UI Components
- The demo app is usually included to the CI CD in order to provide the Designers and the product an overview of all UI Components
- Demo app should allow to switch Dark/Light mode
- Demo app might me difficult to keep up to date.
- Could it be generated ?

Demo App



More technical questions

- Should we use **view model** for each view ?
- Should we put **ViewControllers/ (SwiftUI) Views** in Atomic design Framework
- How can we align to **Apple's Human Interface Guidelines**

References

- Implementations ...?
 - <https://github.com/Orange-OpenSource/ods-ios>
 - <https://github.com/Decathlon/vitamin-ios>
 - <https://github.com/jayeshk/Doric>
 - <https://github.com/GettEngineering/Prism> // Generate from Zeppelin
 - <https://iosexample.com/design-system-mobile-for-ios-swiftui/>
 - <https://github.com/imodeveloperlab/dskit>
 - <https://www.ramshandilya.com/blog/design-system-intro/>
- Atomic Design
 - - <https://lagrandeourse.design/blog/ui-design-et-da/la-difference-entre-le-design-system-et-latomic-design/>
 - - <https://atomicdesign.bradfrost.com/chapter-2/>
 - - <https://uxdesign.cc/atomic-design-how-to-design-systems-of-components-ab41f24f260e>
 - - <https://jobandtalent.engineering/visualkit-ui-framework-74ab8aae0d42>
 - - <https://coyotesystem.atlassian.net/wiki/spaces/DSI/pages/163577861/Design+System>
 - - <https://bradfrost.com/blog/post/extending-atomic-design/>
 - - <https://github.com/mercari/fractal/tree/master/DesignSystem/Sources/DesignSystem>