

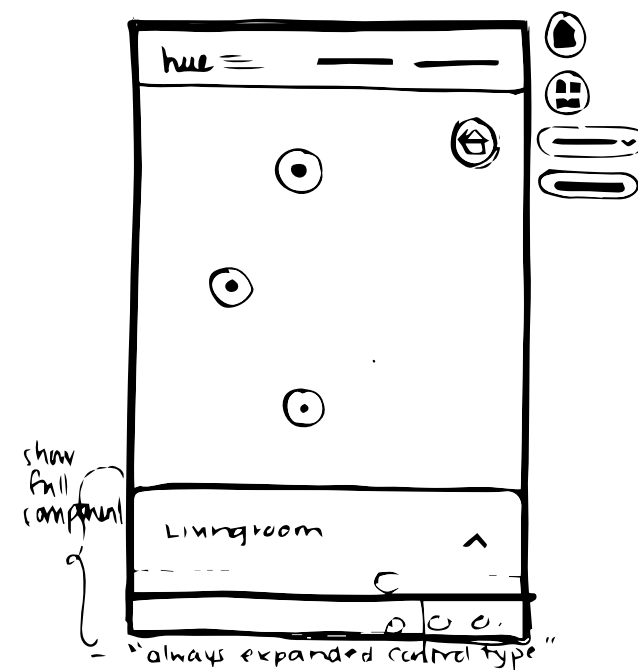
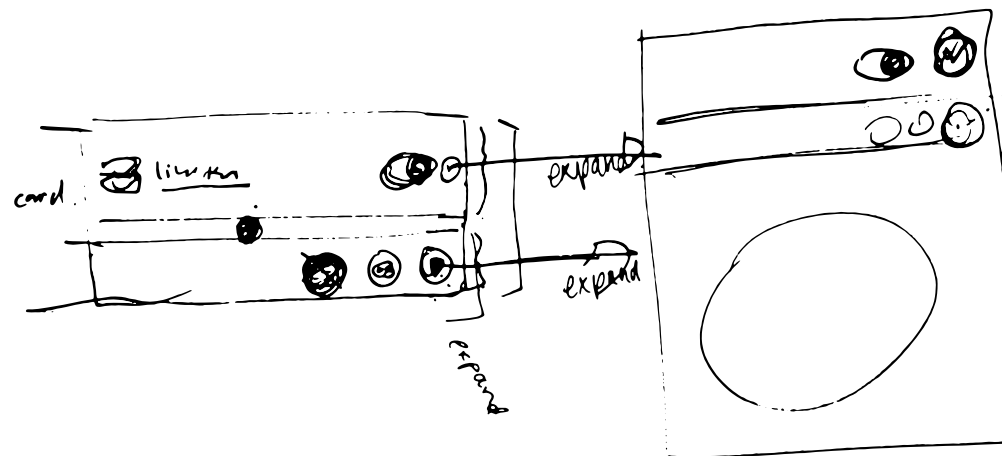
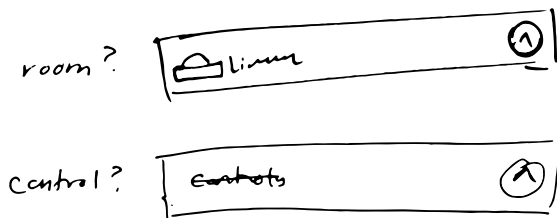
Experience Hue

Sketches and Iterations

Utilizing Existing Design Patterns

When trying to align the application designs with the proposed mobile-web designs, there was the challenge of determining just how much can you adapt the design before it becomes inconsistent and not on brand, all the while making sure it communicates the experience. In this case, the existing design patterns needed to be adapted to the needs of the experience.

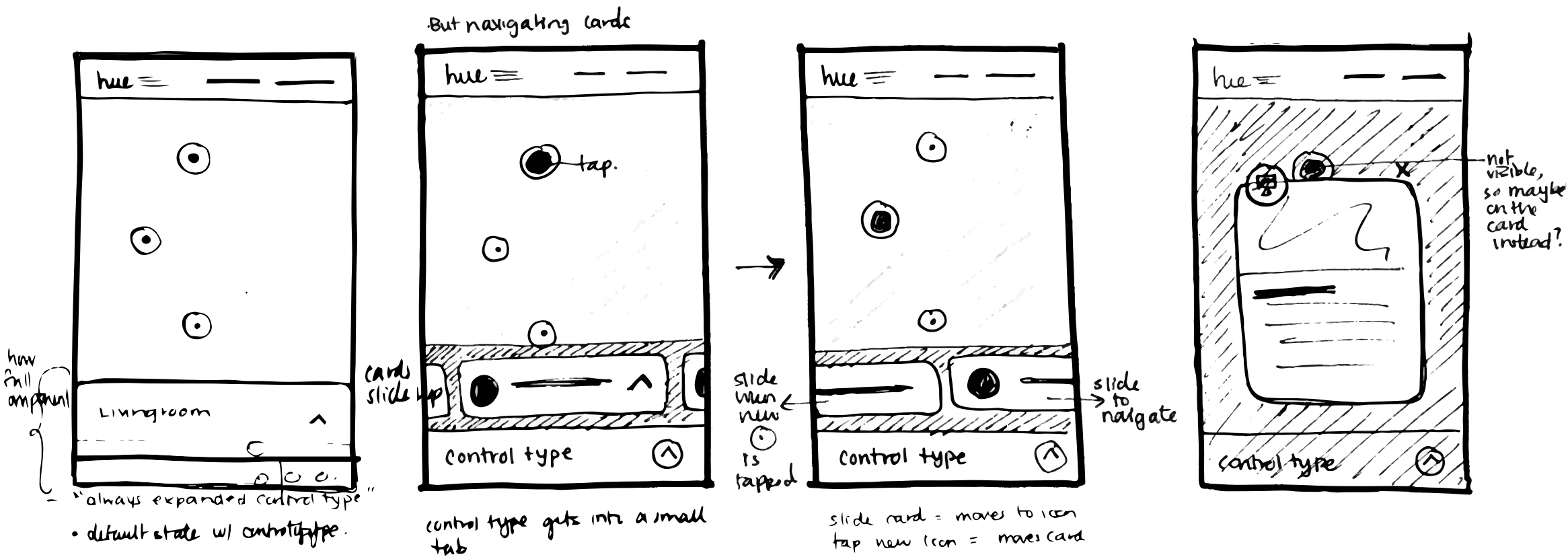
In particular, there were many factors that influenced the design of the Hue App feature control. For example to inform the user on where they were, the name of the room is displayed instead of labeling it with the control feature. In addition, through strategizing the content per room, we were able to focus the control panel to one feature per room. Thus, any additional unnecessary features of the existing pattern were stripped to keep the focus clear.



Prioritizing Interactions

Another area in question was how we can encourage users to navigate around the page and learn at the same time. All while ensuring moments of delight in that exploration. This introduced using icons and info tabs as a way to navigate across the page. When the info tabs appear, the control bar can afford to become smaller and of less priority to what the user wants to interact with at that moment.

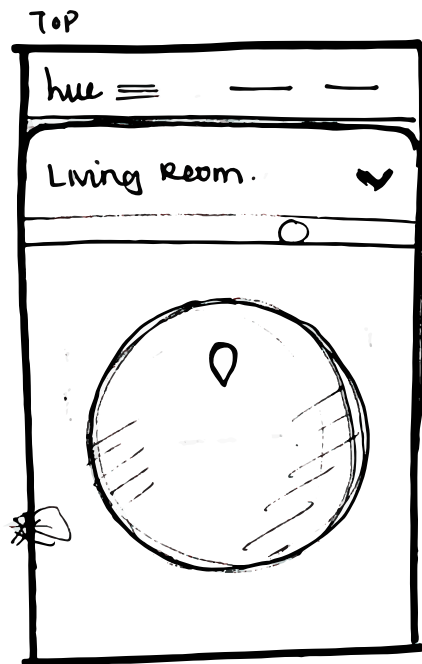
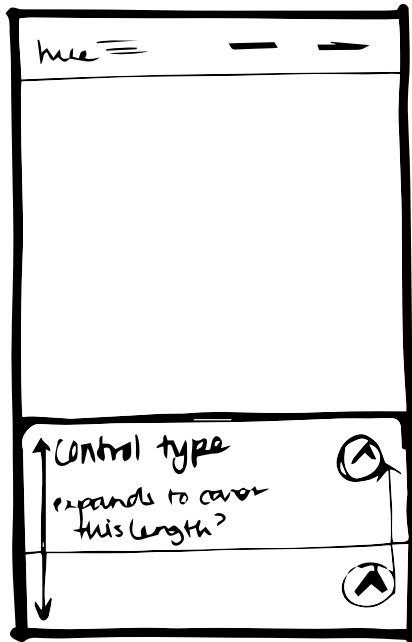
This led to the decision of concepting three sizes of the control bar, depending on what the user was interacting with. At default landing, the user would see the minimum to engage with the control bar, at maximum the control bar would expand just enough to display the type of Hue control, and when the user is learning the control bar is inactive. During default or inactive states, the user will always be able to expand to full control.



Real Estate and Constraints

To benefit the user's experience, intentional adaptations were made to the Hue App design patterns, like the different control types (colour picker, light scene selections). There's a difference between interacting with the Hue App UI to change the physical light, versus, interacting with the UI on mobile-web, all of which must be represented on screen. Where one has immediate feedback of the light changing around them, the other must fit and communicate that same delight on a small, mobile screen.

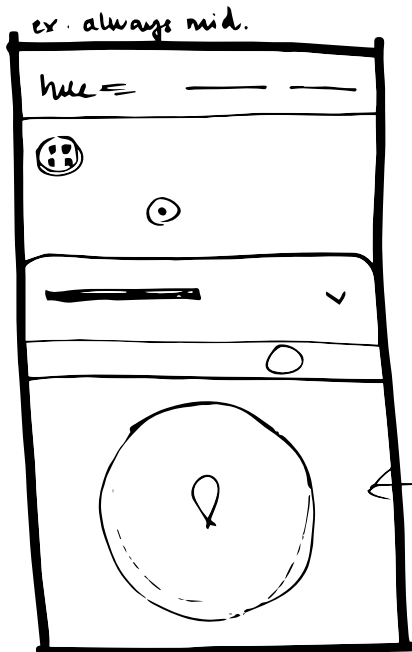
So the decision was to get rid of the black-opacity overlays (that was bringing the control bar to the foreground in one iteration) and shrink the control bar to expand only part way up. The essential Hue App interactions could still be afforded, while solving the real estate constraints. These decisions allowed for the light experiences to come through the UI and give users enough feedback of the lighting changes they were making.



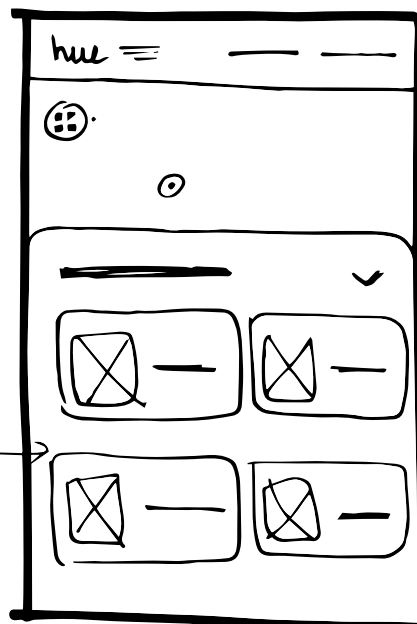
- expanded color wheel.
- choose color, then play with dimming



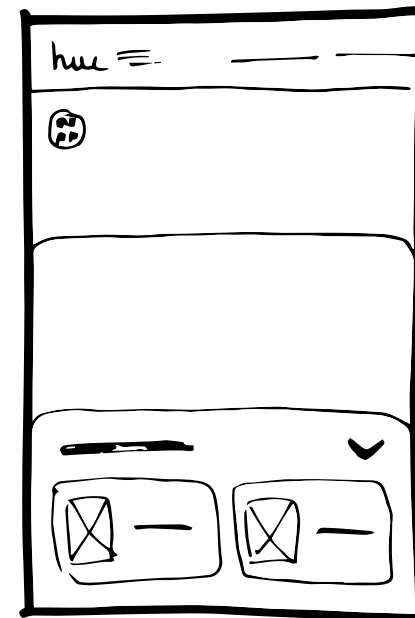
- because with fullscreen it's still visible what is happening.
- vs. fullscreen in app → physical light changes. on screen you don't have the same effect.



color picker



recipes.



⊗ just small ?
— for dan / n. h. k

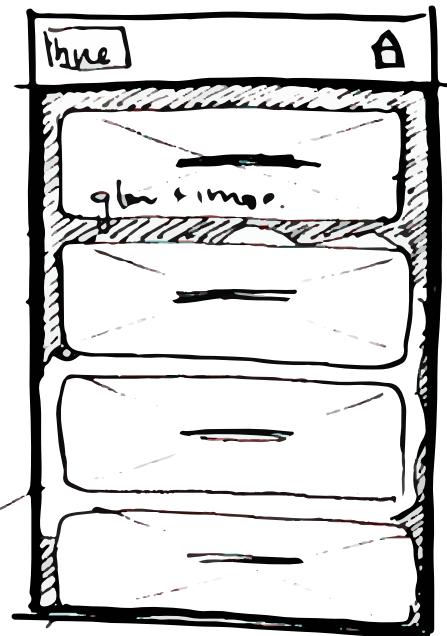
Onboarding and Room Navigation

Experience Hue in itself is an onboarding opportunity for customer to familiarize themselves with the Hue app. Which is why aligning the UI is so important to meeting their expectations should they come across the application in the future. The current design did not afford a smooth onboarding transition into the immersive experience, let alone an explanation to what the users would expect to interact with next.

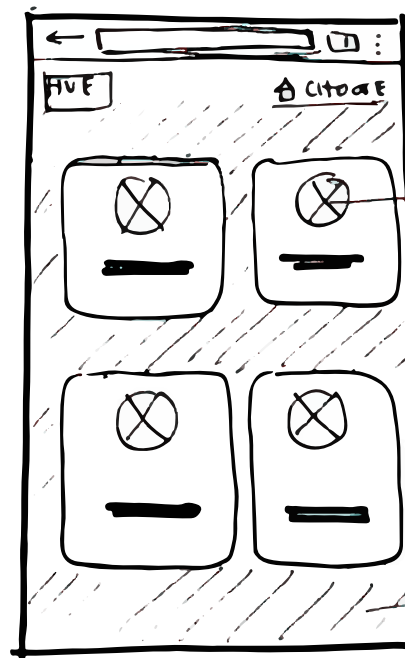
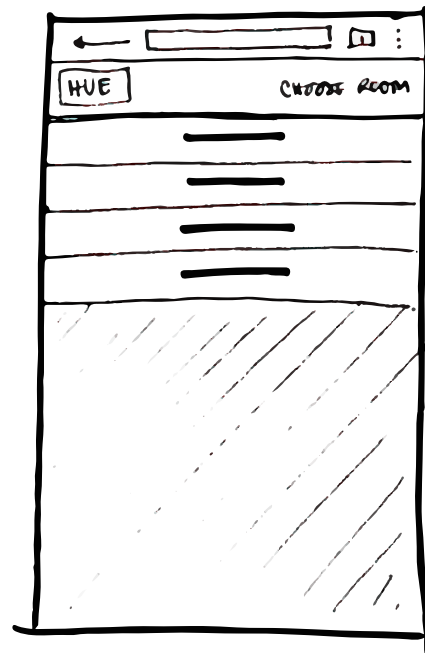
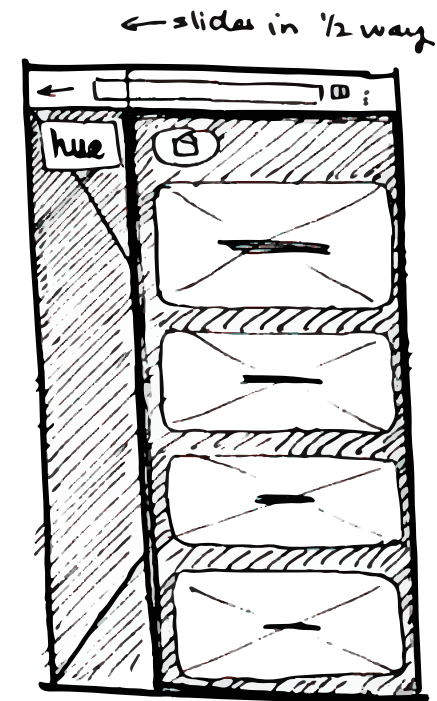
In the proposed landing page, we introduced a selection of rooms that users can choose to enter. I iterated on different ways to communicate the navigation options while keeping in mind the importance of consistency and accessibility on both the landing page, and when users are in the immersive experience and want to explore the next room.



- panels. → expand.



- per cards + buttons.
- component moves in



- overlay?
- card grounded?

