# Capstone I - Final Proposal

Brandon Russell, Daniel Rowett, Alexander Kalmes, Sam Hudson, Cody Sturgeon, Jackson Carlton
(Group 16)

# Project members

**Capstone group members:**

- Brandon Russell
- Cody Sturgeon
- Daniel Rowett
- Alex Kalmes
- Sam Hudson
- Jackson Carlton

**Champion:**

- Nauman Malik

# Problem

**53** hours

The time spent on average per year at the grocery store

**2~** trips

The number of trips per week to the grocery store

**2/5** buyers

Go to multiple grocery stores to obtain everyday items

**15%** sales

Percentage of all grocery sales that were made online prior to COVID

**Autonomy**

Their are a number of factors that can influence one's ability to go to stores

**Availability**

Local stores may not always have stock of the product you need when you need it

# Home Re:Stock™

# Project Introduction

**What is Home Re:Stock:**

- A network of *smart* sensors* that allow a user to monitor the levels of any consumable product(s) they wish
- Works in conjunction with a user interface so you can monitor a product's current levels or track its usage over time

**Why did we choose this project?**

- The project involves both hardware and software
- The software side of the project is quite substantial
- The project provides a realistic product development experience

# Related works / Existing solutions

**Amazon:**

- Dash button (discontinued)
- Dash services
- Auto re-order

**Retail orientated:**

- Prediction algorithms
- Scan-n-track

| | Home Re:Stock | Dash Button | Dash Services | Auto Re-order | Prediction Algorithms | Scan n-track |
|---|---|---|---|---|---|---|
| Based on real values | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ |
| Multi product compatible | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ |
| Designed for the home | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ |

# Home Re:Stock Design goals

**Short term goals:**

- Provide means to an affordable network of sensors
- Ensure a sensor is able to monitor any consumable product
- Enable users to monitor sensor(s) remotely via the internet
- Monitor the usage of a product over time

**Long term goals:**

- Increase user convenience by enabling automatic reordering of a product
- Integrate with company APIs to look for best prices or alternatives

# How we will help Home Re:Stock

**Improve existing hardware:**

- Refine prototype design
- Create 3 total different form factors

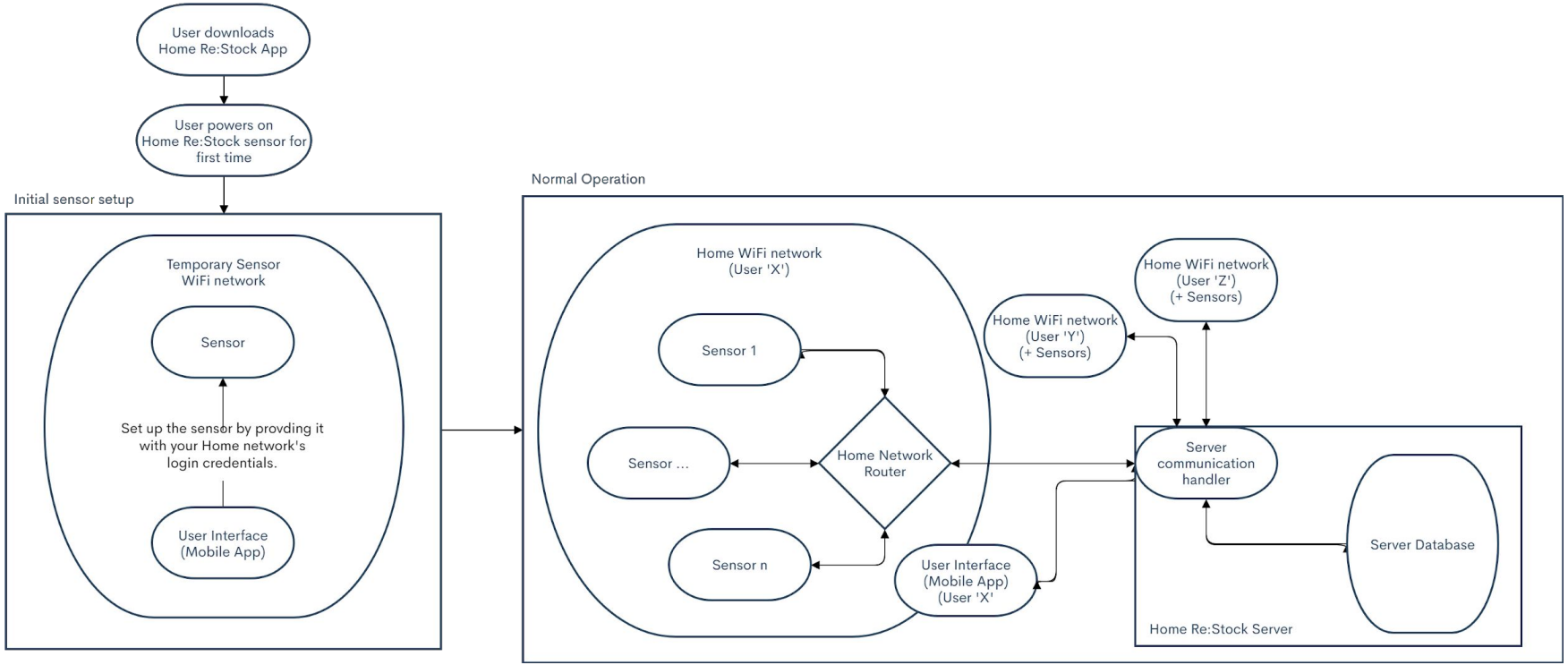
**Create various pieces of software:**

- User interface (app)
- Backend server + database integration
- Sensor firmware

**Deliver a fully working ecosystem of components by the end of Capstone II**

- Provide easy means of increasing feature set without our contribution

# Architecture overview



# Sensor Design and Architecture

**Sensor requirements:**

- Affordable physical design - no more than $8 retail cost
- Have substantial battery life and minimal user interaction after initial setup
- Available in multiple form factors

**Sensor architecture:**

- Hardware: (5 essential components)
  - Casing, Load cell(s), Load cell amplifier, Microcontroller, Power supply
- Software:
  - Two main modes of operation: Setup mode, and Standard mode
    - Communicates with server during Standard mode
    - Communicates with user interface during Setup mode
  - Periodically sends data to server, and can receive commands from server at any time

# Backend Server Design and Architecture

**Backend requirements:**

- Needs to be able to communicate with many different user interfaces and sensors simultaneously.

**Backend architecture:**

- A communication handler will interpret commands received from user interfaces and data from sensors. An action will be performed by the server based on the command issued or data received.
- A database will be used to securely store sensor and account related data.
    - Data will be able to be retrieved only by a user with the proper credentials

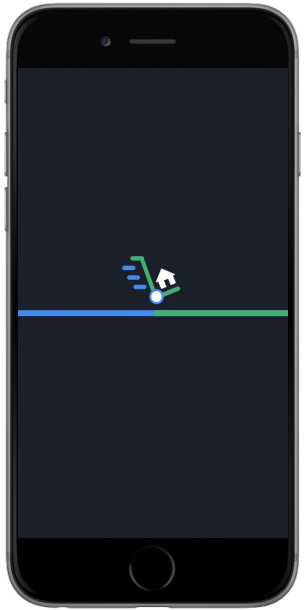
# User Interface Design and Architecture

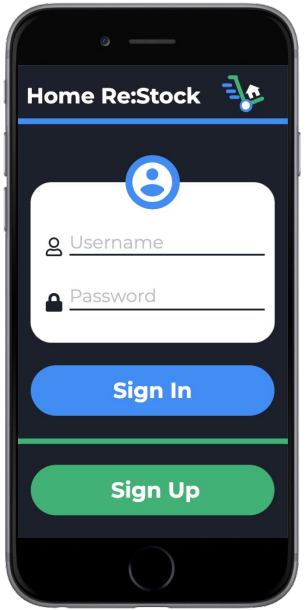
**User Interface requirements:**

- Must closely resemble the provided application mockups
- Still able to work with reduced functionality even when offline

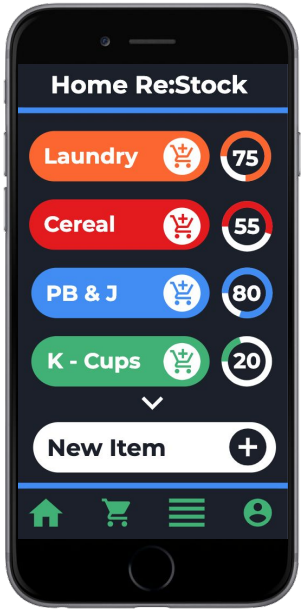
**User Interface architecture:**

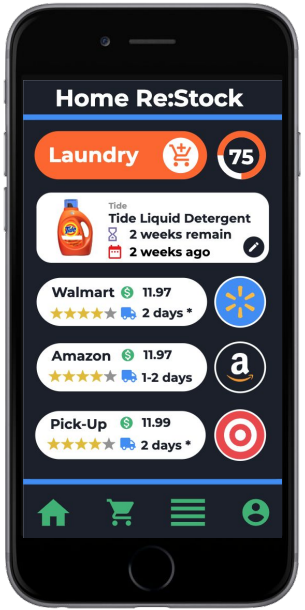
- Only communicates directly with a sensor during the sensor's initial setup
  - Able to provide the sensor with details required for WiFi connection and account info
- Communicates only with the server backend during normal operation
  - Able to issue commands and view sensor data by using the server as a middle man
- Saves 30$^{\sim}$ days worth of data per sensor for offline functionality



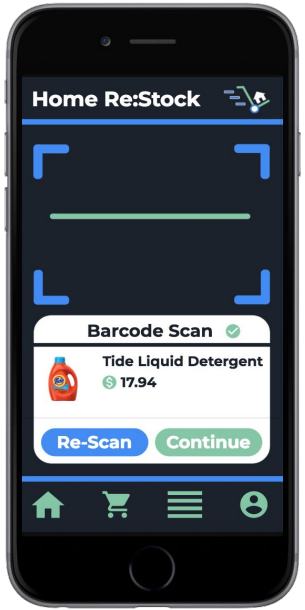


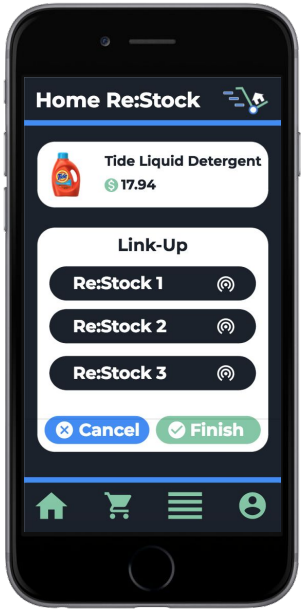

## Start & Sign In



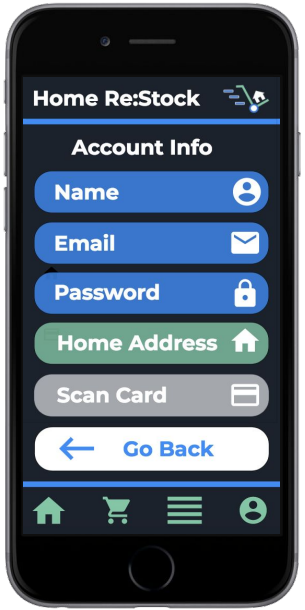



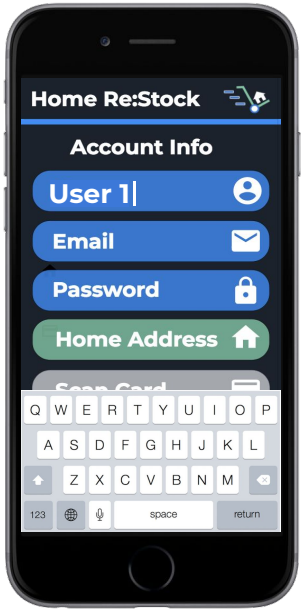

## Home Screen



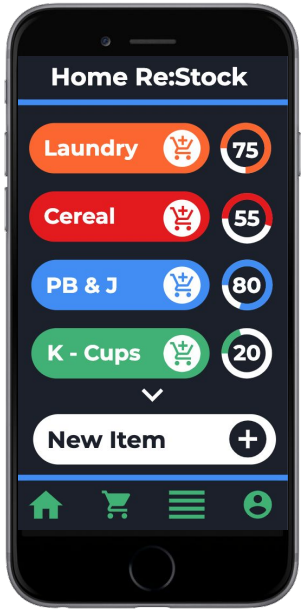



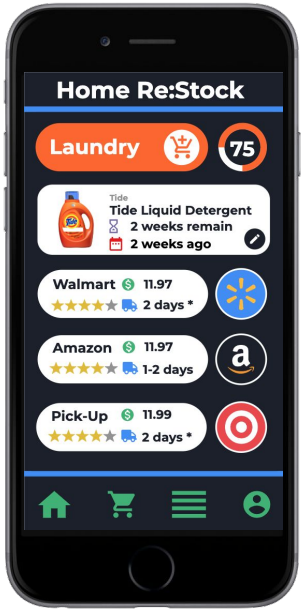

## New Item Set-Up



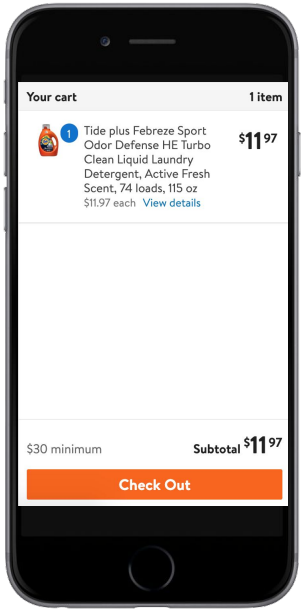




## Account Set-Up




## Home & Item Page





## Cart & Checkout

# Tasks and Schedule

**Sensor:**

- Refine the design of the prototype. Create a small, medium, and large version
- Implement sensor firmware

**Server:**

- Document a list of all [(app, sensor) ➜ server] commands. Create an IO outline
- Design a detailed database schema
- Implement code for each of the above commands

**User Interface:**

- Refine existing mockups. Create more where needed
- Implement the various features of the app. Follow server IO outline closely

# Deliverables

**Hardware:**

- A single physical sensor

**Software:**

- Sensor software / firmware
- Hardware CAD files
- Backend server code* + database scheme and initial data
- User interface source code

**Documentation:**

- Software design overview and hardware design overview documents
- User interface mockups
- Final report

# Facilities and Equipment

**Facilities:**

- FAY fabrication laboratories

**Equipment:**

- PC + Programming IDEs
- Autodesk Inventor (Student)
- 3D Printer for initial prototype / CNC for final prototype
- Electrical tools
- The various components that make up a single sensor