OTG SOFTWARE

TRANSITION TO REACT NATIVE

Steps to accomplish this

REFACTOR FOR MEGA REUSABILITY

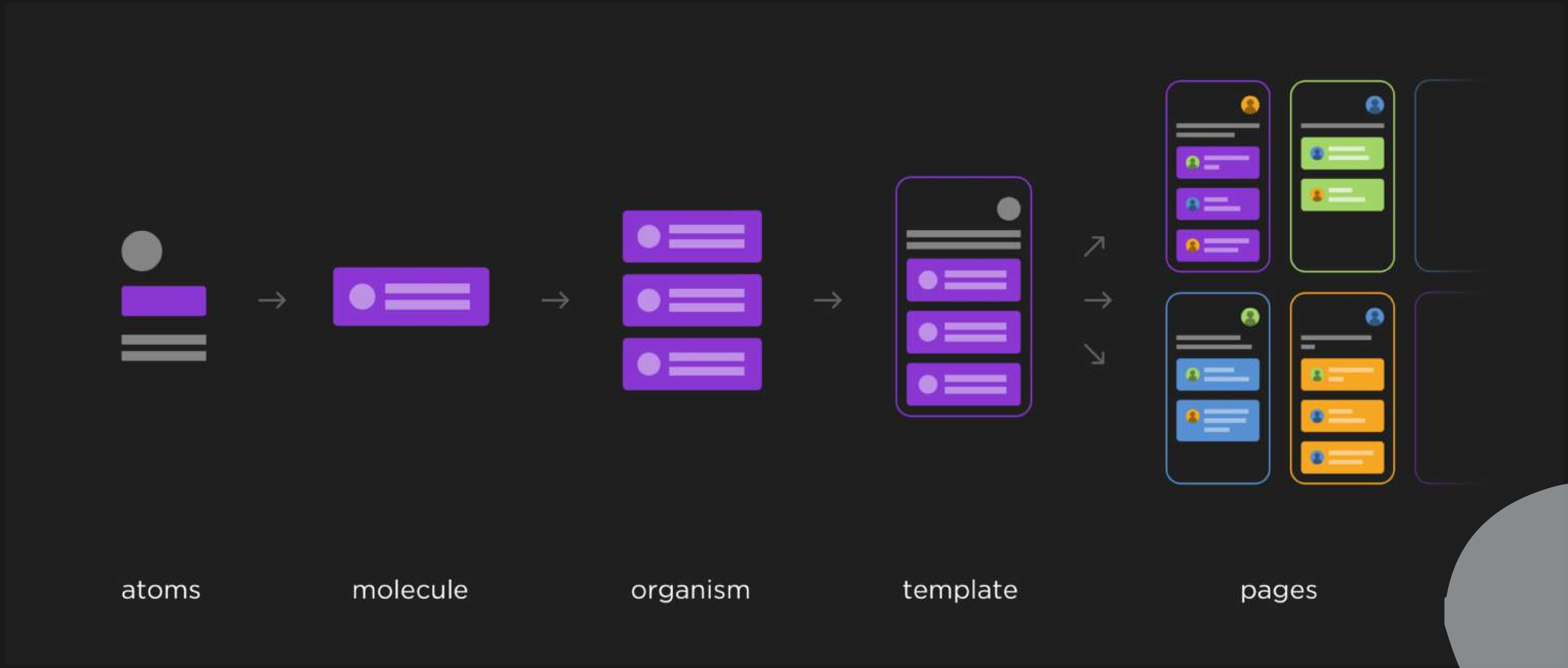
OTG SOFTWARE | 2019

Merge Order Ahead into Web Cart

- Order Ahead is really just a web-based Concierge with a few different rules.
- Order Ahead does most of what Concierge does without any help.
- Web-Cart has polished styling and functionality Order Ahead could share.
- Order Ahead should be converted into modular React Native Web components and shared with the Web-Cart.
- This will both give Order Ahead more capability and expand Web-Cart to take over most of Concierge's responsibility.

REUSING COMPONENTS WITH

ATOMIC DESIGN



ATOMIC DESIGN

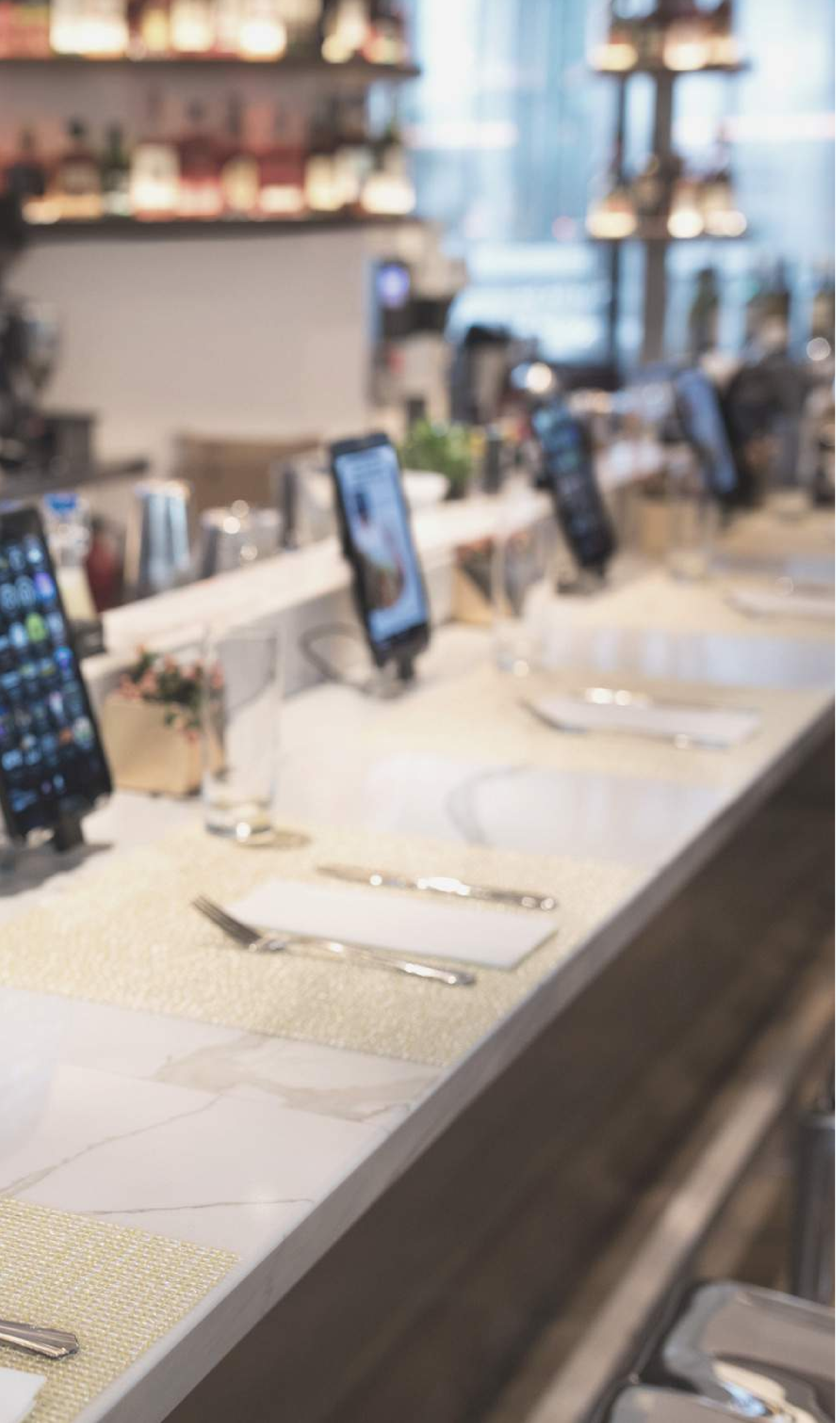
BUILDING AN OTG
UI COMPONENT LIBRARY

- Use Bitsrc.io to host and manage our components
- Each component lives in a mini git repository with a self sufficient environment for maximum reusability.
- Each component can now be imported into other projects and maintained in a single centralized place.
- It's clean and encapsulates code based on a single purpose.
- Each component will document its usage with Storybook and include unit tests to ensure no code regressions.
- Bit React Native Web Components let us share components between web and mobile apps for ultimate reusability.

MOVING TO REACT

REDUCING IOS RESPONSIBILITIES

- Order Ahead has almost all the api calls Concierge has and takes full responsibility for it's data.
- It handles all it's own menu math, but also uses Papi's calcTotal endpoint as well as handling all item views, and payment.
- Web-Cart handles miles and currency payment types, open tab, open check, promotions, modifiers, cart, upsells, item modal, promo/vouchers codes, gratuity selection and translations.
- If we convert the views from Order Ahead to React Native Web and integrate them with the Web-Cart, we will have covered most of the remaining menu screens and functionality in Concierge.
- We could have a prototype for a full menu React app ready to test in 1 - 2 months if Colin and I work on it together.
- If we start 100% from scratch it would likely take 3-6 months to get to a working prototype.



WHAT WILL IOS STILL DO?

BASE APPLICATION AND HARDWARE INTERACTIONS

- iOS will still run the main screen and its connection to the games apps and sponsored ad bubbles. That will stay the same for now.
- iOS will load the entire menu section of Concierge by web-view, but direct React Native integration would be preferable if time permits.
- The native iOS layer will continue to handle the boarding pass scanner.
- Depending on compatibility, the miles library could stay in iOS
- Data from the native iOS app can be "bridged" into React Native similar to how it talks to the Web Cart now, but with even better performance.

Full Menu React

KEY BENEFITS

INSTANT DEPLOY

PLATFORM AGNOSTIC

RECORD & LOG
FULL ORDER EXPERIENCE

ESCAPING THE FIRE

WE HAVE TO BUILD OUR WAY OUT

Concierge has accumulated tech debt over many years, it's original developer is gone, it's hard to deploy, hard to debug, and as a result it's very difficult to support and build onto.

Instead of starting a new cycle of feature development, mobile developers on our team are throwing buckets of water at the flames.

The new system is easier to understand, easier to fix, and will speed up development time through improvements in tooling and component sharing.

Building our way out of this and moving responsibilities from iOS to React is the direct path to stability, control, and instant deploys.

TECHNICAL WISDOM

WHEN THE BUILDING YOU ARE
STANDING IN IS ON FIRE
GET OUT OF THERE.

DAVID MIELOCH