

Case study 1 - Uplinked

Role: Driver, Product management, Product design

Timeline: 3 weeks

Uplinked

Web

Admin experience

Guard experience

Mobile

Admin experience

Guard experience

Exploration

WONDER Problem + Impact

Improve the digital experience for Guards and Admins.

Overview

Improving the experience for Guards and Admins where they can easily use the mobile and web application designed.

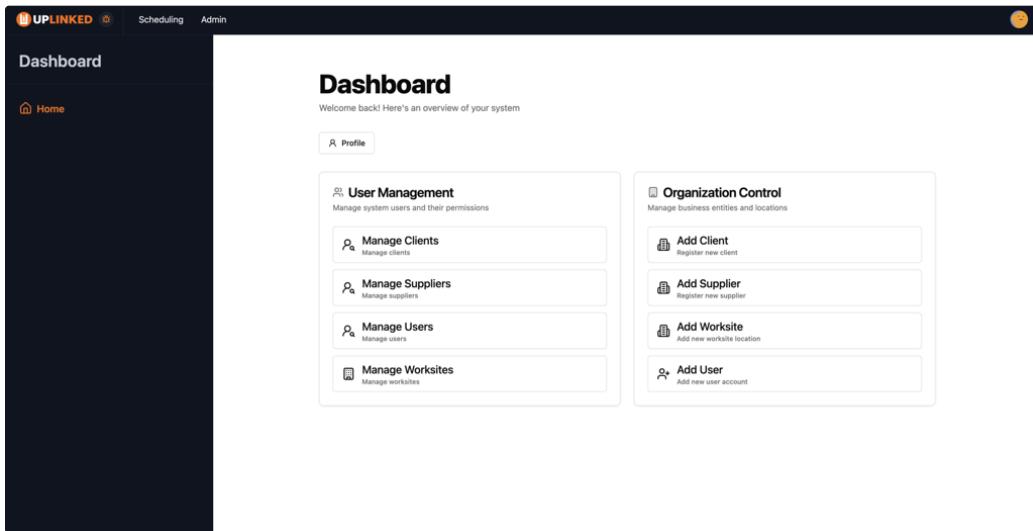
The current space

Uplinked had been built by a full-stack developer and was engineering lead so there's a lack of design direction and opinionated experience.

There are two different experiences set up for Admins and for Guards.

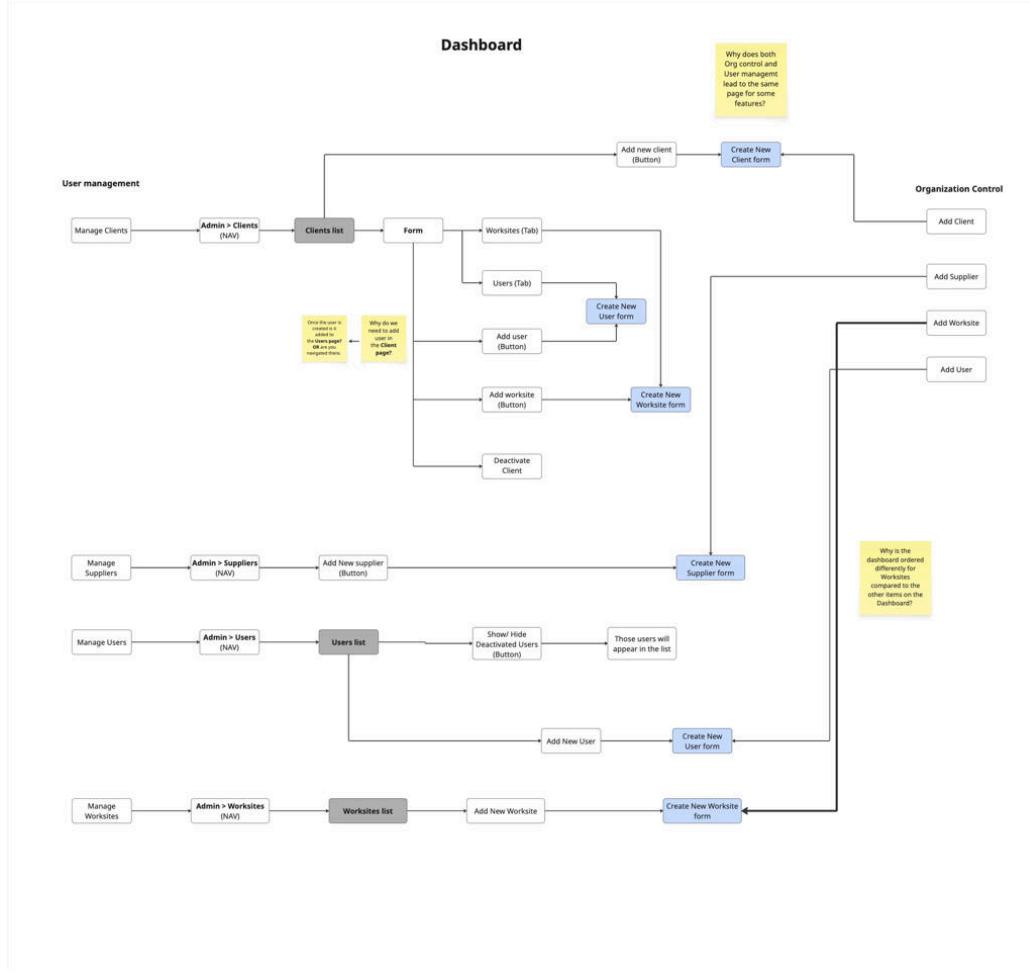
For the Admin experience on website

The existing dashboard and home page navigation is a bit confusing.



Current experience in the home page/ Dashboard

There isn't a difference between User Management and Organization Control except the Organization control are shortcut keys to



Current web experience for Admins

Web (Office)

Admin experience - Manage guards, clients, suppliers and guard shifts.

Dashboard

Welcome back! Here's an overview of your system

The dashboard interface includes a navigation bar with a search icon and the text "Profile". Below this are two main sections: "User Management" and "Organization Control".

User Management: Manage system users and their permissions. Sub-options include:

- Manage Clients (Manage clients)
- Manage Suppliers (Manage suppliers)
- Manage Users (Manage users)
- Manage Worksites (Manage worksites)

Organization Control: Manage business entities and locations. Sub-options include:

- Add Client (Register new client)
- Add Supplier (Register new supplier)
- Add Worksite (Add new worksite location)
- Add User (Add new user account)

Mobile

(onsite)

Admin experience

- Personalized dashboard → No. Shifts that need to be assigned, No. Client requests, No. Supplier requests
- Shortcut actions → Assign shifts, Client requests, supplier requests.

Roles and responsibilities

- Arrange and manage guards as well as the guard shifts
- Connect with clients, suppliers.

Guard experience

- Personalized dashboard → when my next shift is (no. of upcoming hours of shift), No. of hours of shift completed.
- Status of Guard - on shift, not on shift.
- Shortcut actions → Edit time unavailable for shifts, Shift reminders notification on/off.

Roles and responsibilities

- Attend shifts
- Welfare check-ins during shifts

- Licenses are up-to-date

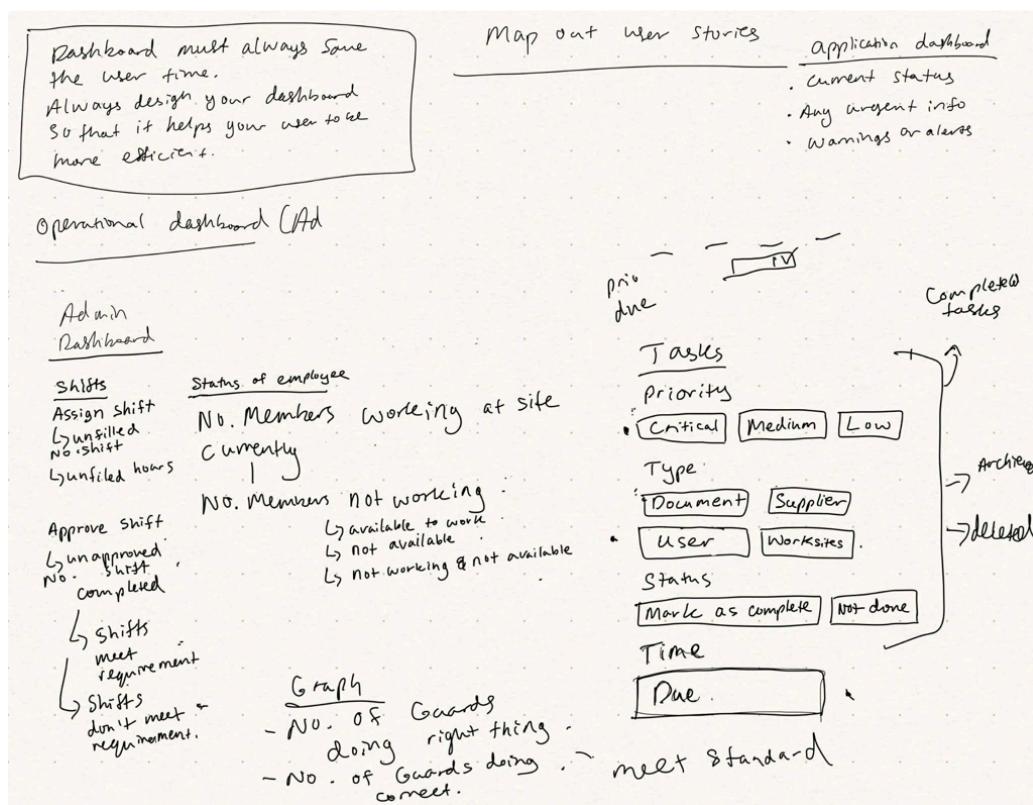
For shifts, the Guard needs to go on X number of patrols randomly. This is done by using NFC tags.

- Each shift will have waypoints that will be scanned. Required number of scans for each waypoint depending on the shift.

EXPLORE Design thinking combined with research/ analysis (Design strategy)

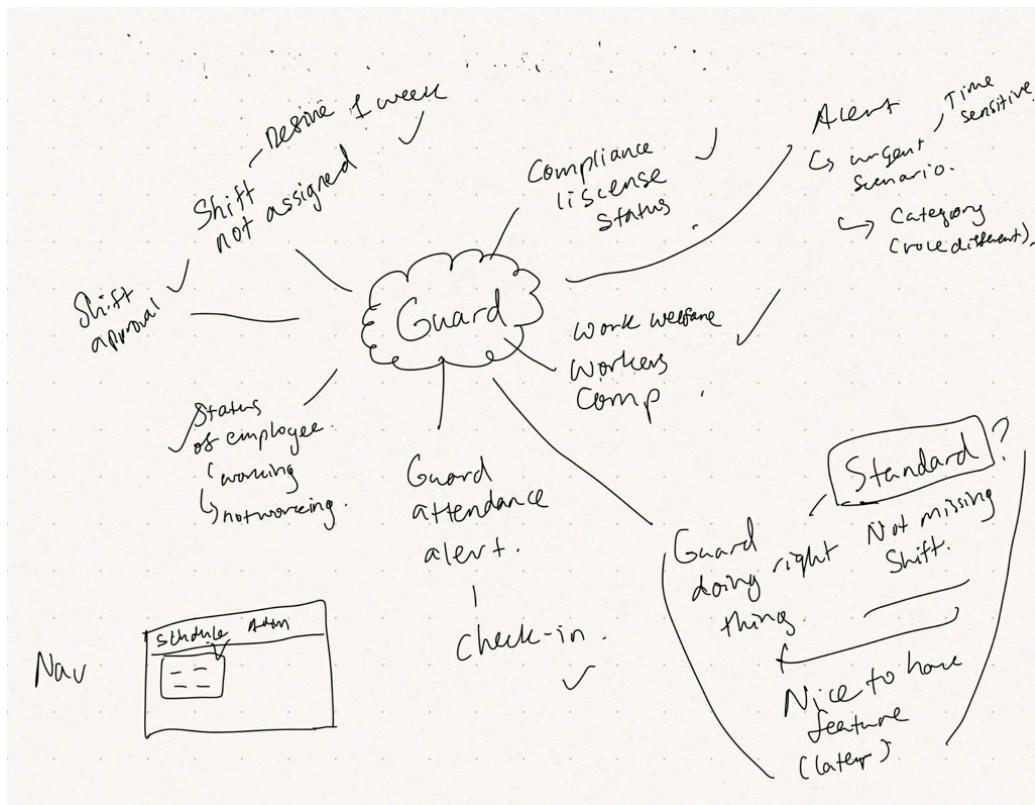
Website admin dashboard

Conversations with the Client and mapping out all the tasks an admin would need to do to try and understand the use case. Initially I proposed a tasks list then realized everything could potentially be a task so it's important to map out what everything does.



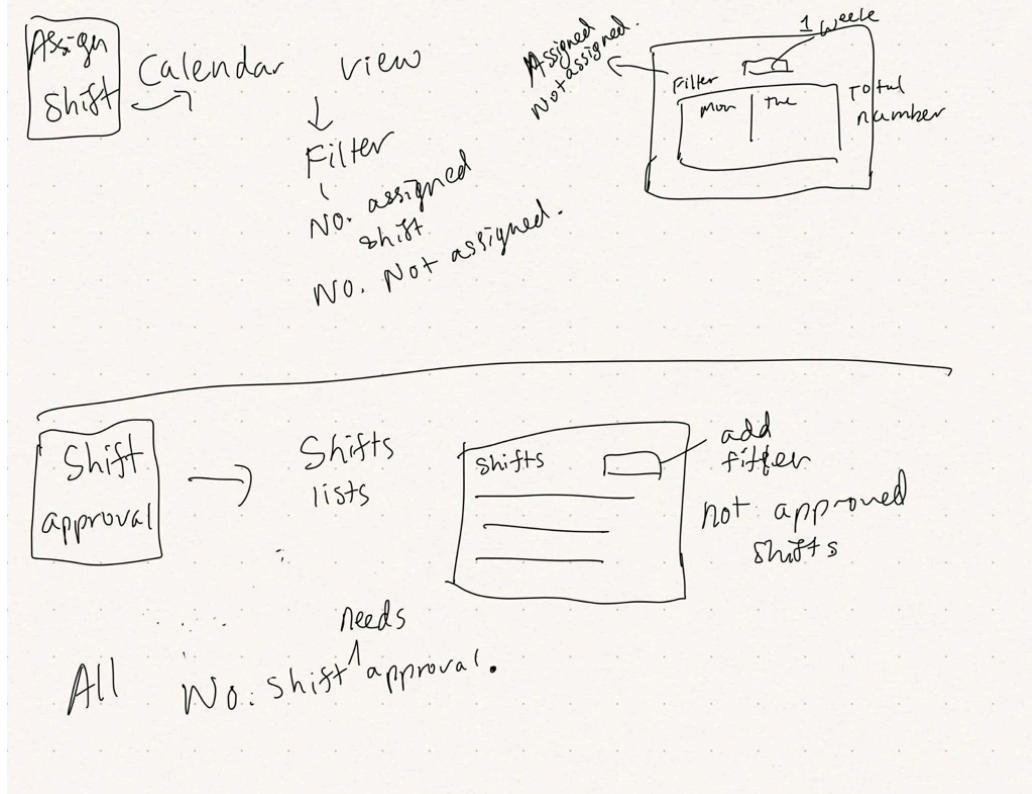
Notes on all the potential use cases

Following that I distilled down the information that needs to be on the dashboard. Also potential nice to have feature such as an overall understanding of Guards that are up to standard. I also simplified the requests into easier understandble concepts.

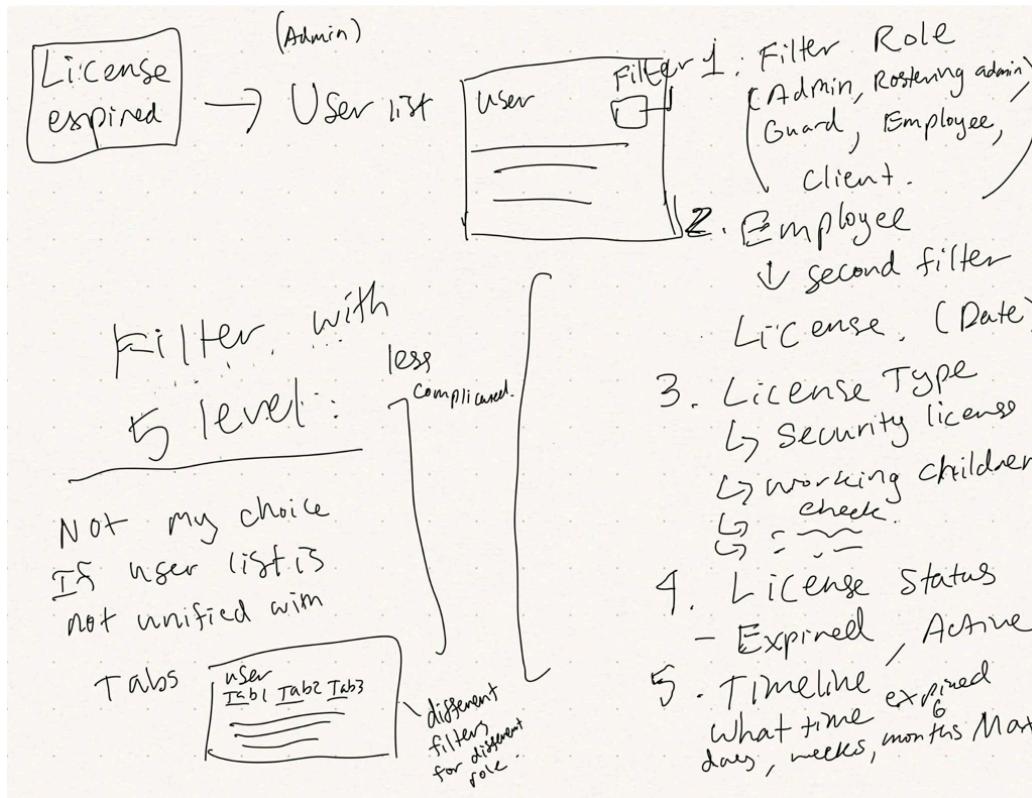


Distilled features

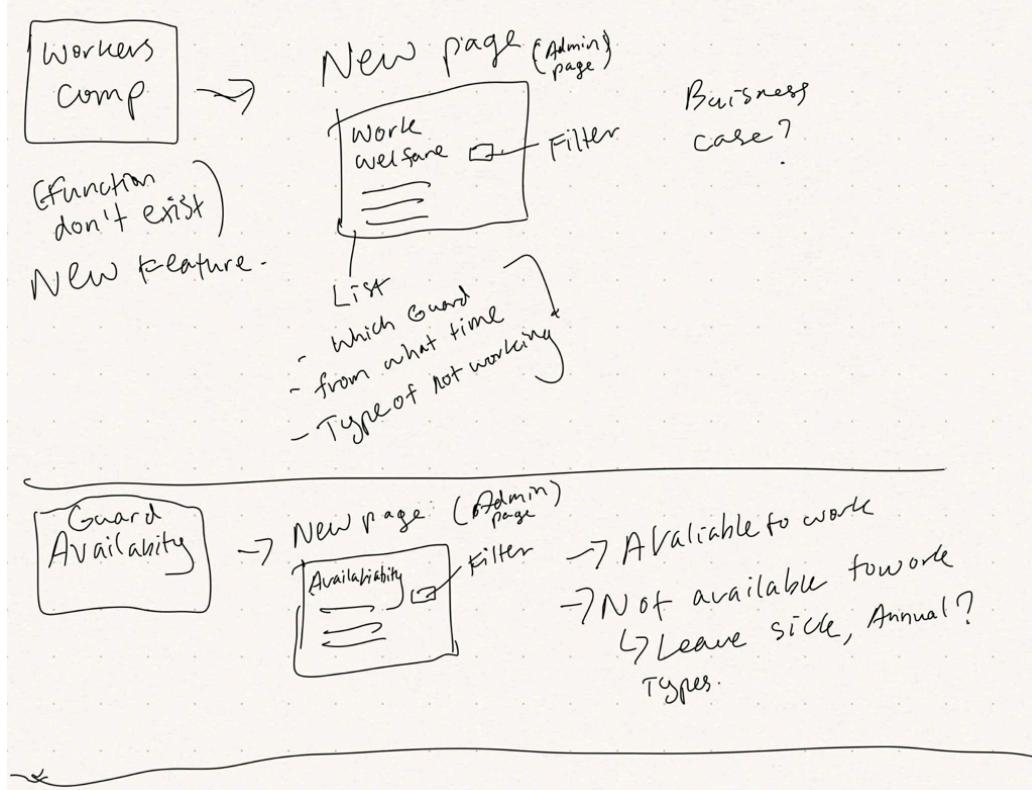
After mapping out all the requirements, I tried to understand each feature by mapping out each individual features.



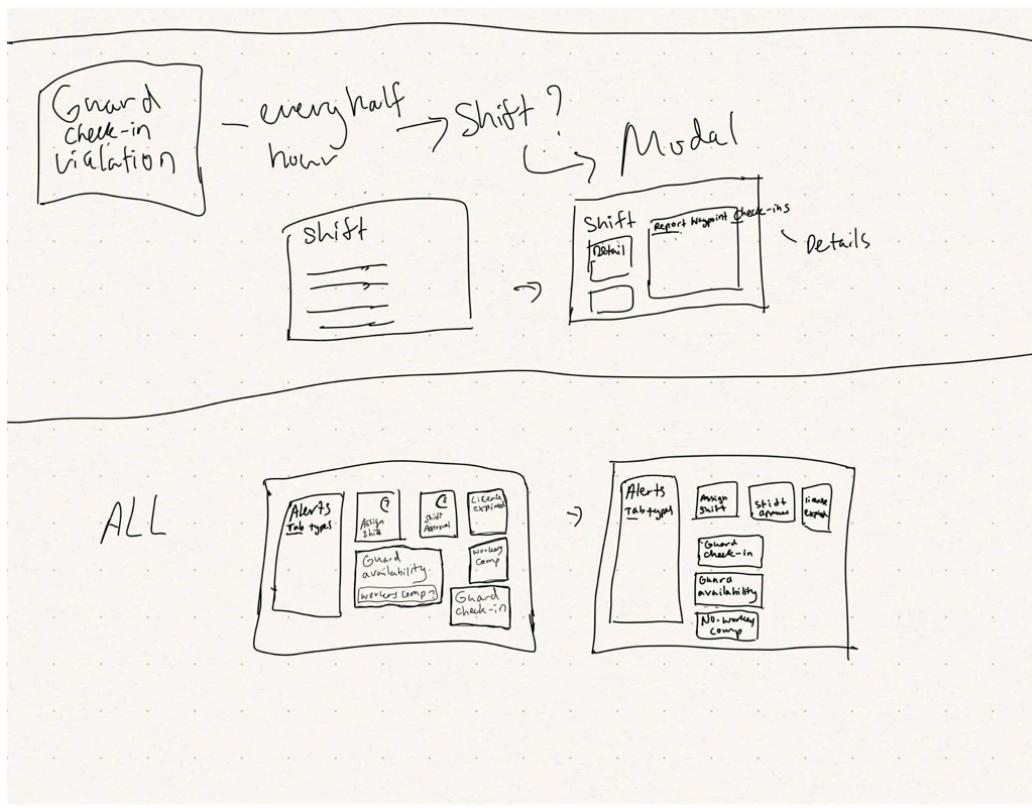
Assign shift and shift approval feature mapping



License expired mapping



Workers comp and Guard availability mapping



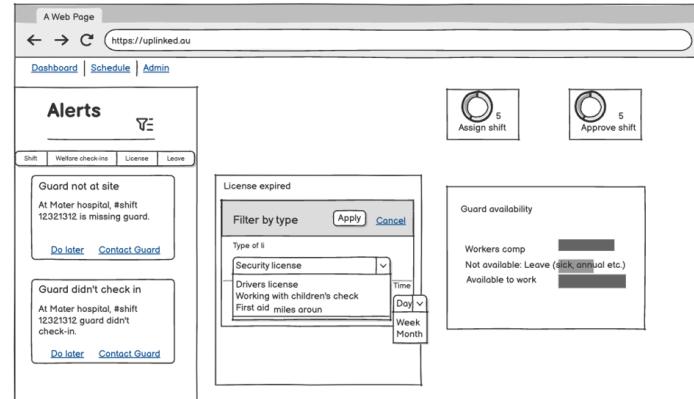
Guard check-in validation and some mockups on the dashboard

Initially I was planning to just design the dashboard but then realized I needed to have a good understanding of the use case and also connect the experience. So I started mapping out areas where the existing website was missing and solutions where we can solve that.

MAKE The process (wireframes, mockups, prototypes-user flow)

Wireframes

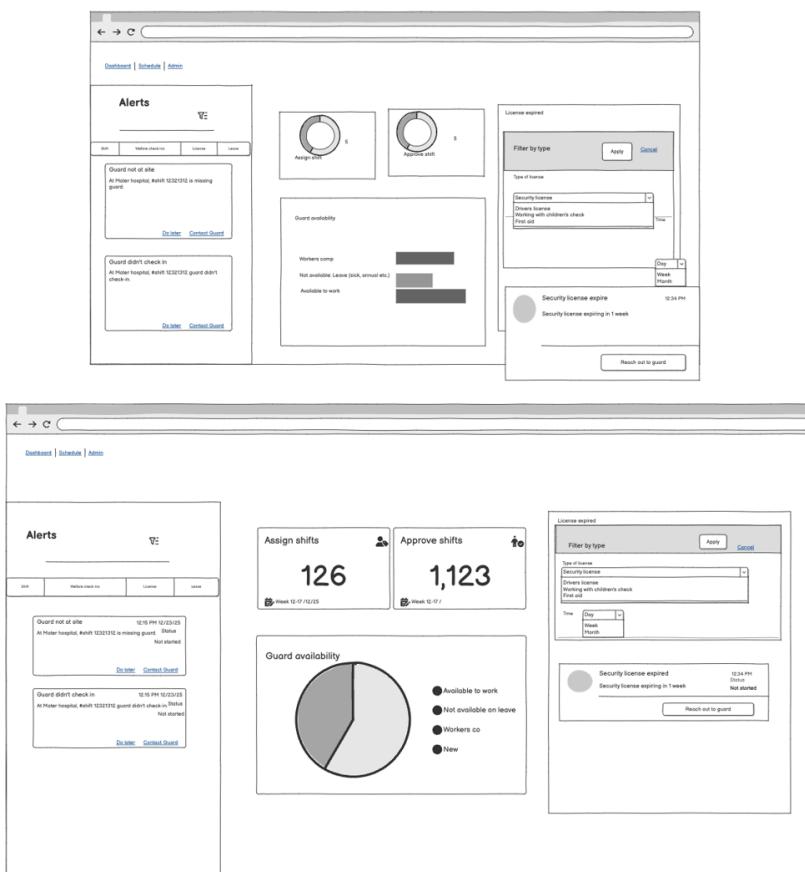
Version one of all features but messy



Version two starting to define hierarchy



Version two starting to define hierarchy



Testing out using different data visualization methods to show the information

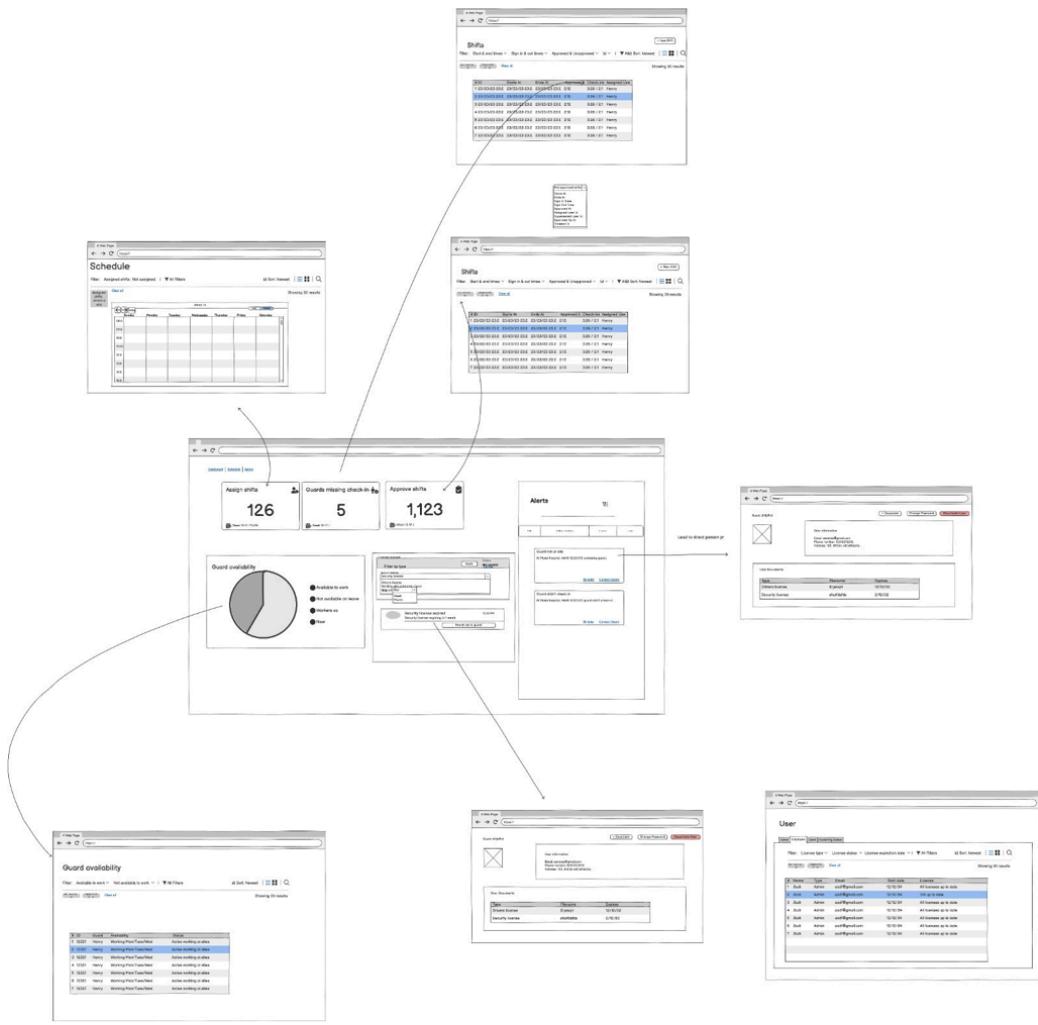
From skimming and comparing the two data visualization methods the second one would be easier to understand. It also gives a better understanding of what is happening in the current time.



Testing different layouts

Mockups

Conceptual flow



The overall flow and navigation

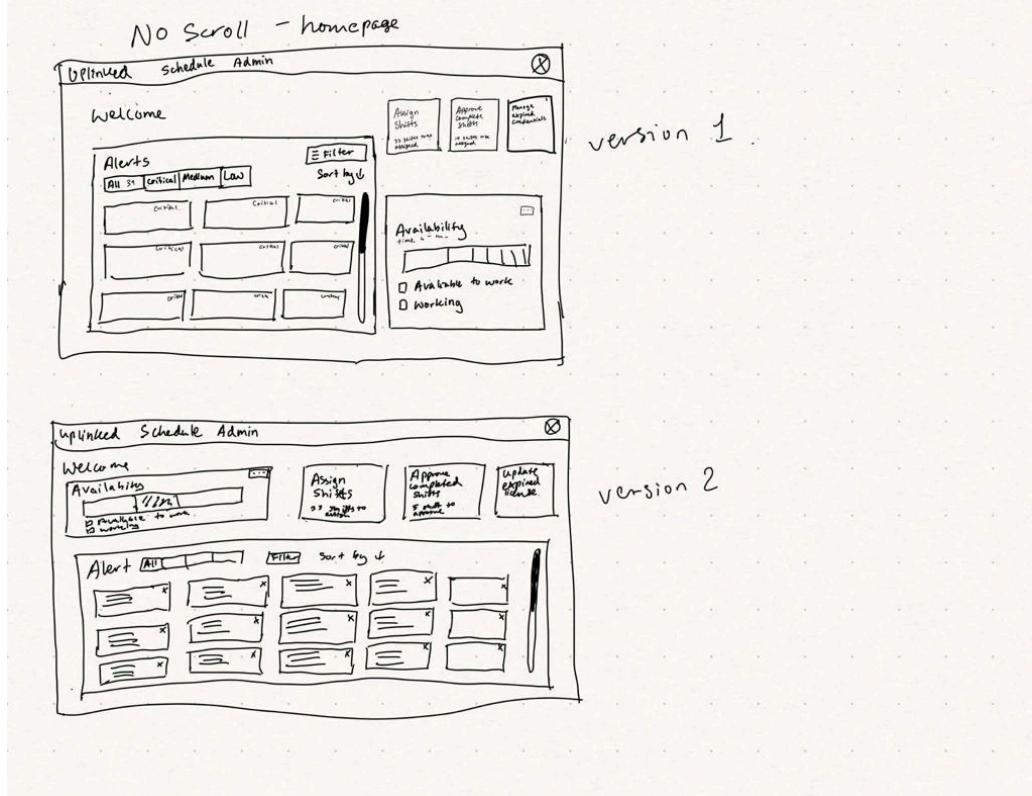
Iteration version 1 for super admin dashboard on website.

This is iteration version 1

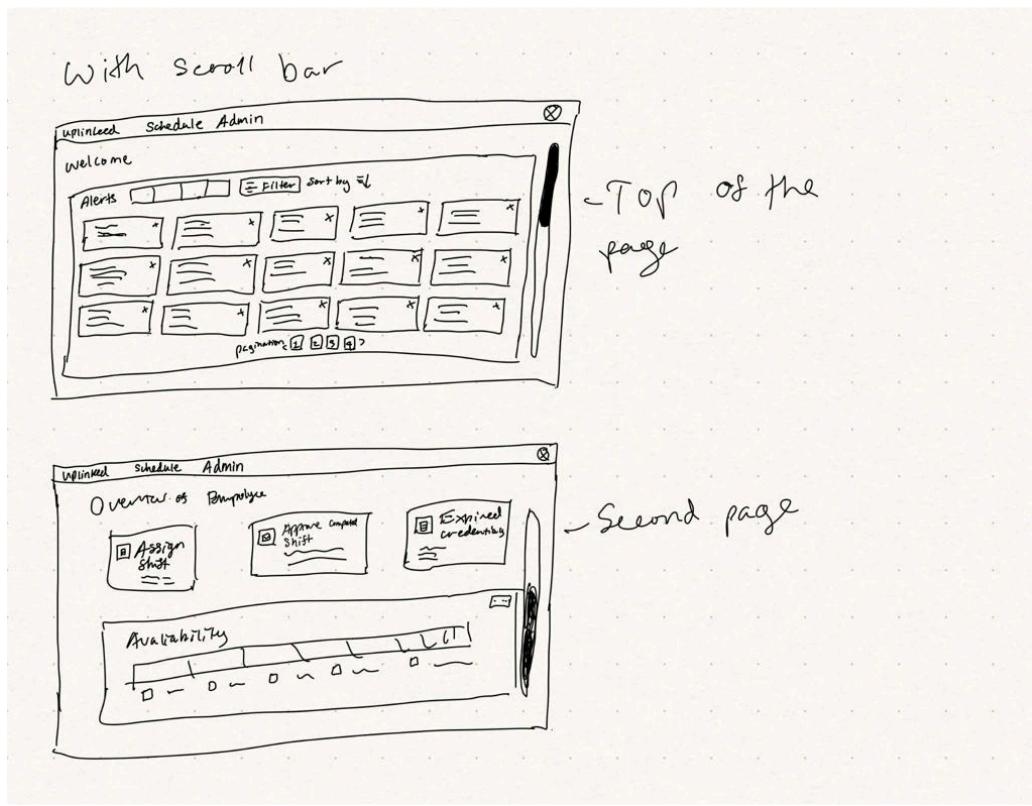
There are some improvements to be made for this feature such as the “Welcome admin name” card is not clearly defined. The select week option is not obvious for guard work availability, unassigned shifts information and shifts waiting approval.

There's also too much information in the home dashboard screen so I'm exploring other ways to showcase that.

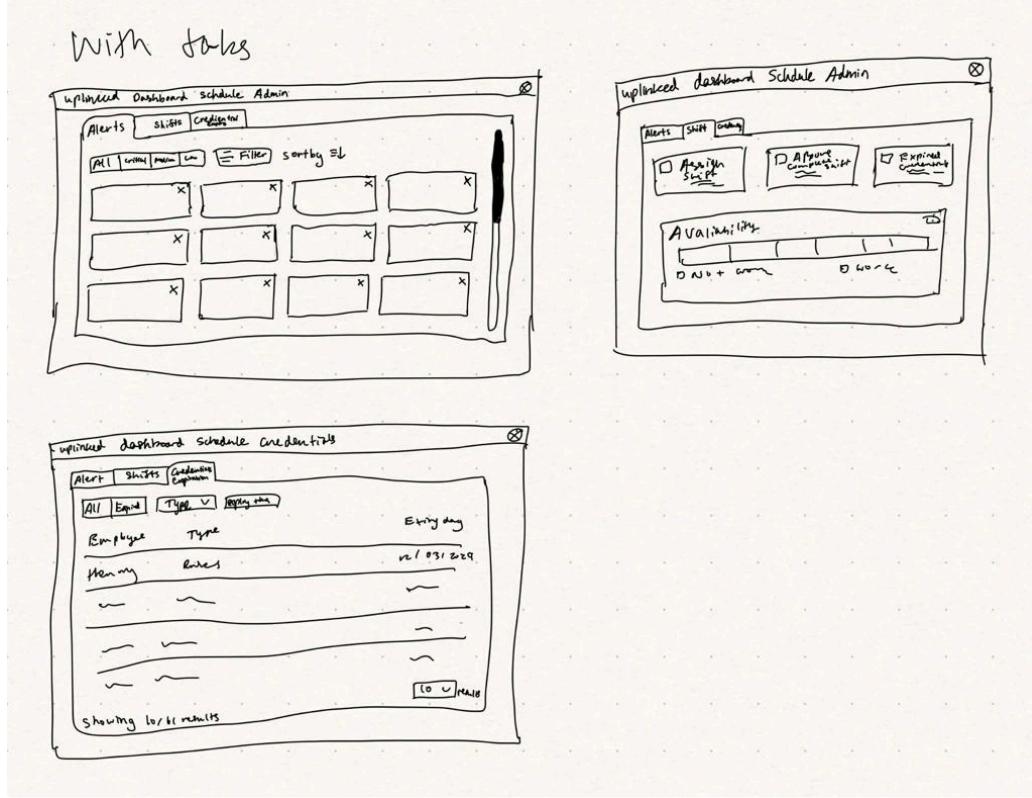
To simplify the dashboard for users I went and drew up three different options including varying layout as well, no scroll option, scroll option and using tabs.



No scroll including layout options



Scroll bar



Tab option

After looking at all of these options and exploring design principles.

The best option is optimized 'No scroll' considering the different design principles and the elements.

The scroll option once we add additional features it would be harder to scale and users may struggle with the need to scroll to find more information. The tab option would have a similar struggle as users need to press the tabs to explore what options are available as well.

Optimized 'No Scroll' - Version 1 by incorporating one crucial element: **internal scrolling**.

A strict "no scroll" page can break if you have too much content (e.g., 50 alerts). The solution is to allow the *modules themselves* to scroll internally while the overall page layout remains static.

1. **Use the Layout of Version 1:** Keep the main structure with the "Alerts" module on the left and the "Availability" and "Action Buttons" on the right.
2. **Make the Alerts Module Scrollable:** The "Alerts" section should be a container with its own scrollbar. This way, if there are 20 critical alerts, the admin can scroll through them without losing sight of the availability chart or the action buttons.

3. Keep Action Buttons Visible: The **Assign**, **Approve**, and **Update** buttons should always be visible, as they are the primary call-to-actions.

This hybrid approach creates the "at-a-glance" benefit of a static dashboard while gracefully handling a variable amount of data.

Quick mock-up of the version 1 no scroll option

The mock-up shows a dashboard interface with the following components:

- Header:** UPLINKED, Schedule, Admin.
- Dashboard Section:** A summary area with three cards:
 - 21 Unsigned shifts (10 Aug - 15 Aug) with an **Assign** button.
 - 5 Shifts waiting approval (10 Aug - 15 Aug) with an **Approve** button.
 - 2 credentials expiring (In 1 week) with an **Update** button.
- Alerts Section:** A grid of six alert cards, each with a close button (X).
 - SOS triggered (Guard Henry Smith triggered at Shift ID 123232, Mater Hospital) with a **Contact guard** button.
 - Guard not at site (Guard Mason Wright not at shift: Shift ID 123123; Mater hospital) with a **Contact Guard** button.
 - Security license expiring in 3 days (Guard Bradley Hall Security license expiring) with a **Review guard profile** button.
 - Security license expiring in 3 days (Guard Bradley Hall Security license expiring) with a **Review Guard profile** button.
 - Missed welfare check (Guard Mason Wright missed welfare check: Shift ID 123123, 10-12-32, Mater hospital) with a **Contact Guard** button.
- Guard availability Section:** A chart titled "Guard availability" showing the current state of employees from 10 Aug - 15 Aug. The chart indicates 39 Not available to work, 9 Sick, 10 Annual leave, 20 Workers compensation, and 61 Available to work.

Version 1 arrangement

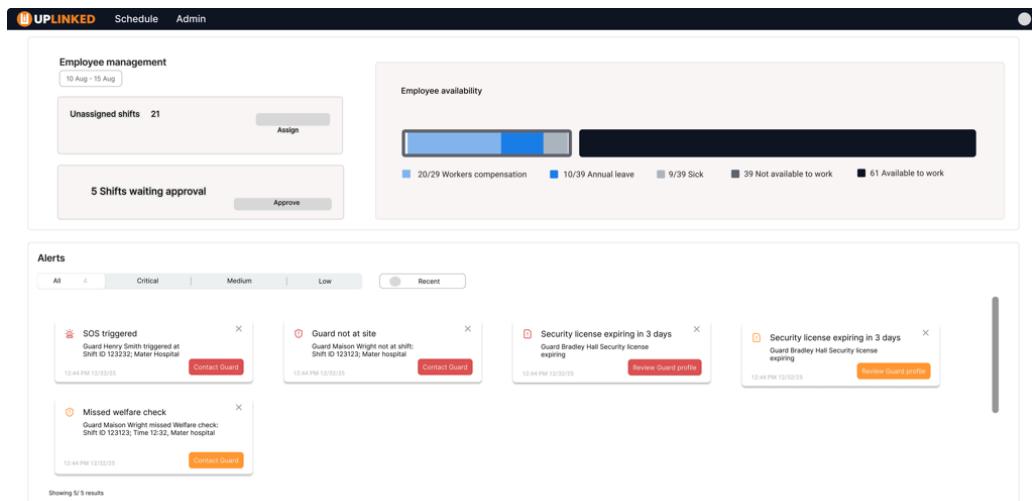
Disregarding the spacing, colour and overall UI conceptually, this version isn't the most suitable design.

I tested out this arrangement, and it's not working in the use case, as there's unnecessary information after testing with the engineer on the Dashboard. The Credential expiring card is a duplicate of information as we already have an alerts section which can highlight these important issues.

Rough dashboard version 2 design rough

Pieced everything together, there's some thinking required in the Employee management in terms of the smaller action cards and how everything comes together.

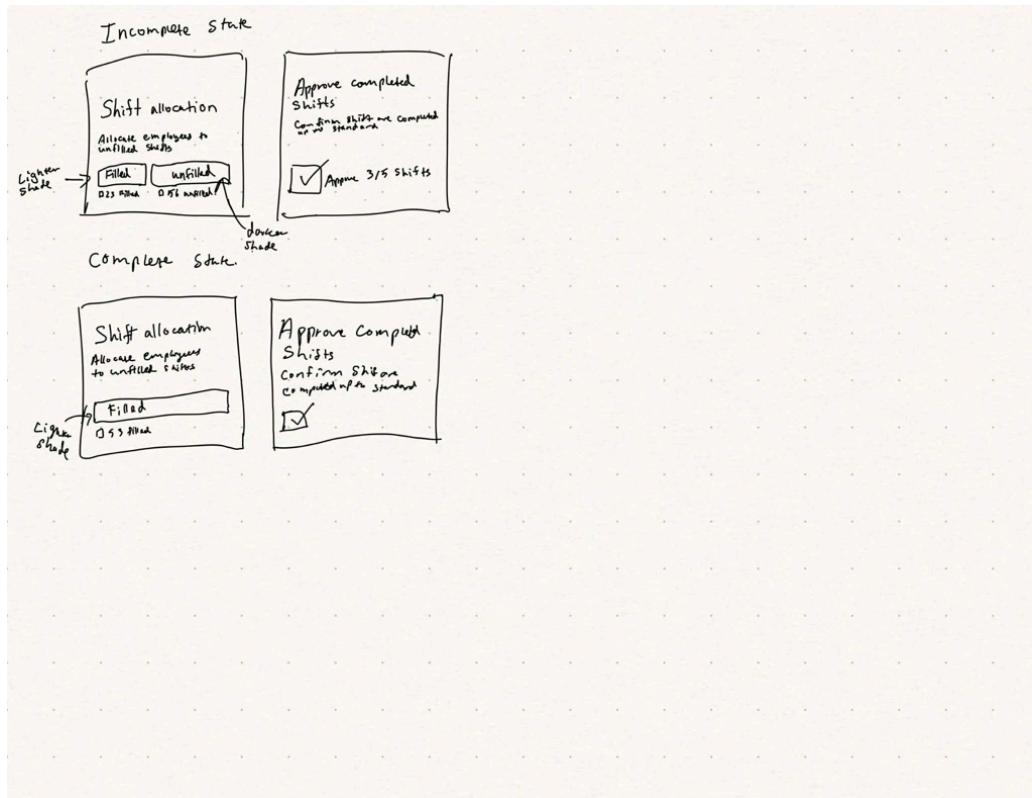
The alert cards also need to be thought through in the way the cards display the information whether it's enough information and what happens when there's longer text.



Medium fidelity dashboard website

After some design reviews of the rough drafts

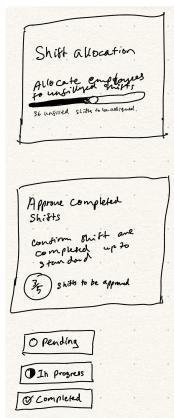
I went back to the drawing board to reimagine the arrangements and create a high-fidelity of the designs.



Card designs drawings version 1

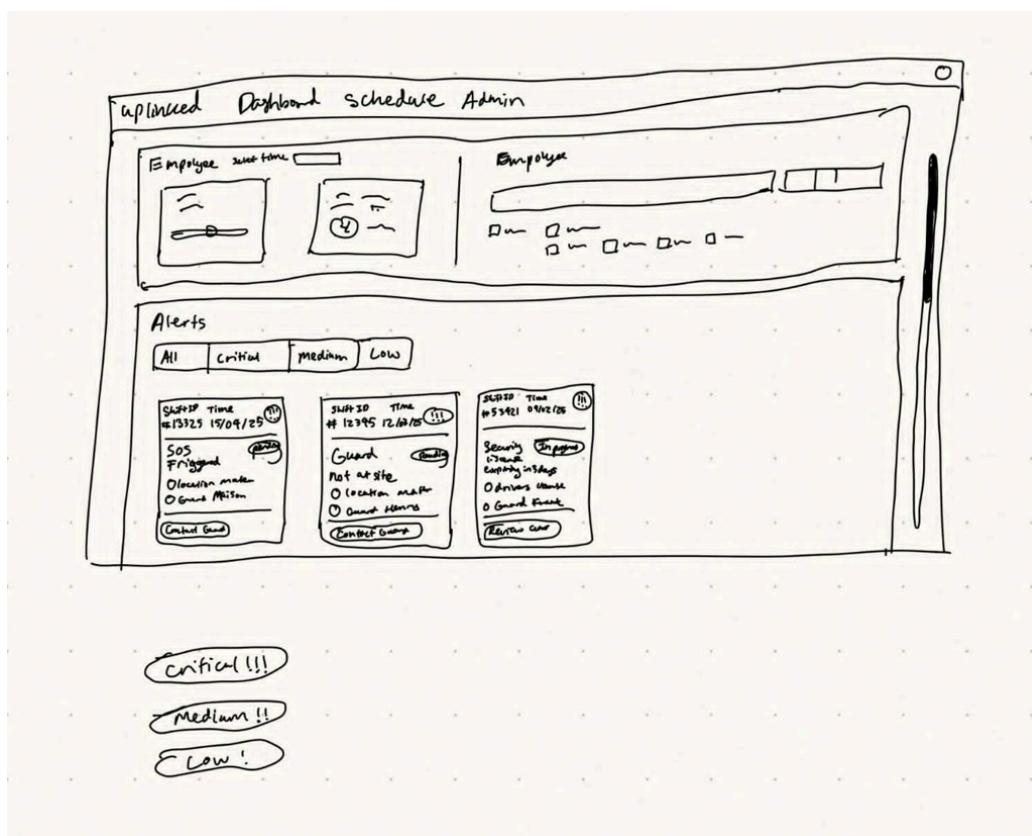
After reviewing the card design drawings version 1 the data visualisation isn't as straight forwards I decided to go and try to simplified it.

Redrew how to display the information which led to the card design drawings for version 2.

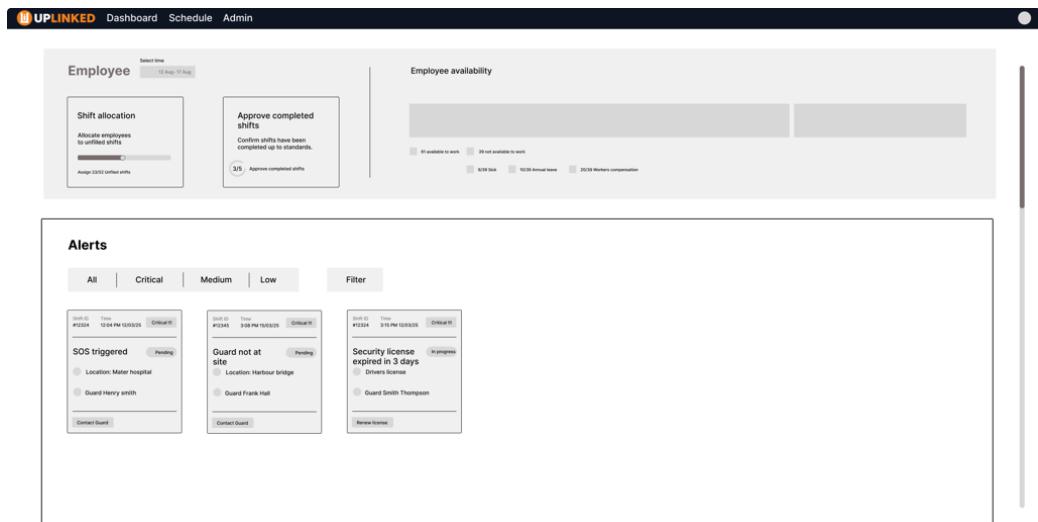


Card design
drawings
version 2

I drew a more higher fidelity version of the dashboard together with the card designs and alerts.

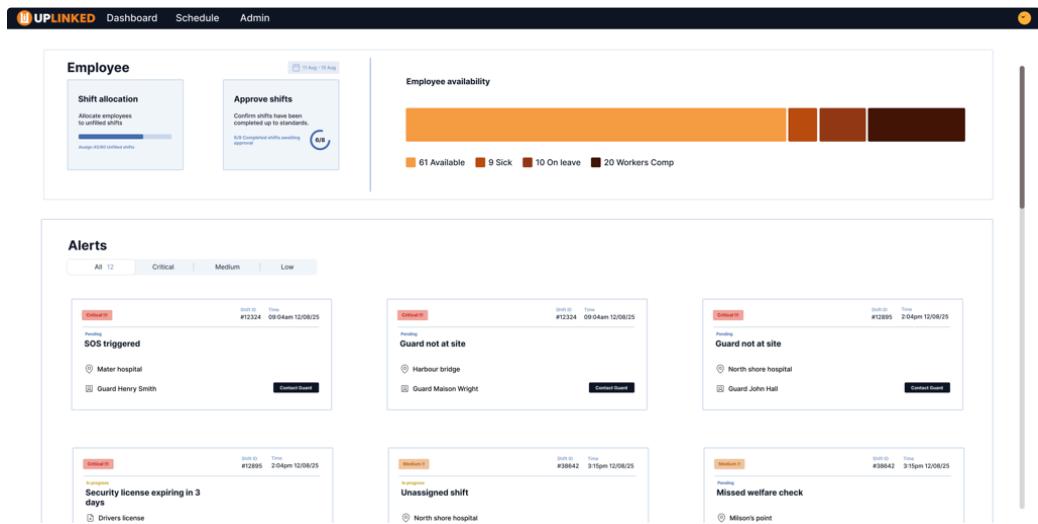


Higher fidelity drawing of card designs



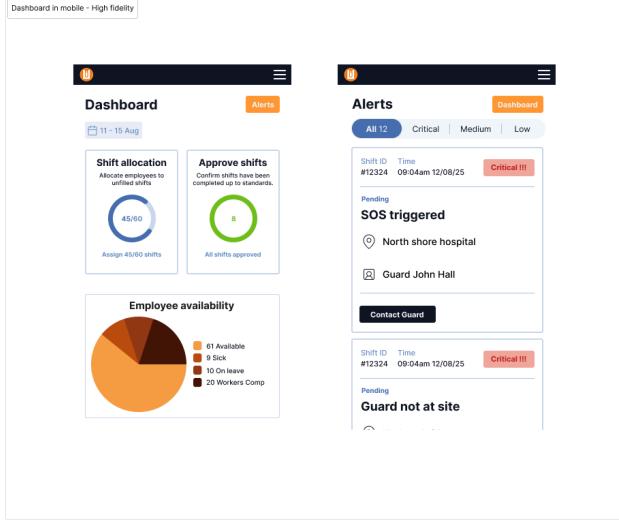
Low fidelity dashboard web

From Low-fidelity I incorporated the usage of colour and added the colour scheme of orange and blue and used different shades of that.



Dashboard high-fidelity website version

I also created a mobile version of the High-fidelity designs as there are mobile versions for the dashboard too.



Dashboard high-fidelity mobile version