# Software Projects Milestone 4

# Prototyping Completed

## Group Number: 24

## Concept Name: Scale-shelves

## Activity Planning

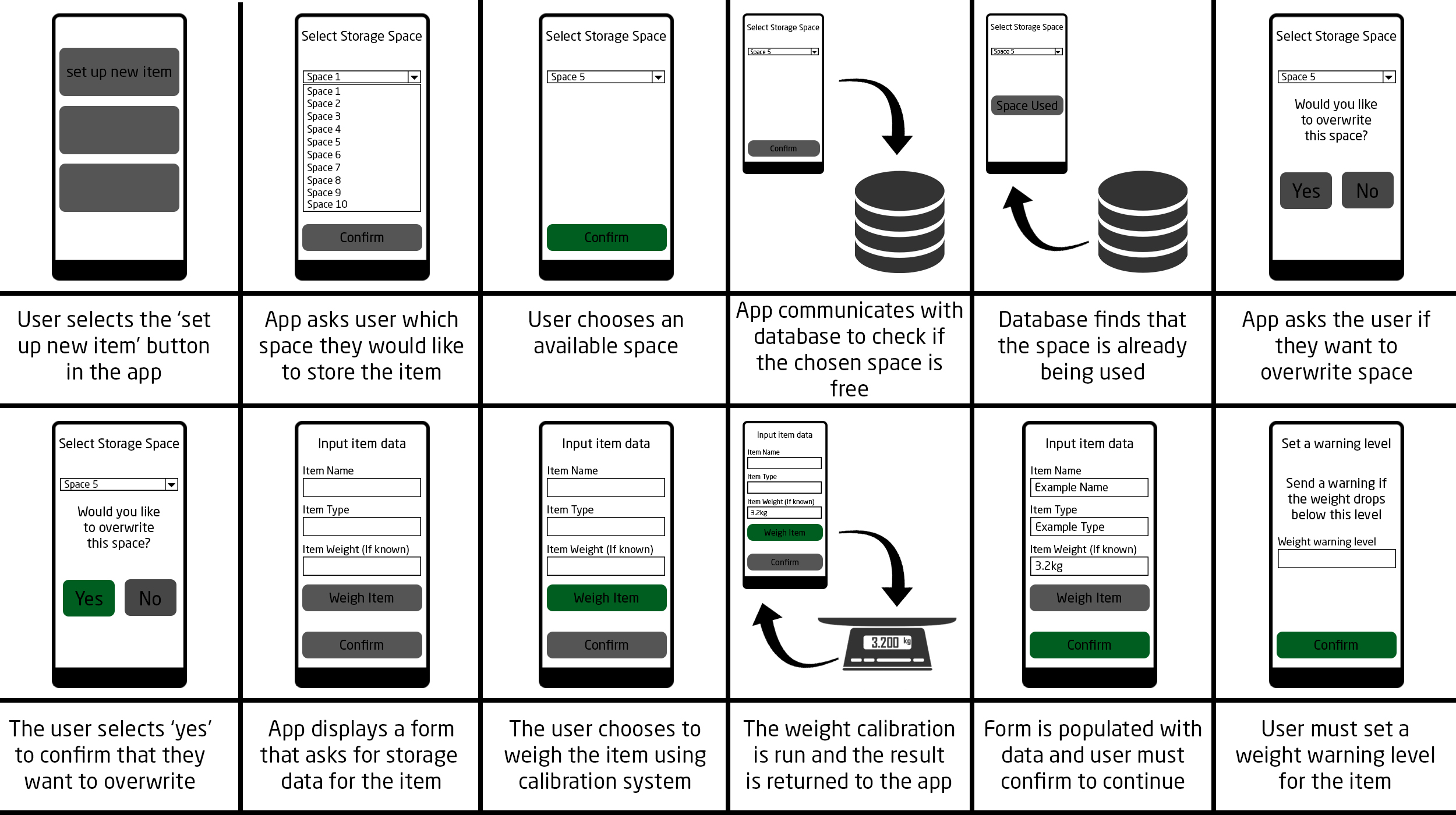
We split up the task of researching the prototype before reading week. Split up the task of making lo-fi prototypes over reading week and finishing hi-fi prototypes by the end of Sunday 17/11, below is a GANTT chart of how our time was planned to be spent (there were some minor changes).

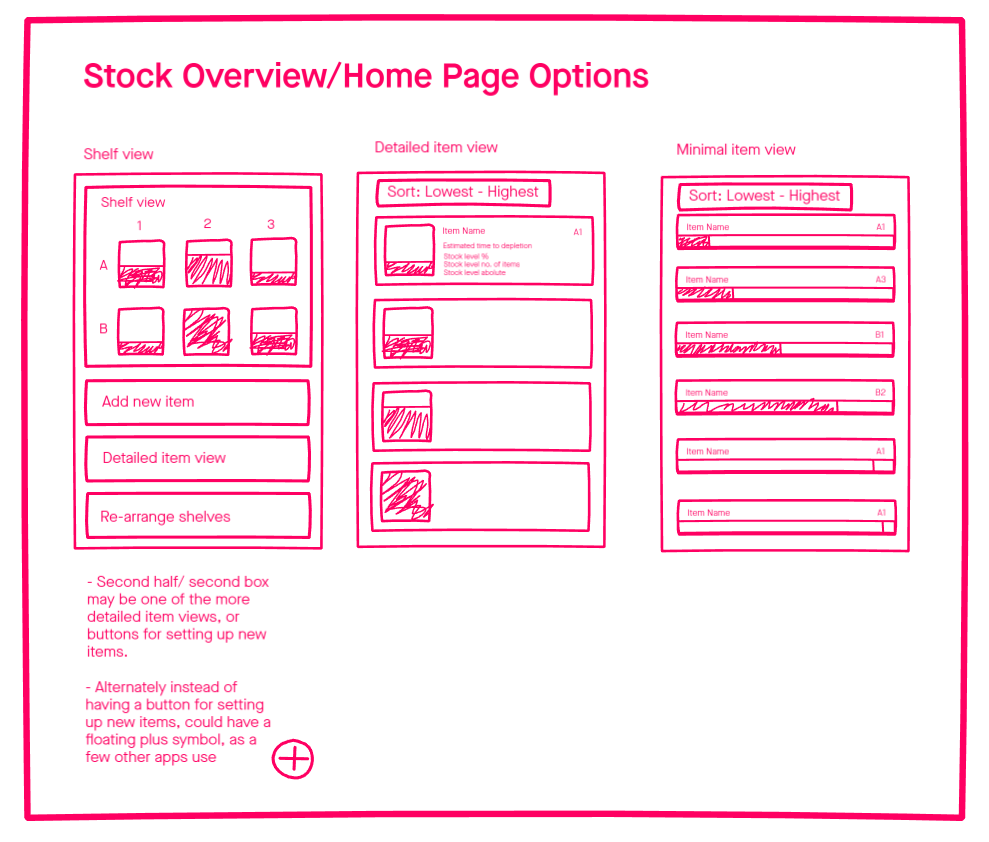
## 

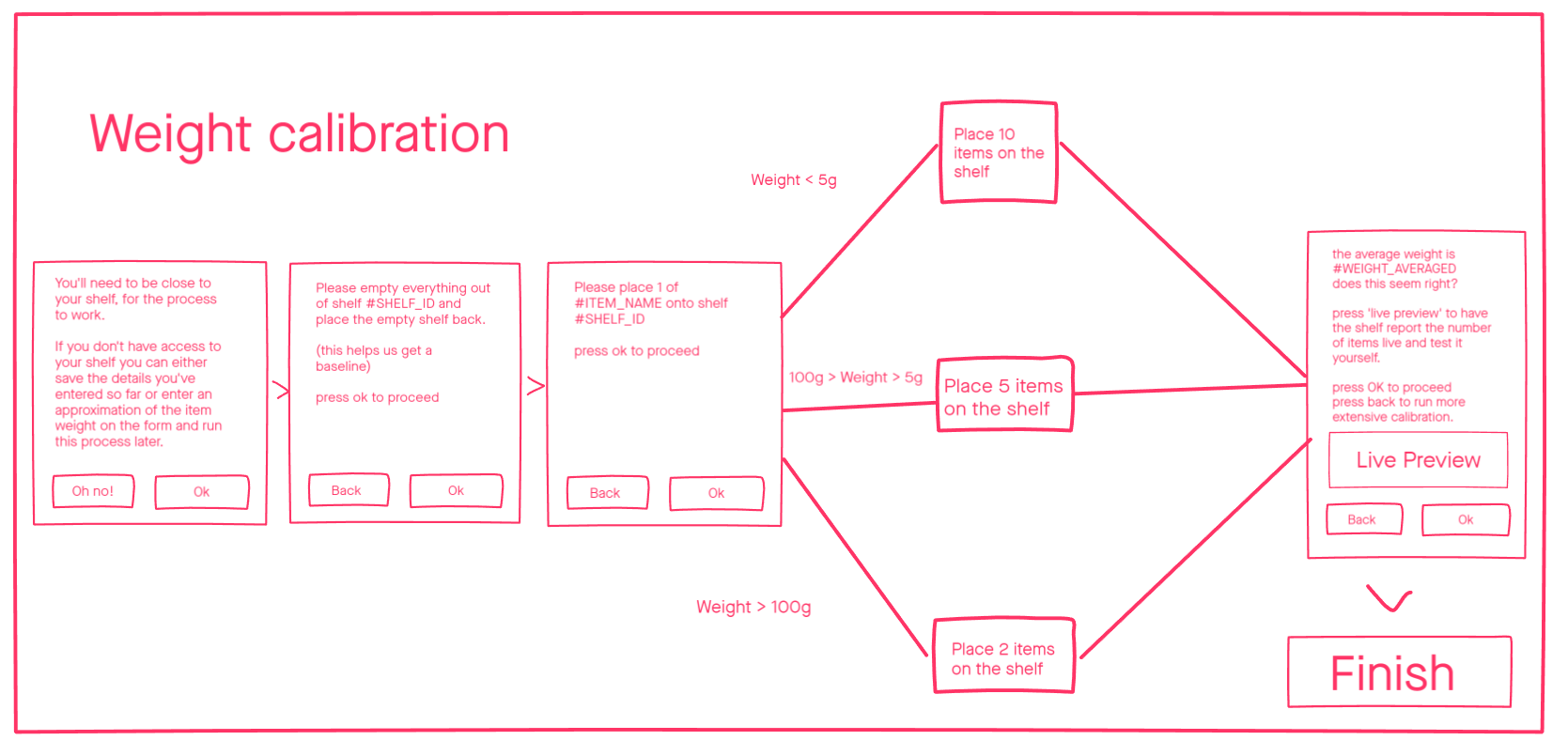
## Low-Fidelity Prototyping

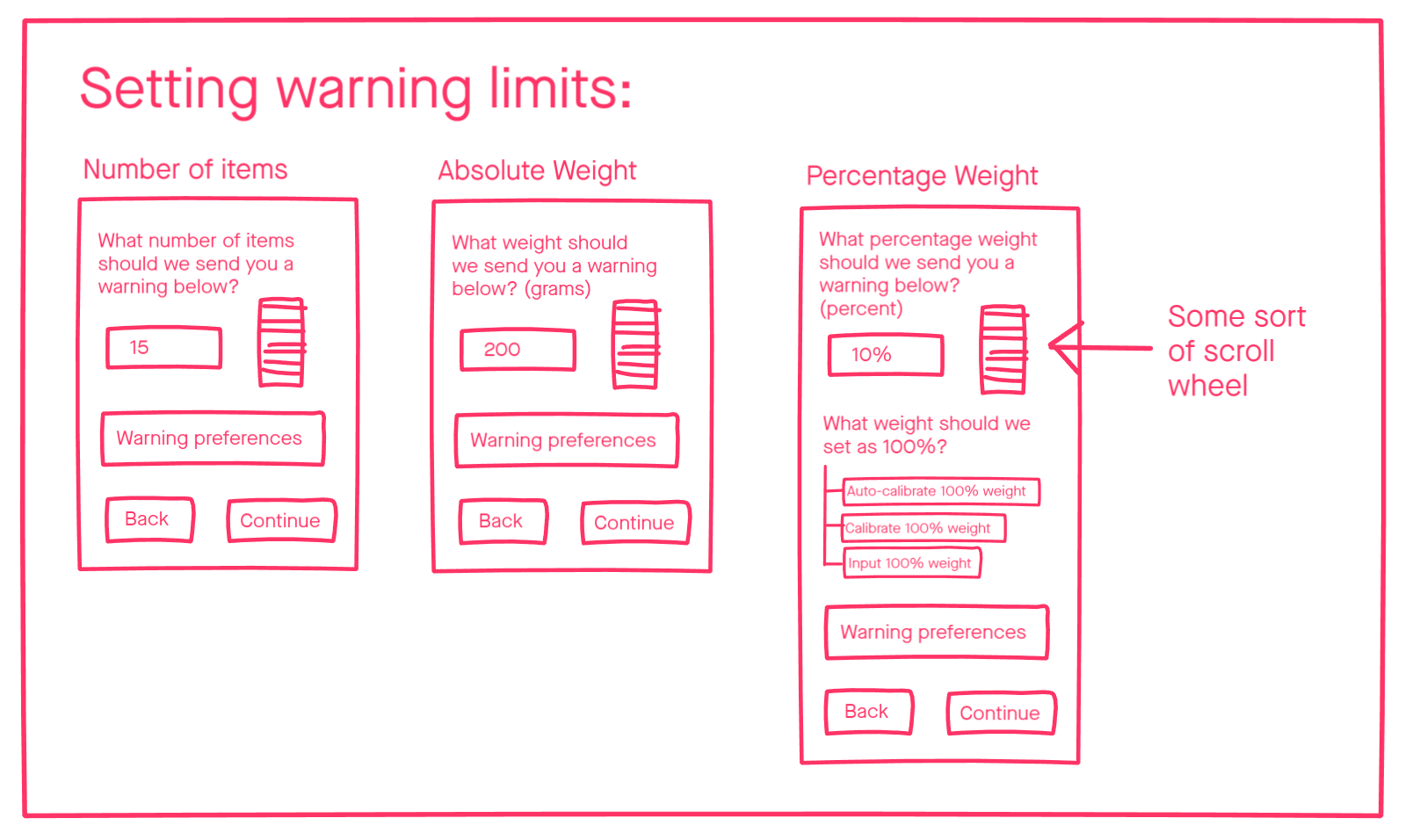
For our lo-fi prototype we made a storyboard version of the item setup process based our UML flow diagram. We also made some mock-ups of various screens and processes, so that some of the design work was done before moving to high fidelity.

Feedback was positive, generally they liked the idea of having several alternative views for the home screen rather than any one specific layout.









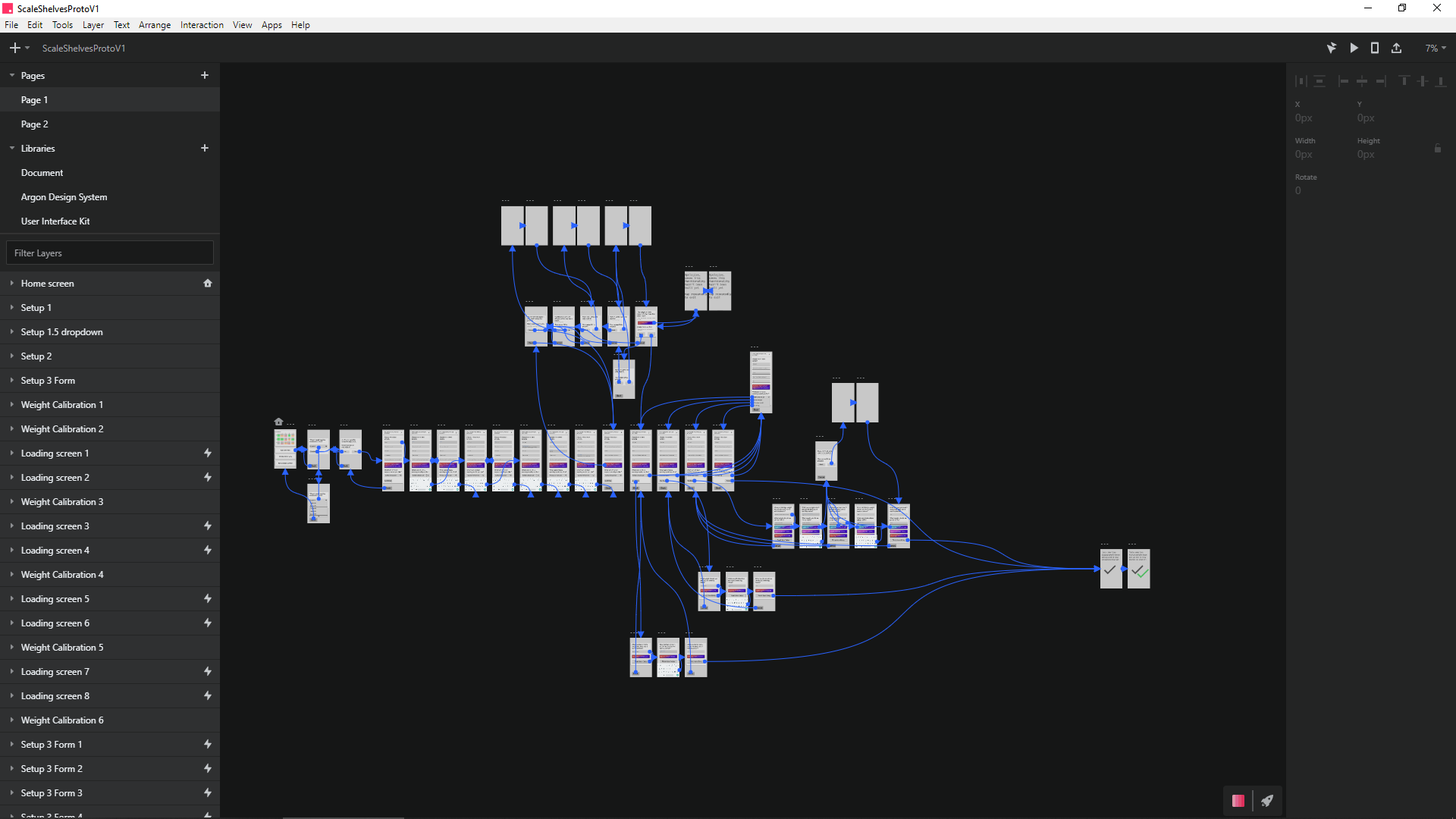
## High-Fidelity Functional Prototype

We constructed a high fidelity functional prototype using invision studio. We mainly focussed on modelling the item setup process. We also made some screens in other pieces of software to see the benefits of using different prototyping tools.

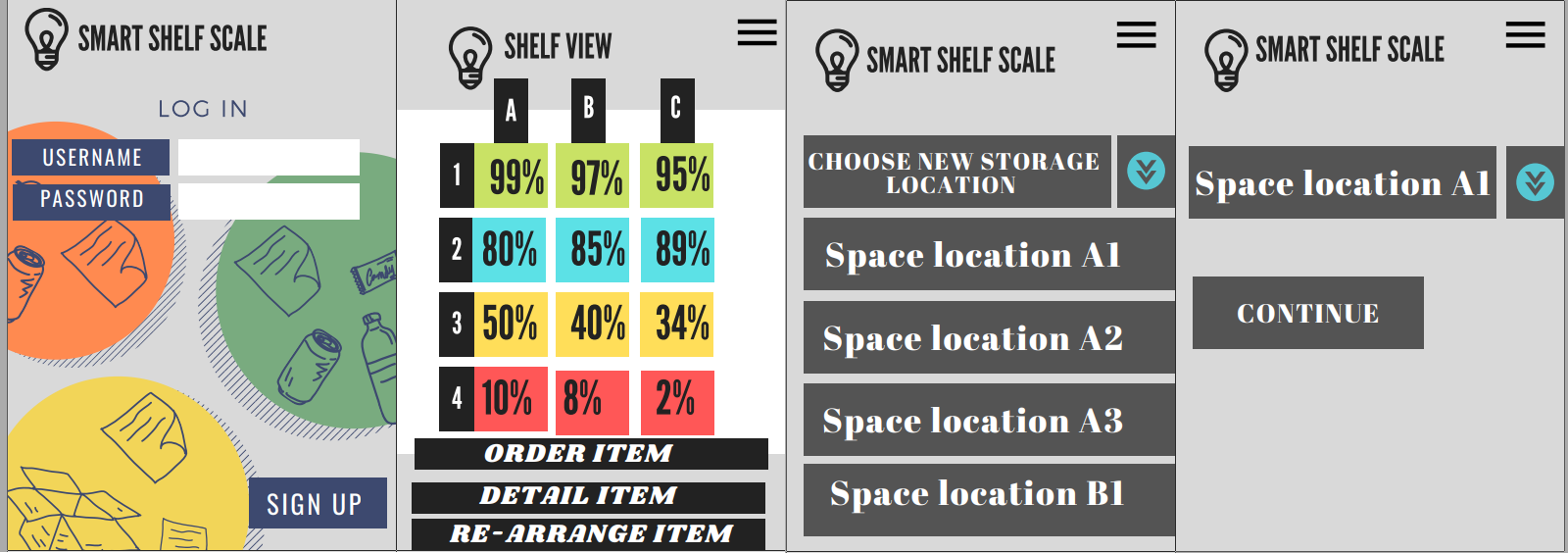
Link to invision prototype web-view: <https://jasperrobinson415543.invisionapp.com/prototype/ck3a323ri004bik01i34myegt/play>

The feedback was positive, the stakeholders we tested with said that things were well thought out pretty much complete. We did get useful suggestions for things to consider and possible changes to make:

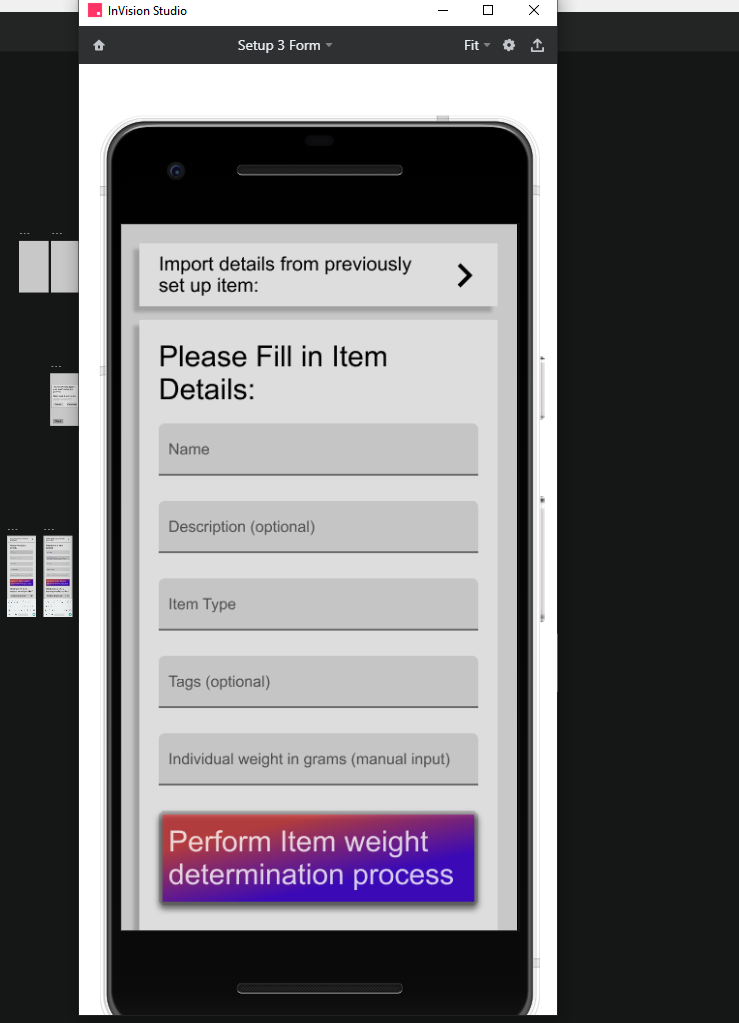
* Both stakeholders said they would prefer the format of the application to be web/computer based rather than mobile/app based.
* Both stakeholders suggested that we add a section to the form for links to restocking information or procedures(URLs). Or supplier codes.
* One suggested and both agreed that having price information on the items stored may be useful, allowing the app to show the value of the items used over periods of time.
* Both talked about the importance of being able to define the layout of the shelves and wanting to have shelves that accommodate different sized bins/trays in a non-grid layout.
* The science lab technician who worked with chemicals had lots of considerations in terms of safety. Such as the predominant displaying of hazard labels, the containing of health and safety information for each chemical in the app and possibly method of contacting relevant services in case of a chemical spill.
  + He suggested that possibly the app could have separate layouts/ forms for different applications of the shelves.
* There are also considerations for chemicals and items with ‘dry weight’ and how we deal with that. (implementing a solution may not be too hard).
* Hatch lab would like if the app could connect to the airtable database.



Zoomed out view all the screens made for the hi-fi prototype in invision studio



Alternative prototype style



One of the hi-fi prototype screens

## High-Fidelity Technical Prototype

We performed research into all areas to do with the technical implementation of the prototype to determine if there were any open technical questions. Most areas seemed to be solvable with ample evidence of feasibility. Some small tests were carried out. The Technical prototype research is appended to the bottom of this document.

## Open Questions about this concept

We have a few design decisions to make. We also need to determine which features are essential (and should be built first) and which are wanted, but not essential.

Generally, our idea of what the product will be has been refined down and there are few major questions left to answer.

## High-Fidelity Technical Prototype Report:

You should present evidence that you constructed high-fidelity technical prototypes that you as a team used to test a technology (e.g. a database, web vs mobile app etc) that might be suitable for your project. This section should also describe how the prototype was used by the team to improve the concept.

Database:

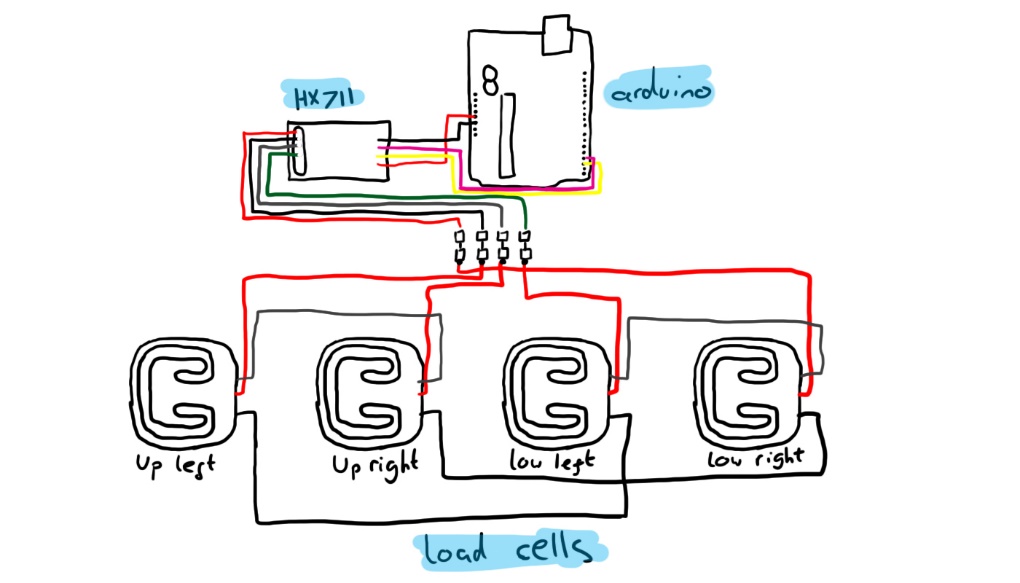
The database system we will be using is MySQL due to its scalability, exhibiting the ability to handle deeply embedded applications with a footprint of only 1MB to running massive data warehouses holding terabytes of information. This is valuable to our concept of an inventory system because it has unlimited potential in terms of how much may be stored using it. We will also be able to set up a MySQL database with either a web app or mobile app, depending on which route we go down for the final build. We will all have experience with this database system when the building phase of the project takes place, so it makes sense for us to use it over other options.

Web Vs Mobile App:

We are currently planning on building a mobile app to pair with our inventory system that allows the user to add, check and keep track of stock levels within the physical inventory system. It is understood that this process may prove difficult, so we have opted for a Web App as a backup if we’re unable to create a mobile app in the given time frame. We will be using Java in JavaFX to build this app. JavaFX builds on top of JDK 13 and is a standalone component. There are 2 different options for developing JavaFX applications, using the JavaFX SDK or a build system like maven or gradle. We plan on using the JavaFX SDK from Eclipse.

Hardware:

For the hardware it is likely that we will use a variation of load cells, an HX711 amplifier and an Arduino to link everything and send data to our database. The HX711 is specially made for amplifying the signals from load cells and reporting them to another microcontroller. The load cells plug into this board, and this board tells the Arduino what the load cells measure. Load cells are specially shaped metal parts that have strain gauges glue to them. The strain gauges are resistors that change their resistance when bent. When the metal part bends, the resistance of the load cell changes which is what the HX711 measures accurately.

There are a lot of examples online of setups like this or similar with code snippets for calibration and setup which we will refer to when setting ours up. To the right is a rough sketch for the setup of the weighing system.

Login System:

If we choose to build a login system for the app or web app, it will most likely be done using MySQL to store the user information. A prototype has been set up for saving user input details to a database and hashing the password for security using bcrypt. This prototype has been setup using node and express for web use, however the processes are similar and this understanding will translate over into an app scenario. Below, a code snippet was taken from the register section which shows how the user inputs are saved and input to a database.

