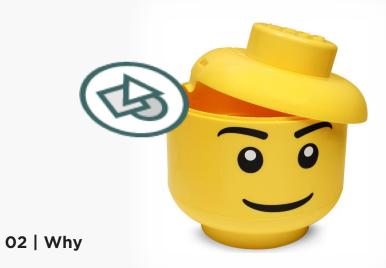
# BOSE PATTERN LIBRARY

RE-USABLE UI COMPONENTS FOR BOSE MOBILE APPLICATIONS

BOSE

## Why a Pattern Library?

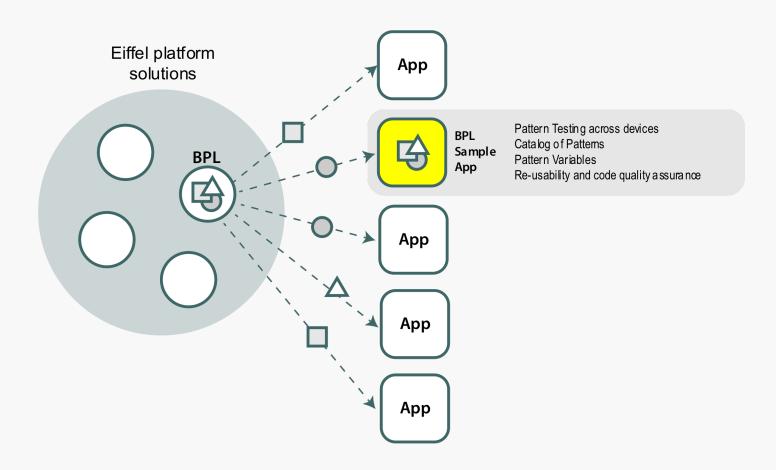


- Common/consistent visual language between design, development and mobile
- Central control and inventory of mobile UI components
- Create a component/pattern once and use it/iterate infinitely
- Cost savings: development, design, maintenance
- Increased agility; faster build out of Go to Market mobile experiences

# What is Bose Pattern Library (BPL)

#### 03 | What

- A tool that offers mobile UI re-usable components and is used by developers and designers
- BPL lives on a common platform like Eiffel so that any Bose app vertical can use it
- BPL is illustrated by a native mobile BPL sample app (iOS and Android) Its purpose is to catalog, display and test each pattern and its variables
- BPL is built with Bose development & design standards so that any Bose product team could consume it or build upon it



SOURCE:

# What is a reusable UI component

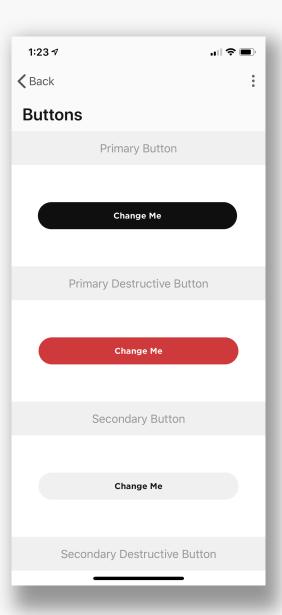
#### 04 | Reusable Component

- It's a basic stand-alone UI element/control that is branded, such as a button or a text input field
- Some components can be built with multiple atomic UI elements ie. an error screen pattern
- Each component is build modular with variables and extensibility in mind, serving wide spectrum of needs and applications
- Each component is build with mobile accessibility and is tested across multiple devices, screen sizes and rotations
- Component code is well commented to document variables, UI options and its use









SOURCE: 4

# What we've done so far for BPL

#### 05 | What we've done as a BPL POC

- We've re-factored about 40 Madrid components including patterns
- These components are represented inside a BPL sample app (iOS and Android)
- We've integrated several BPL patterns back into Madrid code base
- Each pattern was built with mobile accessibility, supporting screen rotations
- Created iOS automation test scripts to automate pattern testing



## Project Elixir

What's Next

#### 06 | Project Elixir

- Bring project through SWAG as an agile program
- Determine goals, create an aligned roadmap and start executing on clear set goals



## Project Elixir

#### 01 | Technical Debt

- BPL code refactor, architecture debt
- Continue moving UI components out of Madrid into BPL and making them more modular and reusable

#### 02 | UX Debt

- While we are moving components out of Madrid into BPL we work with UCD to complete pattern vision
- Document and organize existing components in Confluence

#### 03 | New Patterns

- New patterns, added to the mobile app backlog as work to be prioritized by NPM in collaboration with UCD
- Other product feature teams can contribute new approved components into the BPL. Each feature team collaborates closely with BPL team to determine if a pattern is a good candidate for inclusion. Once the pattern is accepted BPL dev team ensures pattern is built to the library code standards



### Elixir Vision

#### 01 | Short Term

- Elixir is a living and ongoing effort that adapts to new processes as we encounter issues and discover new needs
- Determine UI component governance for inclusion into BPL
- New mobile UI components are created as part of the BPL, so that all Bose app verticals can leverage them
- Patterns comply with platform development standards and are easily consumable
- Bose mobile accessibility rules are clearly defined and BPL components abide by them

#### 02 | Long Term

- All Bose mobile app products leverage BPL for base UI
- Creation and iteration of UI components is well tuned process backed by experience
- BPL grows into a central control for updating and maintaining UI components across all mobile applications
- Rapid prototyping, BPL is enabling designer or a developer to put together mobile app experiences that can be designed, developed and tested in a short time

