In 2014 University of Melbourne launched its first iteration of a University-wide intranet and the first of a series of themed hubs. The goal for this project was to consolidate a number of key University staff-only web resources under a unified information architecture and to present staff-only information and communications in a secure, consistent and user-focused way.

**Staff Hub / University of Melbourne**

**MY ROLE:**
Working with a diverse and changing team and stakeholders, I was involved in many aspects – including concept pitching (developing a series of ‘straw men’ mock-ups), html wireframes and prototypes, and responsive /mobile-first template development in Squiz Matrix CMS on a Bootstrap framework. I also helped with importing and editing content.

**TOOLS USED:**
- Illustrator / Acrobat (wireframes and functional specifications)
- Fireworks (straw men)
- Bootstrap (HTML prototype / template framework)
- Squiz Matrix CMS
- HTML, CSS, jQuery
- Photoshop (image editing)
- Excel
- Basecamp (team / stakeholder comms)
- Jira agile (team comms and project tracking)
- Git (site back-ups, basic versioning)
These pages in particular underwent a lot of design and content iterations even after they were developed in the CMS, as more and more people became interested in the project and contributed new feedback to be integrated before launch. This required a continually flexible and agile approach and a commitment to quality and attention to detail in an environment of fast paced flux.

My work on these pages included:

- The base responsive design / layout
- Designing and developing a variety of banner areas which could integrate images, video and text
- CMS asset builders so that staff with minimal web training could maintain / update them
- Designing and developing link asset types and grouped asset lists that were also easy to populate and manage by staff with minimal web experience
- Adjusting these somewhat complex CMS structures quickly and frequently in response to senior stakeholder feedback whilst maintaining some systematic code / build integrity
My work in this page included:

- a responsive interface design
- integrating a variety of public and private RSS feeds via the CMS and presenting them in a consistent manner
- integrating the events feed via API
- developing a variety of banner areas and CMS asset builders so that staff with minimal web training could maintain / update them
These are some early wireframes (at different stages), which featured a layer of basic functional specifications outlining where the content could be sourced from, and some of the behaviours. These were for the benefit of a variety of stakeholders (project managers, senior stakeholders, content managers, fellow developers). 10 page types were modelled. More detailed interaction design prototypes were built into a basic Bootstrap-based HTML site at a later stage.
Early concepts featured quite a busy and buzzing dashboard look, accommodating a lot of information up front, which was a popular style with many of our users and stakeholders. There were approximately 15 pages developed which illustrated some possibilities for the presentation of complex functionality and integration of information from a variety of other systems into a new holistic visual system and user experience. These straw men helped to get the project approved to progress into the next phase.
A later version of the concept prototypes illustrated some of the possibilities for integrating existing intranet content into the Staff Hub and how we might maintain a consistent and intuitive navigation pattern across a complex structure of subsites.
A LATER VISUAL CONCEPT IN THE SERIES OF ‘STRAW MEN’ CONCEPT PROTOTYPES

This is version 6 in the series of ‘straw men’. It responds to feedback and input we had been receiving from new sources and illustrates:

• An integration of some cross-branding with the student portal (same but different background image)
• Alternative navigation patterns
• Plans for responsive design, including use of both wide and narrow screens
This is version 8 in the series of ‘straw men’. It responds to feedback and input we had been receiving from new sources and illustrates:

- Utilisation of very wide screens
- A responsive card view pattern that fills the screen with content no matter the screen size
- Utilised horizontal scrolling and also allowed for vertical scrolling for more information on any topic.
- Introduces theme icons (eg. BIP, EB)
- Allows for more icon buttons to be presented (when debate about button prioritisation)

(Note: It looked a lot better and cleaner close-up on the screen)
This is the built version that went into beta and then live. It was received very well by staff and tested very well in the first rounds of user testing.

I worked on the base user interface and new responsive CMS templates (with new custom interim branding whilst new web branding was being developed in another team). These were developed on a Bootstrap framework and used FontAwesome icons (these were chosen for ease of maintenance/handover).

In addition to the base template, some of the features I worked on included:

- new custom content types and form interfaces to allow editors with no or little HTML or CMS experience to create and manage content - eg. link types, news types, news banners with and without images
- customised trigger emails to allow moderators a non-intrusive way to monitor changes to the site
- integration of RSS feeds, yammer and twitter widgets and an events API into the news and events pages.

As the University is a highly consultative environment, at every stage I needed to be very flexible and responsive to feedback.
Along with custom handover training (in person), this support site is provided to site administrators and maintainers inside their site folder and provides tools and instructions specific to maintaining their site as well as links into relevant information provided elsewhere by internal and external web communities to help them with their maintenance tasks.

Each support site is based on a standard base template so I can implement it quickly for each new project. It uses nested content areas for efficient reproduction, ease of updates and consistency of messaging, but is also easily customisable for each client or project and administrators can easily build upon it.

**Information architecture**

- User interface design
- Content
- CMS template build
- Asset builders / module development