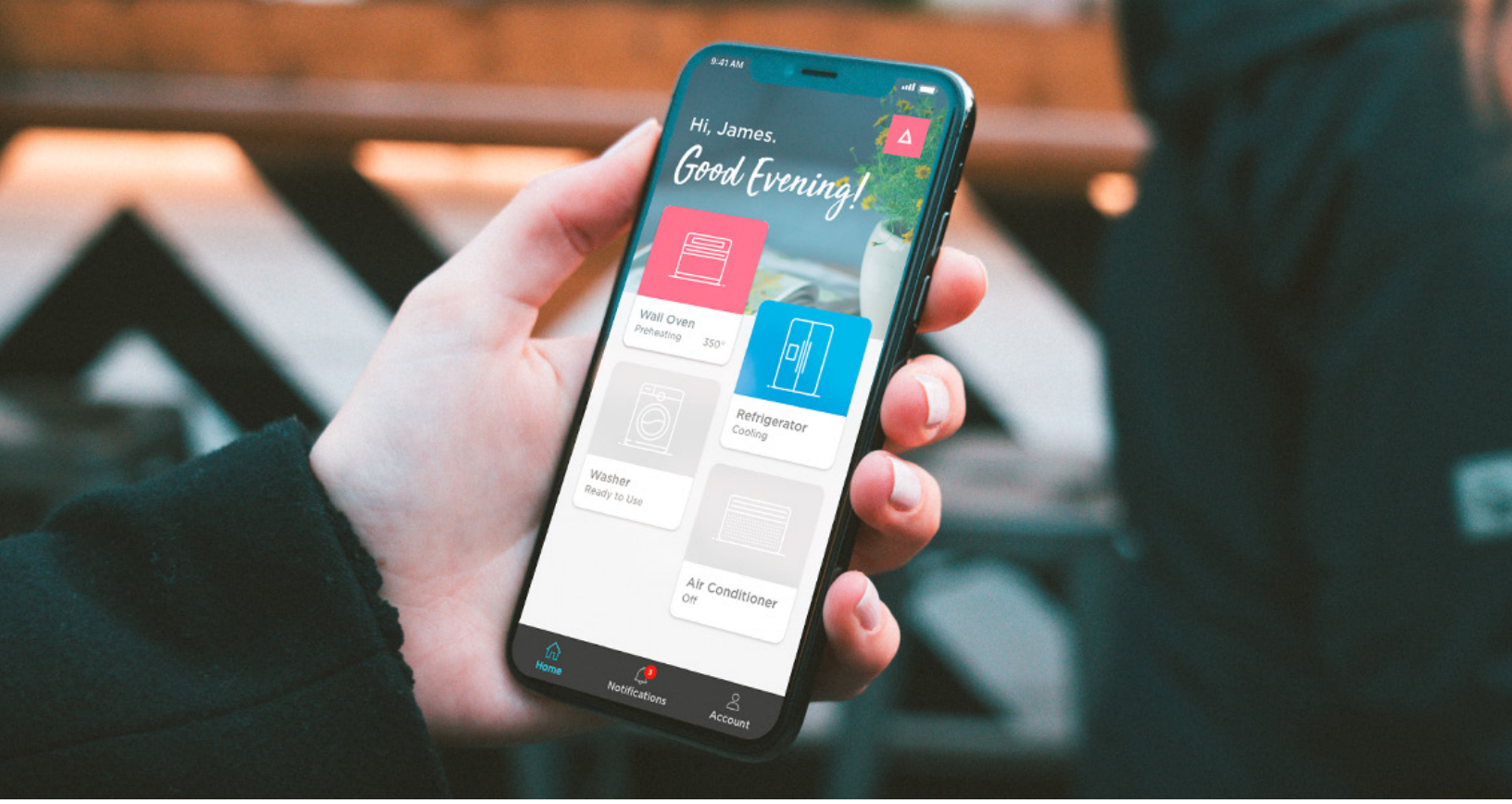



FRIGIDAIRE

# Advancing Smart Home Technology with Connected Appliances

## Project Goal

Launch iOS and Android applications for Frigidaire and parent brand Electrolux. Design remote monitoring and control features for a growing product line of connected home appliances.

## My Role

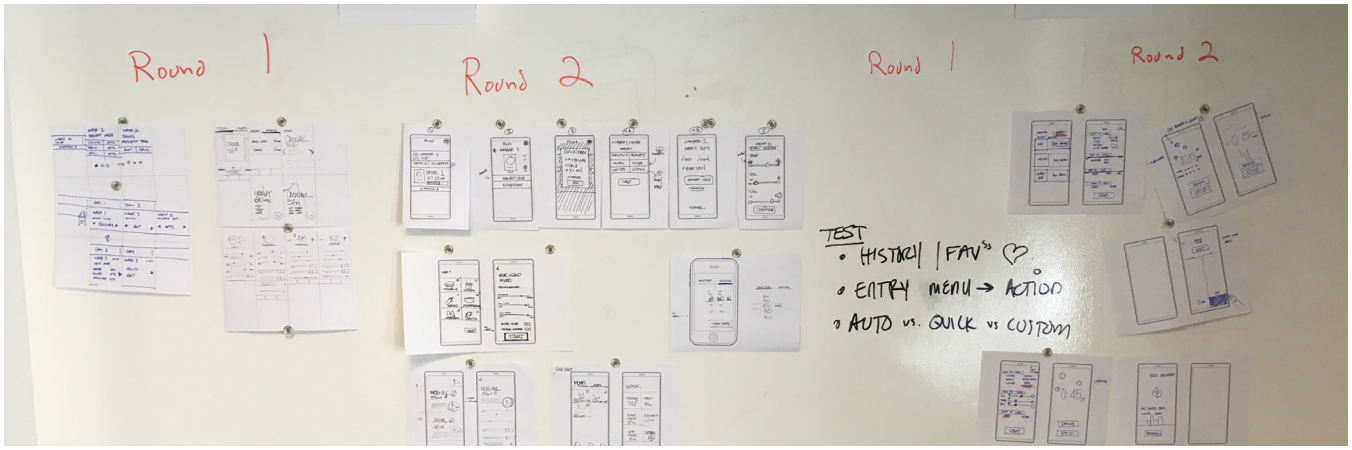
As UX Lead on the project it was my responsibility to facilitate workshops, clarify requirements, define hypotheses to test weekly, document functional specifications and deliver iOS and Android UI assets for development.

# Getting Started

We quickly ramped up for this 12 month project by reviewing personas, user stories, legacy Frigidaire apps and research from previous projects. With only 4 weeks to deliver the first connected appliance experience we had to move fast.


### Design Studio Workshop

We assembled internal and client-side design teams for a 3-day design studio to help jump start the design process with a holistic understanding of feature sets across product lines. Four teams of designers sketched 6 appliance experiences in 2 days.

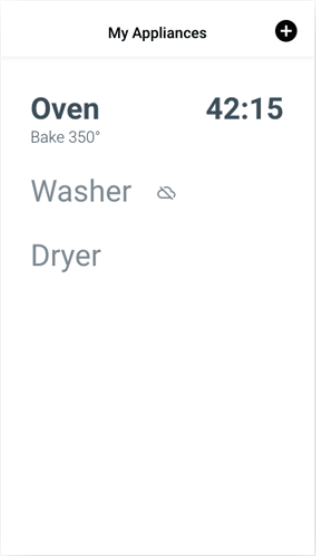


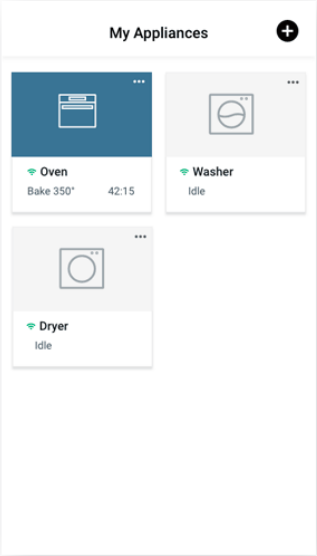

Each attendee brainstormed several appliance control and monitoring concepts to present to the team. Teams identified strengths and weaknesses of each concept then sketched collaborative concepts to share with the other groups. Together we identified common themes or UI patterns then determined which ideas to test on day 3.

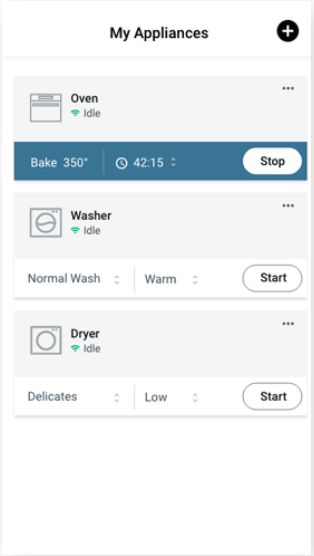
# Low-fidelity Prototyping

On day 3 of the design studio we tested low-fidelity prototypes for usability and desirability. The goal of this research was to understand how much control participants expected during various statuses (idle, running, offline), and how they expected to navigate appliances with multiple cavities.

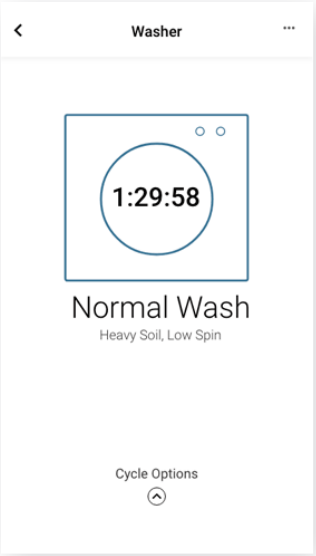
## Dashboard Experiments

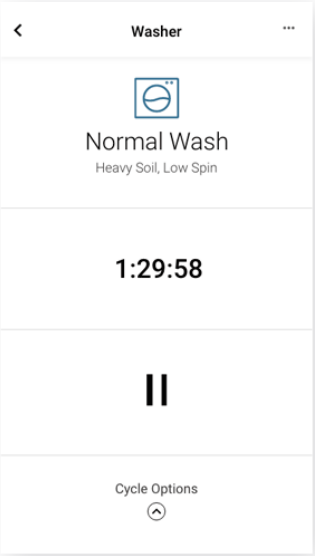


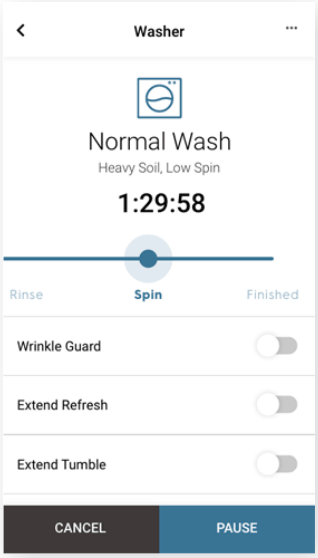



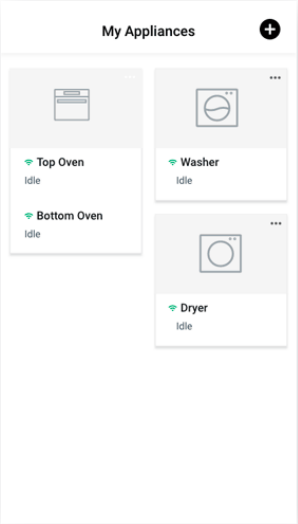

## Appliance Monitoring Experiments

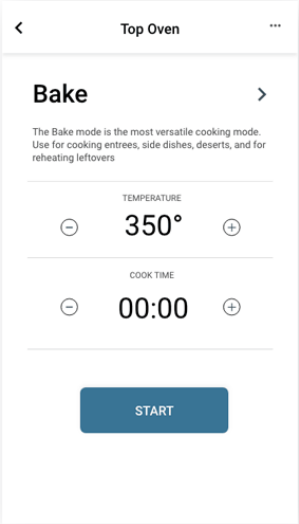


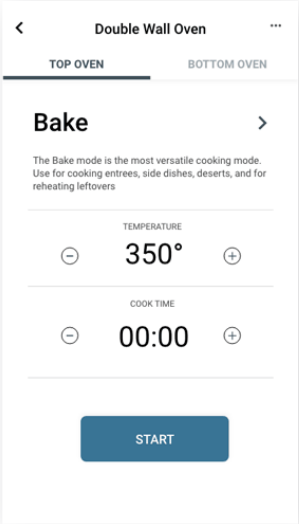



### Navigation Experiments



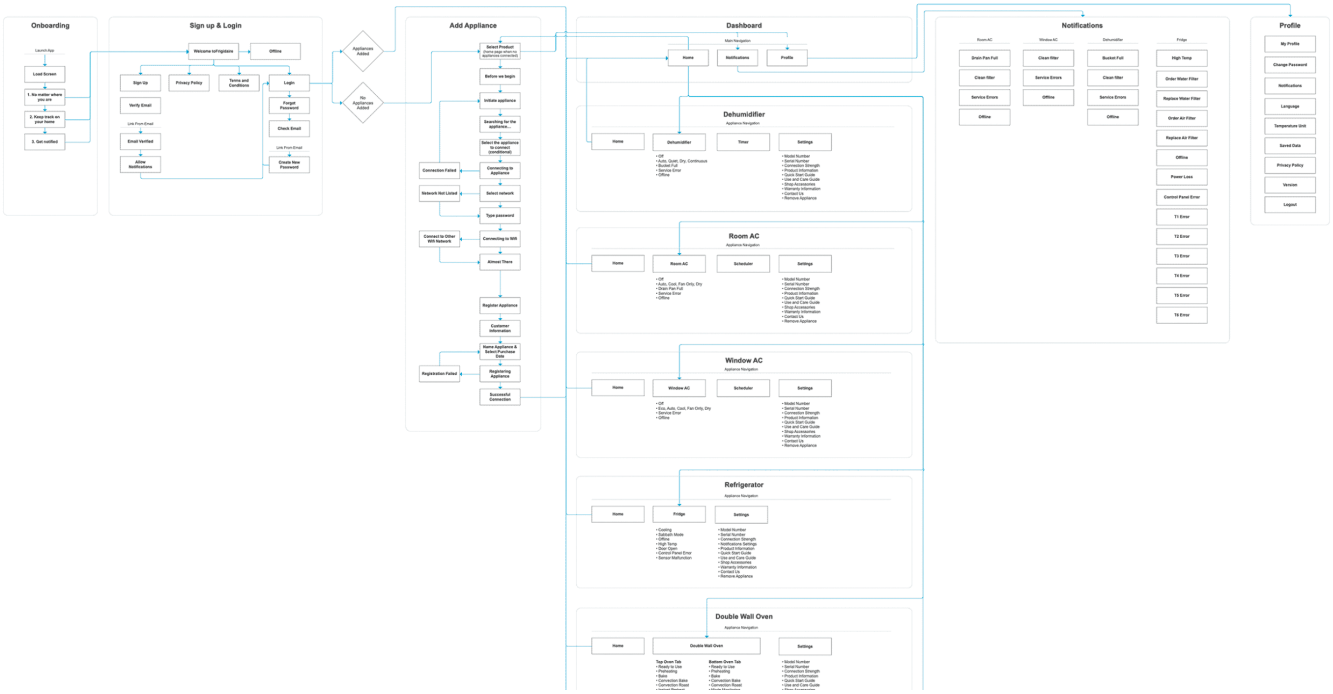




## App Architecture

With hundreds of user stories and a growing design team, tracking design work was a challenge. At the end of each sprint we updated this architecture document with changes to the app and used color coding to show what was in progress and what was completed

View wireframes and app architecture deliverable

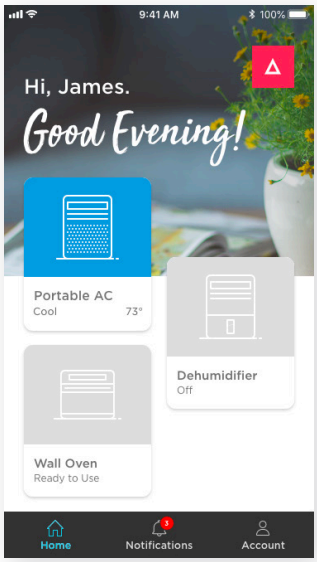


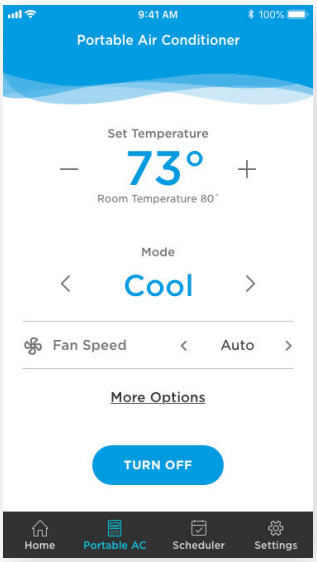
# Iterative Design Process

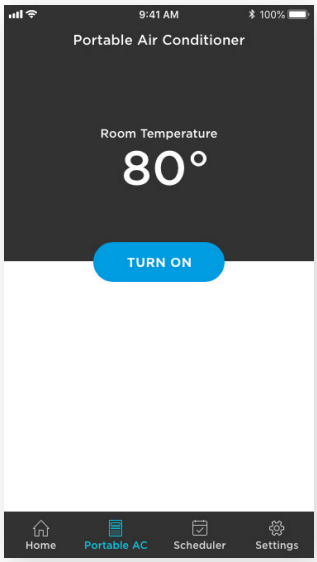
Each week we created prototypes to test with participants. Regular testing throughout the design process helped expand the design system with usable UI patterns that could be used across product lines.

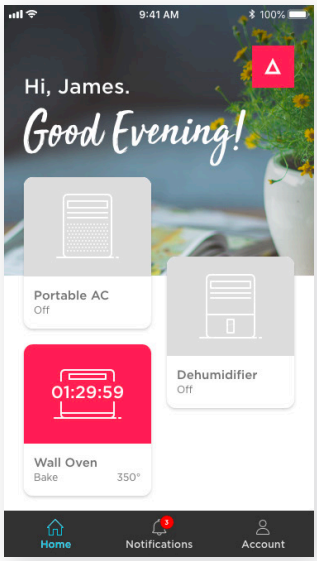
### Appliance Control Paradigms

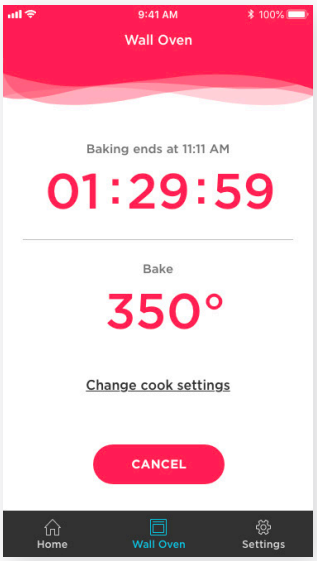
An important design pattern we obsessed about was appliance status display. The on/off paradigm of an air conditioner was different than the start/stop paradigm of an oven. All appliances worked in one of these 2 ways so getting this correct was essential to designing a usable experience.

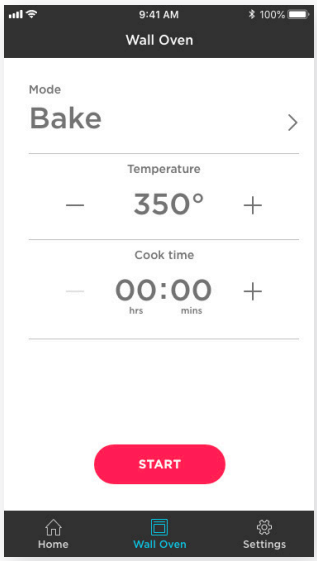


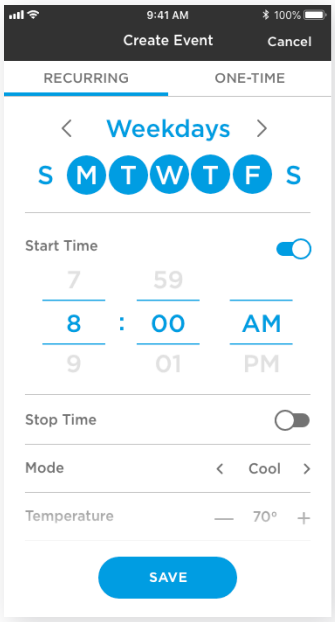



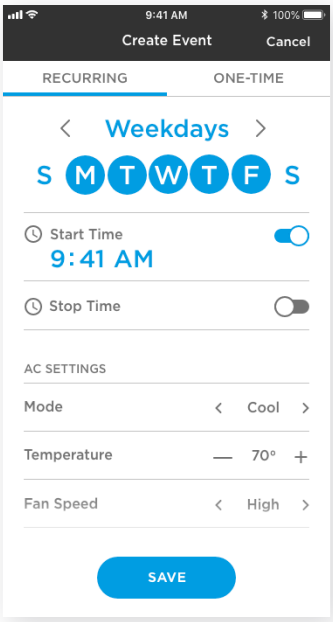
### Appliance Automation

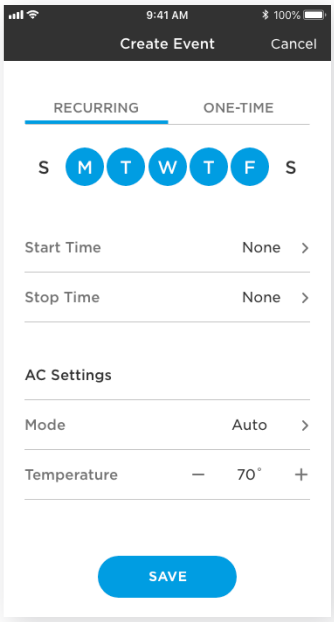
The scheduling feature for air conditioners took several weeks of iteration to finalize. We started by looking at apps from Nest, Hue and Apple for inspiration.

The visual matrix of time and temperature offered by Nest seemed like an intuitive approach but wasn't technically feasible in the time we had to deliver this feature.

Firmware limitations also eliminated some early design concepts from testing. We decided on the calendaring paradigm used by Hue Lights and native alarm functionality from Android and Apple.

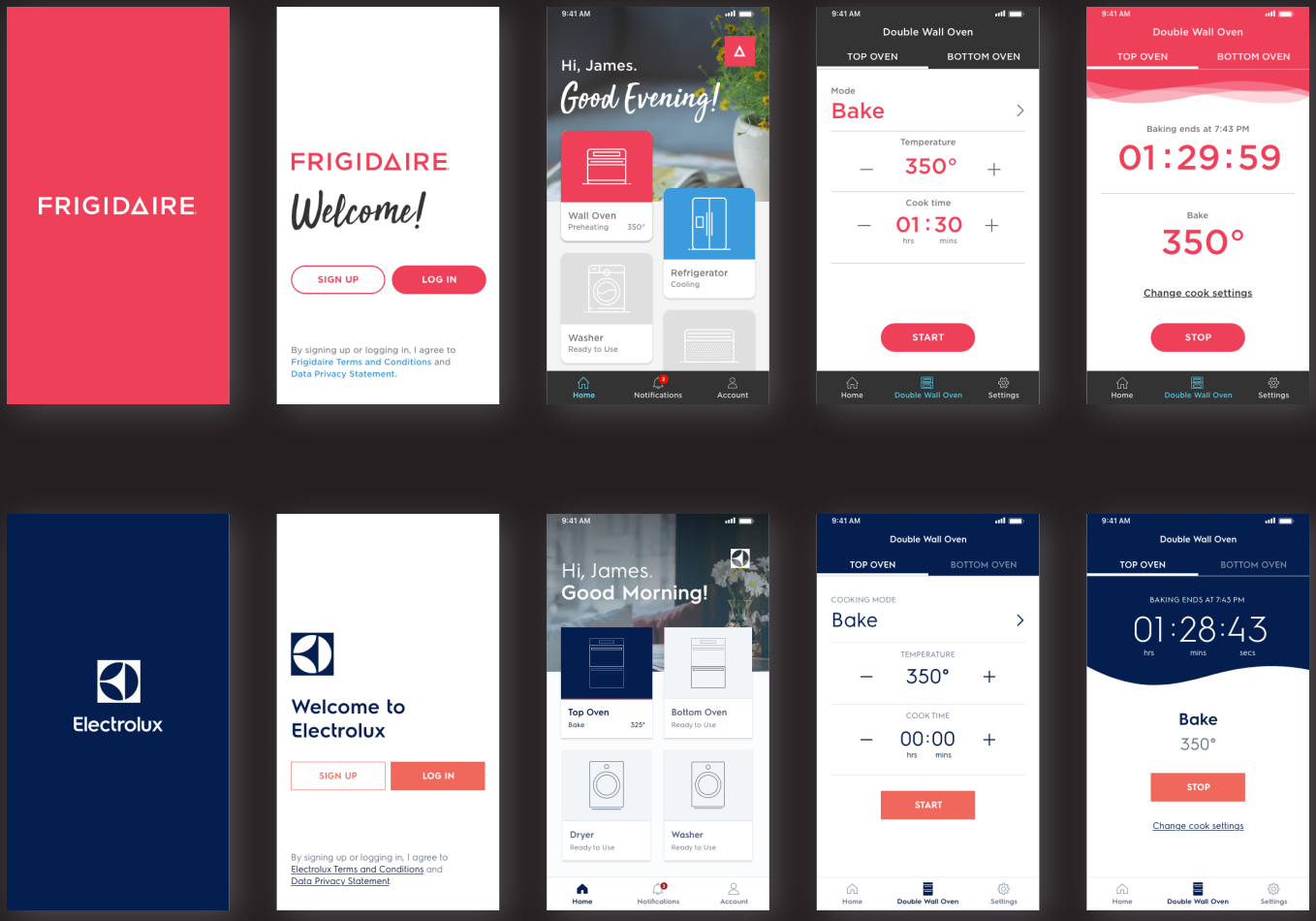


# Design System Theming

After a few months of design we started experimented with theming the design system. We wanted to provide enough customization for brand expression but without compromising usability and consistency. After many color, type and photo experiments, a UI kit emerged for both brands.



# Interactive Documentation

At the end of each sprint we documented the experience with Overflow, published UI assets with Zeplin and shared everything in Jira.

View design hand-off to development