

# Smart Kitchen Containers

Akhil Surya Vamshi 120050067

Bhargav Chippada 120050053

Jagadeesh Nelaturu 120050055

Pranay Dhondi 120050054

# Description

Build a platform on which different containers can be placed and replaced. The weights of these containers are to be logged and displayed to the user, along with daily consumption and calorie consumption in an appealing and easily comprehensible manner. If the amount of a commodity goes below a certain level, the user is to be alerted.

# Requirements

- Whenever a container is put on the platform or taken from it, both the RFID readings and combined weight is sent to the Raspberry Pi, and is stored on the SD card.
- In the above case, the data is processed and the weights of the relevant containers are stored separately for easy retrieval.
- A HTTP server continuously runs on the R-Pi, to which the user can connect whenever desired.
- The home screen continuously shows the commodities present, and the current amount of every commodity. Also, it alerts the user if the quantity of a commodity goes below a set minimum value.
- Whenever the user requests for the details of a particular commodity, the data pertaining to this commodity is processed and plots of its weight, consumption and calorie consumption are shown to the user.

# Project Plan

- Make an RFID Reader work with R-Pi. Next make 2 RFID readers work with R-Pi using chip select pins
- Interface load sensor to tiva-c. Send the weight reading to R-Pi using USB
- Code the UI on laptop. Run the python server on R-Pi to host the web page
- Process the weight reading, rfid tag readings to output the weights of containers placed on the platform
- Work Division:
  - Akhil Surya - Interface load sensor to Tiva-C. Weight reading calibration.
  - Bhargav Chippada - Interface 2 RFID readers with R-Pi using chip select pins (SPI interface)
  - Jagadeesh and Pranay - Processing the weight and rfid tag readings to find the weight of individual containers. Coded the website UI.
- Critical tasks: Balancing the platform on load sensor, calibration. Interfacing multiple rfid readers.



# Diagram



# Innovation and Challenges

- Innovations:

- Platform design
- Multiple rfid reader interfaces
- Load sensor mechanical design

- Challenges:

- **Load sensor readings**

The load sensor sends out analog signals so we need to take the readings after the value has reached equilibrium so we cannot take the readings at a high frequency hence we take the readings after every 5 seconds.

- **Load sensor calibration**

We could not calibrate the load sensor accurately because it requires using many different weights

- Interfacing multiple rfid readers to R-Pi

# Tasks

- Make an RFID Reader work with R-Pi. Next make 2 RFID readers work with R-Pi using chip select pins

Problems faced:

- The RFID reader provided by lab didn't work. We tested it meticulously by connecting it to laptop using USB-serial communicator. We ordered new RFID readers online
  - Interfacing multiple RFID readers to R-Pi using SPI interface
- Interface load sensor to tiva-c. Send the weight reading to R-Pi using USB

Problems faced:

- Load sensor is connected to HX711 for ADC and amplification
  - Load sensor didn't work with R-Pi due to some inherent timing issues so we had to use Tiva-C board to get the readings and then communicate them to R-Pi



# Tasks

- Code the UI on laptop. Run the python server on R-Pi to host the web page

Problems faced:

- None

- Process the weight reading, rfid tag readings to output the weights of containers placed on the platform

Problems faced:

- Processing the data dumped by the readers was a challenge because we get the total weight of multiple containers and we had to see the changes in rfid to figure out individual weight of containers

- Mechanical Part

Problems faced:

- Building the platform was a very hard challenge because the whole platform had to be balanced on the load sensor

# Review

- Test Plans:
  - Load sensor readings  
We calibrated the load sensor using a devised formula and tested it for different items
  - Multiple Rfid reader  
We meticulously checked the polling of readings by multiple rfid readers
  - Interface load cell to measure weight  
We interfaced the load cell connected to HX711 to TIVA as we had timing issues with using R-Pi and HX711 together
  - GUI with data visualization and alerts  
We have implemented both visualizations through graphs and alerts by highlighting containers that are running low in amount

# Review

- Performance metrics:

- Load sensor readings

The load sensor sends out analog signals so we need to take the readings after the value has reached equilibrium so we cannot take the readings at a high frequency hence we take the readings after every 5 seconds. This sampling frequency effects the overall performance by slowing speed down.

- Load sensor calibration

We could not calibrate the load sensor accurately because it requires using many different weights

- Changes on the platform have to be made with a gap of 5 seconds. We want to reduce this delay as much as possible (future work).

# Re-usability features

- Our design is very modular. Code for load sensor and rfid reader interfacing is separate
- You can change the rfid reader code, load sensor code, server code individually without affecting one another. The readings are exchanged mutually using ports or by reading/writing to a file
- We adhered to the coding standards specified

# Future enhancements

- The user interface can be extended to be viewed from an Android/iOS mobile phone for the users' convenience.
- The platform can be made to accommodate a larger number of containers by using more load sensors and RFID readers.
- Visiting the website requires the IP of raspberry which hasn't been found automatically. We could customize the server to send a particular reply on particular request to pair the device with R-Pi without having to enter the IP
- The mechanical design can be improved to make the platform sleek.
- This product can be marketed in collaboration with other Smart-Kitchen and Modular-Kitchen platforms.
- When a commodity is beyond a certain minimum quantity, the user can be alerted through SMS, or the application can place an order to the seller directly.