

IOWA STATE UNIVERSITY

SD May 20-19

# To Online Shop or To Not Online Shop

**Team Members:** Amiah Gooding, Matthew Martin, Maxwell Minard,  
Travis Stanger, Smruthi Sandhanam, and Yana Aleksandrova

**Client/Advisors:** Goce Trajcevski

# Outline

1. Introduction
2. Design Plan
3. Statement of Work
4. Timeline and Resources
5. Testing and Implementation
6. Future Plans

# Introduction

# Trends

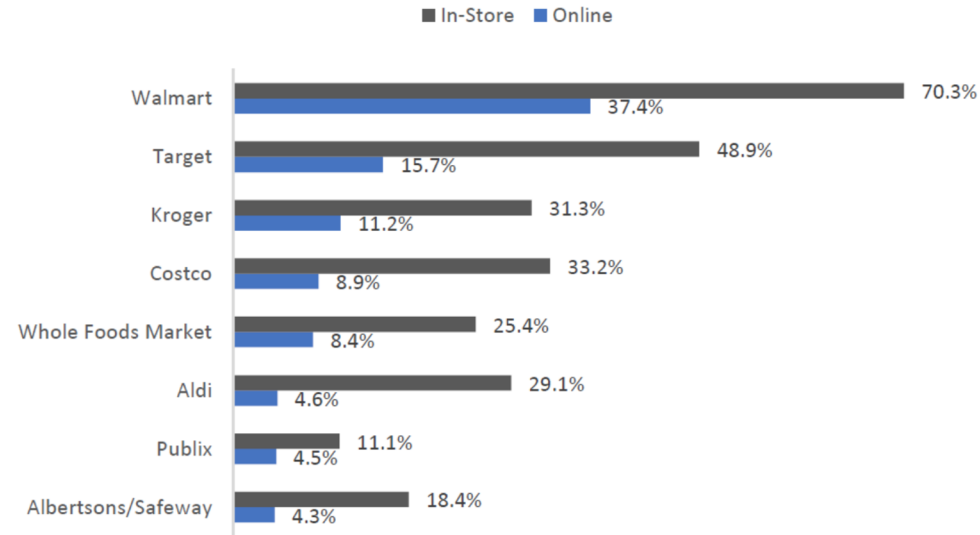
## Online Shopping:

- E-Commerce growing 3 times faster than retail
- Companies have made it easier to buy groceries online
  - Offering discounts on items and delivery
- In 2019, 93 million online grocery purchasers

## In-Store Shopping:

- 56% of shoppers would like to see and touch items before purchase
- Marketing trend towards increasing in-store purchases using location aware advertising
  - Offer discount coupons based on user's proximity of particular stores

# U.S. Online Grocery Survey (2019)



*Base: US Internet users ages 18+ who have bought groceries in-store/online in the past 12 months (1,803 in-store and 695 online)*

*Source: Coresight Research*

Figure 1: R. Redman, "Number of online grocery shoppers surges," Supermarket News (SN), May 2019.

# Goal

- Design a solution that will help users find a balance between in-store and online shopping experiences
- Develop an end-to-end IoT solution that will:
  - Monitor the status of items in a shelf or cabinet
  - Generate a list of items “to buy” and prepare an online order
  - Location-aware notification for users that certain items needed are available at a nearby store at acceptable prices
  - Automatic update of online orders if user decides to purchase items in store

# Design Plan

# Design Plan

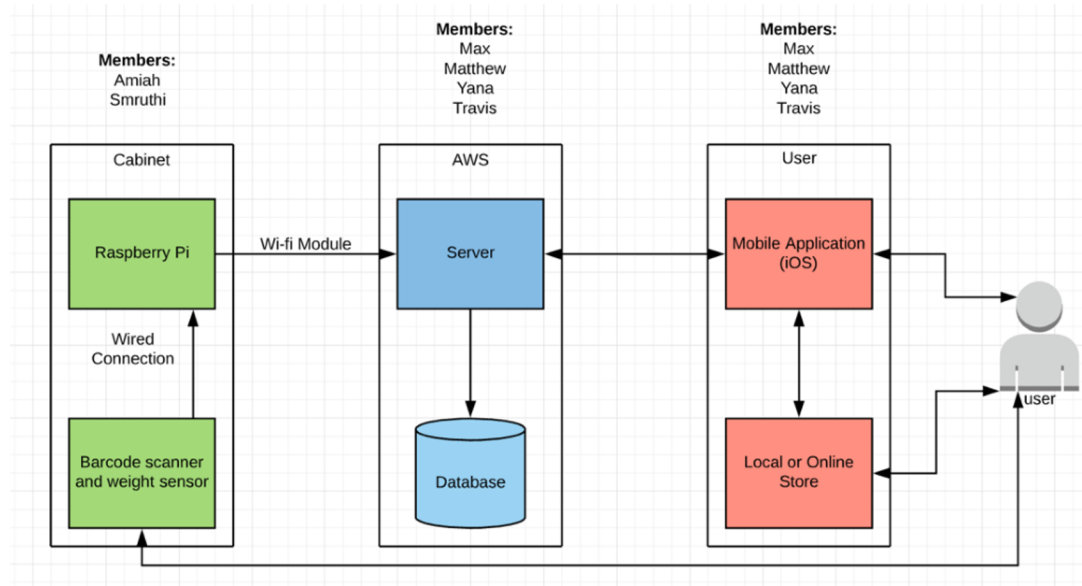
## Non-Functional Requirements:

- Availability: Available to update automatically every 12 hours
- Data Integrity: Accurate & consistent data
- Deployment: Pantry cupboard
- Scalability: Modular designs
- Resilience: Constant change
- Usability: Intuitive, clear steps for setup and access



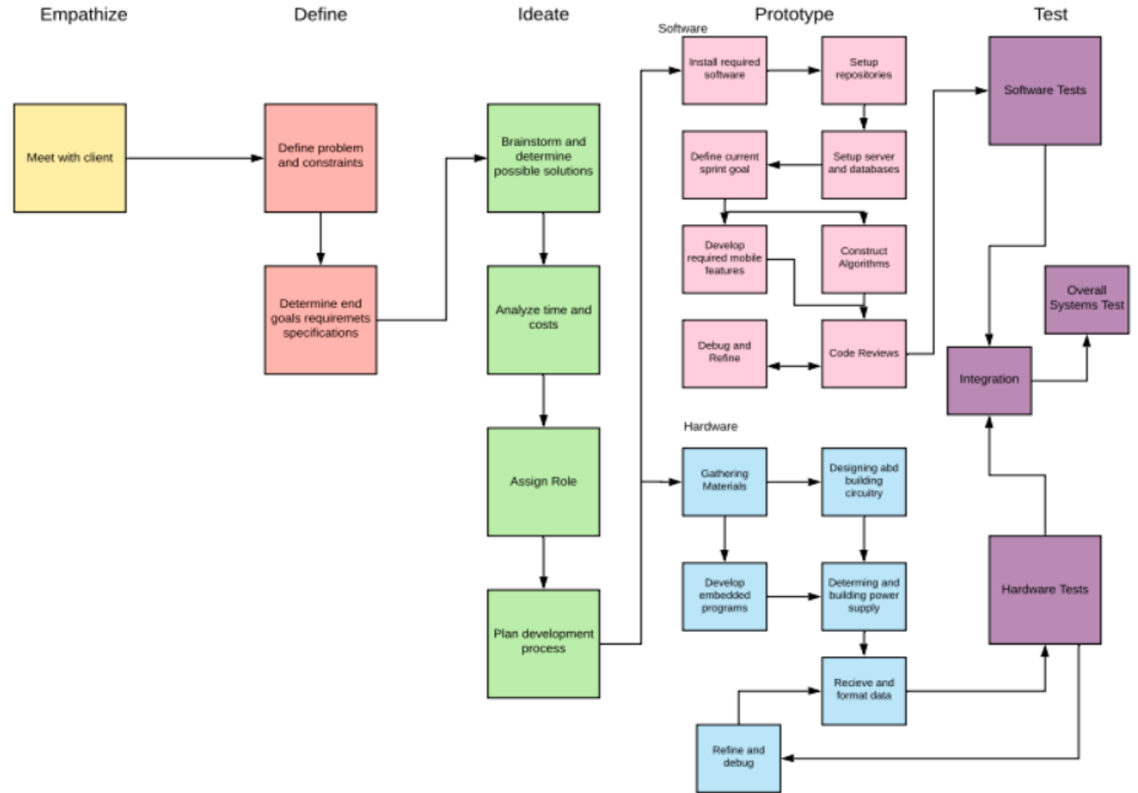
# Design Plan

## Proposed Design



# Design Plan

## Design Plan:



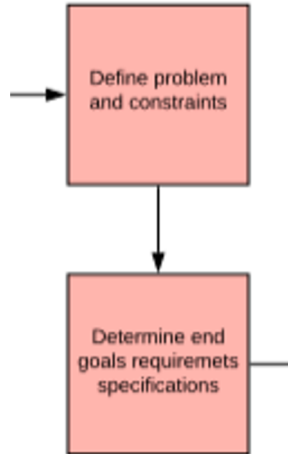
# Design Plan

## Design Plan: Empathize



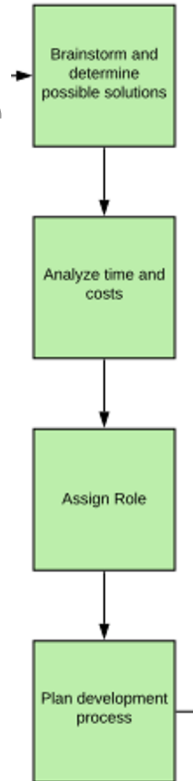
# Design Plan

## Design Plan: Define



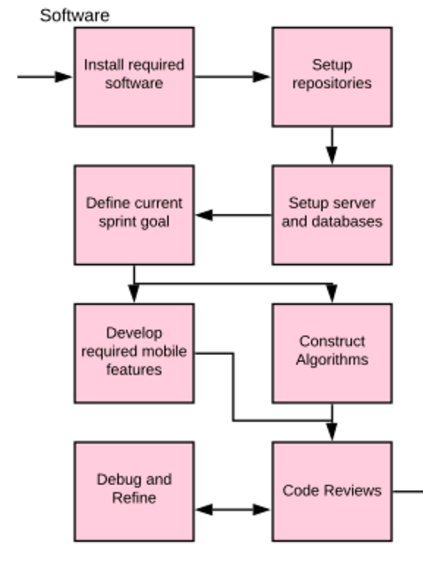
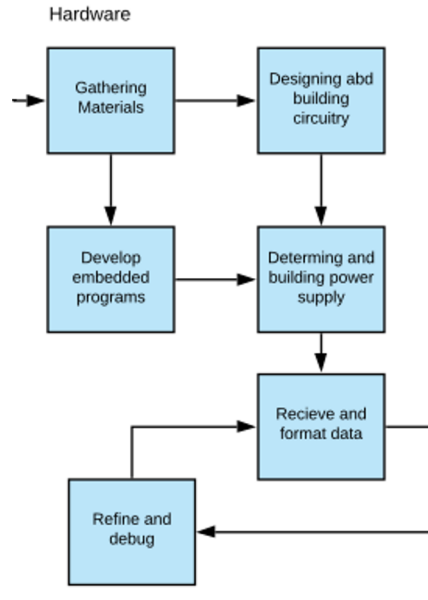
# Design Plan

## Design Plan: Ideate



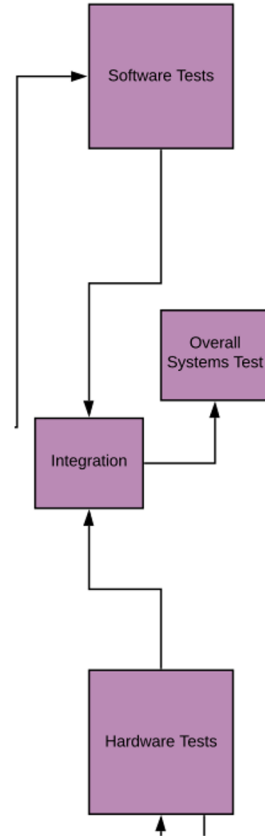
# Design Plan

## Design Plan: Prototype



# Design Plan

## Design Plan: Test



# Statement of Work



# Technical Considerations and Constraints

- Choice of Sensors - RFID vs. Barcode Scanner
- Database - Amazon Web Services
- Master System - Arduino vs. Raspberry Pi

# Possible Risks and Mitigation

- Sensor Degradation - Sensors may not calibrated over time
  - Response - Calibrate the sensors on a regular basis
- Communication - A connection with the application and the database updating timely
  - Response - Having strong time constraints on sending updated information to the database
- Unfamiliarity - New Technology
  - Securing outside sources to compile information and placing in uniform location

# Phase Decomposition

**1**

Proof of Concept

- Data collection from sensors
- Data is being transmitted to database from sensor
- Front-end can visualize data from the database

**2**

Minimum Viable Product

- Integrate system for multiple sensors to be registered to a specific product
- Demand for new item purchase can be generated based on pantry contents

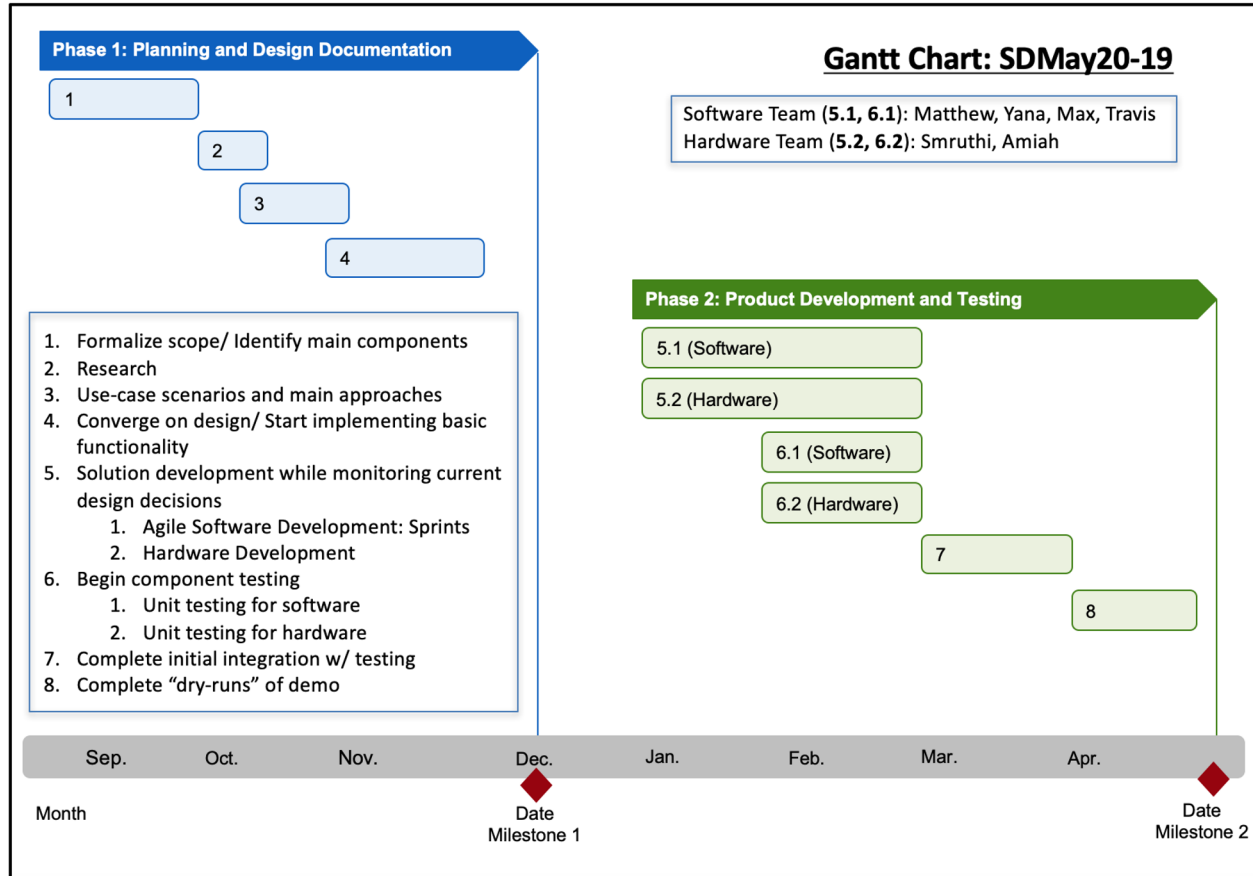
**3**

Finalized Product

- Complete integration of sensor arrays for multiple products
- Purchase suggestion for multiple products can be determined

# Timeline and Resources

# Project Timeline



# Feasibility Assessment

Expectation:

- Proof of concept IoT solution
- Target towards one spot in the home

Main Challenges:

- New technologies (Ex: AWS)
- Time Management



# Personnel Effort Requirements

- Set up AWS server and database (40+ hours)
- Develop front-end user interface (100 hours)
- Algorithms and data analysis (50 hours)
- Construct circuits/hardware components (80 hours)
- Embedded programming (80 hours)
- Testing and Integration (150 hours)

# Resource Requirements

- Raspberry Pi 3 Model B
- Barcode scanner
- Weight Sensor
- Wires and circuit components
- Power supply
- AWS server and database
- Makeshift cabinet and items for test simulation





# Financial Requirements

Budget: Under \$200

Material Costs:

- Raspberry Pi: \$35-\$40
- Weight Sensor: Potentially free?
- Barcode Scanner: \$50-\$100

# Testing and Implementation

# Functional Test Plan

- **Hardware Testing**
  - Manually verified connections
  - Python Scripts
- **Software Testing**
  - XCTest: built-in unit testing with XCode
  - Manual scenarios
- **Integration Testing**



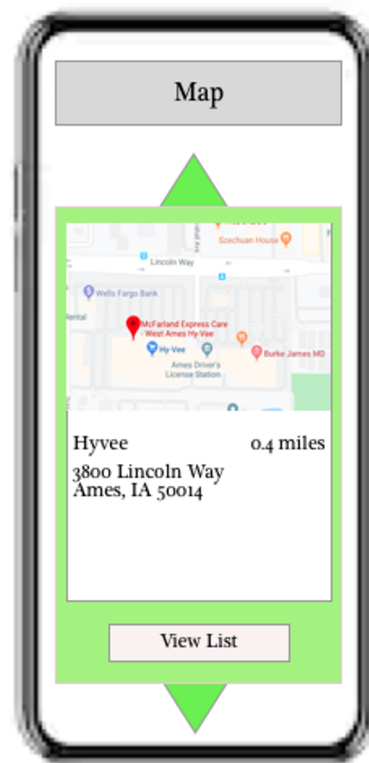
# Non-Functional Test Plan

- **Scalability:** Progressively add users to database and ensure all have access.
- **Data Integrity:** Send data from sensor to database and compare log files in both places.
- **Usability:** Compare time to manually find a deal and the time for automatic alert to be sent.

# Current Status

- Raspberry Pi setup
- Front-End screen sketches
- Amplify framework setup
- Database Initialization

# Current Status



# Future Plans

# Spring 2020

- Revisit design decisions (January)
- Define and start individual component testing (February)
- Complete initial integration, define and start integration testing for plausible use cases (March)
- Complete dry-runs of demo, finalize deliverable version (April)

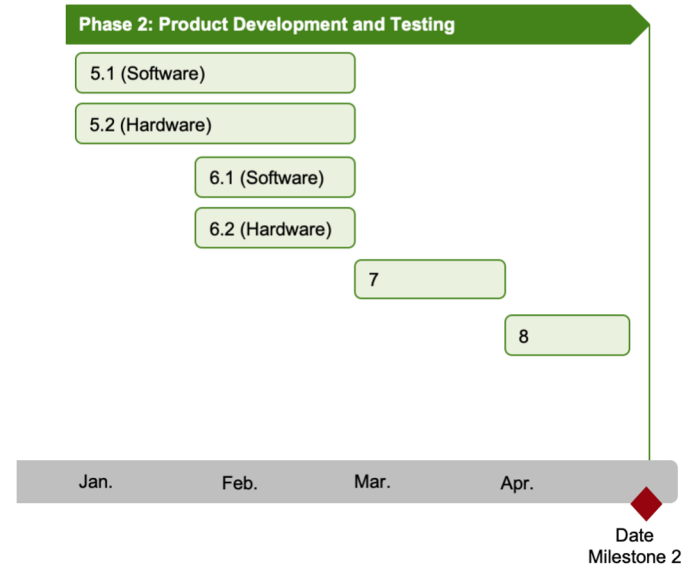


# Spring 2020: Gantt

5. Solution development while monitoring current design decisions
  1. Agile Software Development: Sprints
  2. Hardware Development
6. Begin component testing
  1. Unit testing for software
  2. Unit testing for hardware
7. Complete initial integration w/ testing
8. Complete "dry-runs" of demo

## Gantt Chart: SDMay20-19

Software Team (5.1, 6.1): Matthew, Yana, Max, Travis  
Hardware Team (5.2, 6.2): Smruthi, Amiah



# Questions?