GNG1103

Final Design Report

PORTIONATOR 2.0

Submitted by

GROUP C2

Yigit Gunes, 300051282

Sandeep Shrestha, 300090525

Tian tian Wang, 300062562

Symoom Saad, 300063843

April 15, 2019

University of Ottawa

Abstract

Automated kitchen is being adopted everywhere by many restaurants. uOttawa wants to do the same, and they need help with the distribution system. Our team went through the engineering design process to try to find the solution to this problem. Our final prototype was unique because we focused on space efficiency, washability and reliability.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Definition

1 Introduction

The restaurant industry nowadays has a very high turnover rate. This means that once people receive enough training in their current restaurant, they often leave to search for a higher paying job. uOttawa's Thai Express is experiencing this problem as well. In addition, Thai Express also wants to reduce wait time for their customers. Automating their kitchen is one of the best solutions available for those problems today.

Our group had to design the portioning part of the automatic kitchen, which would include:

1. A bulk container where ingredients would be stored;
2. A mechanism that would portion out the precise amount of ingredients when required;
3. A cooling system that would cool the food ingredients to a temperature less than 4°C.

Our design was unique as we focused on three features:

1. Washability: Ensuring that every part that contacted food could be easily detached to help the sanitation process at the end of the day and let our customer experience a greater level of hygiene.
2. Space efficiency: Ensuring that the least amount of space and material is being wasted
3. Reliability: Minimizing the number of moving parts to only one mechanical part that is capable of moving things horizontally. It turned out that having one more motor would really ensure the precision of the portions dispensed. As a result, our future models will try to utilize two motors.

Distributing each portion only takes 11 seconds and it could be reduced further by increasing the motor speed.

We used Engineering Design process (Empathy, Define, Ideate, Prototype, Test) to work on our distribution system.

2 Empathy

After the first customer meeting, our team discussed on what our customer wants from the design, based on the information given by our client. Using his direct quotes, a list of needs was generated, and later grouped and ranked. Below, you can find all the relevant information organized in a table.

Table 1 : Customer Needs and Priorities

List of Needs	Type of Need	Importance
Durable parts Easy to fix parts Esthetic	Durable, interchangeable parts	4
Consistency Shorter wait time Distribution of bulk food into a cooking vessel Ingredients distributed precisely Sequenced distribution Eliminate human mistake Fast and efficient distribution Distribution of sauces, seasoning	Efficient distribution	5
Store up to 40-50 servings Minimize space Well organized in the back Food stored in a temperature-controlled environment	Optimized storage	5
I/O and program Esthetic Ability to skip ingredients Improve food quality Better service	Customer Service	2
Storage of food Storage height (>6 inches but not too high) Sustain up to 185 F Food stored in a temperature-controlled	Safety	5

environment Easy to sanitize		
Lower cost Decrease staff Replacement for cook Reduce cost Increase profitability Minimize space	Cost	2

Based on this table, we concluded that food safety, efficient distribution, and optimized storage were the three most important features to consider for our prototype because:

1. Food safety: Customer wanted food to be stored in a specific temperature (<4^{o}) (due to legal issues) and above a specific height (>6 inches but not too high to minimize the chances of workplace injury without placing the food on the ground).
2. Efficient distribution: Customer wanted a fast and efficient distribution system that would shorten the wait time and eliminate human error factor.
3. Optimized storage: Customer wanted to reduce costs since the cost to rent a plot per square feet is expensive, especially in Downtown Ottawa. Optimizing the amount of space the mechanism occupies can help reduce costs.

If we wanted our designs to perform well, we would have to excel on all three factors above.

3 Need Identification and Product Specification Process

After recognizing the needs of our client, we formulated the following problem statement in order to help focusing our final design:

uOttawa's cafeteria wants to build an automated kitchen, and our part in that process is to design an automatic distribution system that:

i. *Prevents bacterial grow;*

ii. *Is space-efficient, in order to place as many subsystems for different ingredients beside each other as possible, and*

iii. *Is reliable for distributing precise amount of ingredients.*

After formulating our problem statement, we moved on to identify the design criteria by using our customer's needs and ranked them according to their priorities:

Table 2 : Design Criteria for each of the Customer's Needs

Number	Type	Needs	Design Criteria	Priority
Durable, interchangeable parts				
1.	Non functional	Must be accessible, replaceable and have fixable parts	Parts readily available on market (item quantity)	2
	Non functional		Minimum amount of simple, adn separate parts	1
2.	Non functional	Should have durable parts	Parts last: 3-5 years maximum use (or) 8-10 years normal use	1

	Non functional		Material like stainless steel	3
Efficient distribution				
3.	Constraint	Must reduce the wait time of customers	Total wait time: 8-10 mins Average recipe cook time: 5.75 mins Therefore, distribution time: 2.25-4.25 mins	3
4.	Functional	Should distribute precisely and sequentially	Efficient measuring technique for portioning (kg or volume)	3
5.	Functional	Should distribute ingredients consistently	Accurate measuring technique for portioning (kg or volume)	3
	Functional		Timing for distribution is precise (consistent)	3
6.	Non functional	Should have an efficient way to transport the ingredients from storage space to the cooking vessel.	Ingredient storage space close (m) to cooking vessels	2
	Functional		Reduce time (s) and energy (Watts) while distributing ingredients	3
Optimized storage				
7.	Constraint	Must have optimal number of servings stored in storage space	40-50 servings	3
8.	Functional	Must have easily accessible storage locations for each ingredient.	Min 2.86 ft (Average Waist Height) Max 4.68 ft (Average Shoulder Height)	3
9.	Non functional	Should use as little space as possible	Volume (m3)	2

10.	Constraint	The ingredients should be stored in a cold and sealed place	Temperature 4℃	3
Customer Service				
11.	Non functional	Could have nice esthetics	Nice shape	1
	Non functional		Pleasing to eyes	1
12.	Non functional	Must be able to customise order	User Interface	3
13.	Constraint	Must have less wait time	Distribution time: 2.25-4.25 mins	3
Cost				
14.	Non functional	Should cost as least as possible.	Cheap, but durable parts ($)	2
Safety				
15.	Constraint	Temperature controlled storage	A cooling system that detects temperature > 4℃ and cools the ingredients down.	3
16.	Functional	Containers must withstand 185 F	It should not melt/deform in temperatures > 185 F	3
17.	Constraint	Storage containers must be 6 inches off the ground	Height > 6 inches	3

This helped us identify all functional requirements, non-functional requirements and constraints that were to be considered for our prototype. Assigning metrics to our design criteria helped us narrow down our design. Most of the metrics above were set as constraints by our client, Patrick. We decided to research and benchmark for the rest of the unknown metrics, results of which can be seen below.

Table 3 : Benchmarking for Best Metrics

	Importance	Spyce		Creator		FuA-men	
Cost starting at (USD)	2	7.50$	1	6.00$	3	6.40$	2
Speed of ingredient distribution (Automatic)	5	Yes	3	Yes	1	Yes	2
Durable parts	4	No information available		No information available		No information available	
Customer wait time	5	3 min or less	3	5 minutes	2	~5 minutes	2
Storage of ingredients	5	No information available		Organized neatly in their own compartments	2	Stored in containers around the robot arms	2
Customer Service	2	2 staffs	3	9 staff	1	3 staff	2
		Touch screen self service	3	A cashier takes the order and inputs it themselves	1	No information available	
		Cooking processes displayed	3	Cooking processes displayed	3	Cooking process is displayed	3

Safety (of food)	5	Everything is kept hidden (probably in a fridge)	3	Stored in fridge and restocked by human when empty	3	Everything is kept hot and within an arm's reach of the robot arm	3
Total Score			65		56		59

From the table, you can clearly see that Spyce is our biggest competitor, in terms of performance, and if we use their metrics, our design should be one of the best ones in the market.

4 Conceptual Designs

We used three methods to assist us in the ideation process – brainstorming, sketching and cross-stimulation. We brainstormed and cross-stimulated our ideas during our weekly meetings, sketched between our meeting sessions. Down here is the final concept for our first conceptual design:

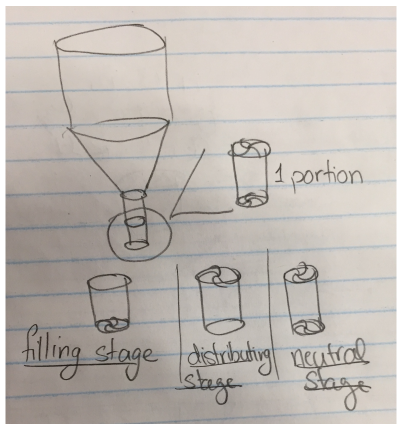


Figure 1 : First Design Idea

This is a simple design that we came up with and it encompassed all our different ideas. There would be a bulk at the top, which would be funneled down to the portioning part. In the portioning part, there would be two "doors" in the pipe that would open and close. This mechanism had three main stages:

1. Neutral stage: Both doors would be closed, and no food would be moved.
2. Filling stage: Top door would open, and the ingredients would fill in the empty part between the two doors. After the part is filled, the top door would close.
3. Distributing stage: Bottom door would open and the portion that is separated between the two doors would be distributed to the container below.

The mechanism would cycle between the three stages to distribute ingredients when necessary.

The design had to be scrapped, as the customer had many concerns regarding its functionality. A new conceptual design had to be made, and this one was to be moved using a linear actuator. More details regarding this change can be found under the Prototype I – Customer Review section.

5 Project Plan, Execution, Tracking & Bill of Materials

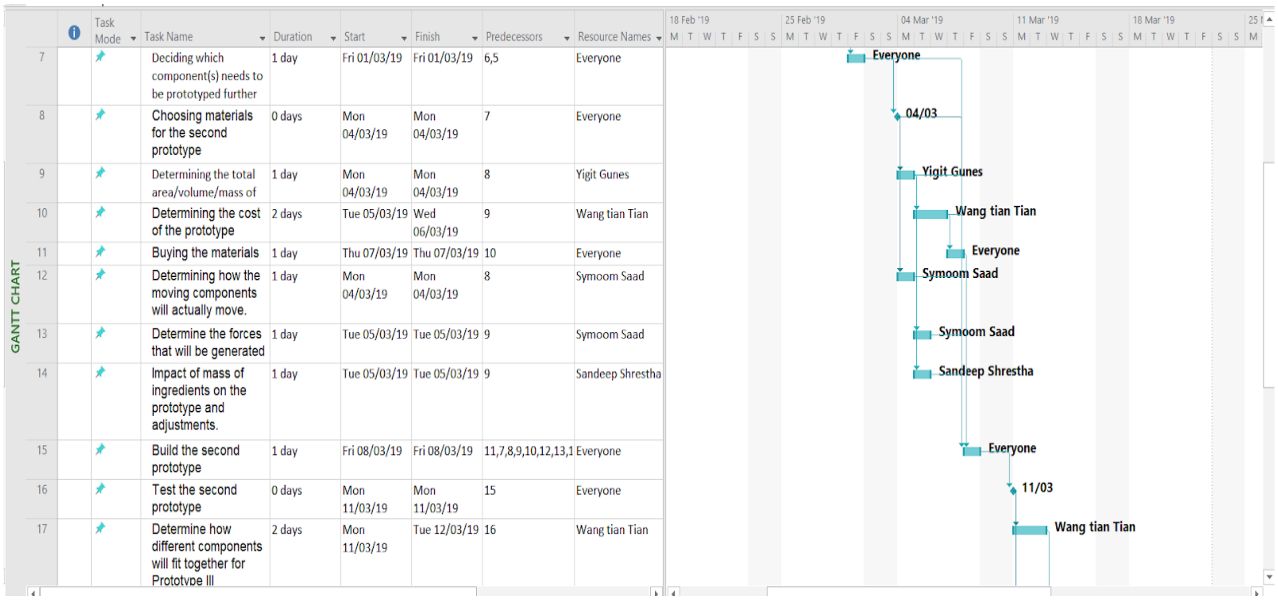
Now that the conceptual design for the prototype was finalized, we started to assign tasks to everyone and organized that information on a table first and a Gantt Chart afterwards. We had to estimate the duration of each tasks and most of that was done using Planning Poker as no one definitively knew how long each task should take. The tasks and their duration were adjusted along the way, but below is the original list of tasks with their respective owners and the Gantt Chart to give an idea of how our project went.

Table 4 : List of Tasks to do with Duration and Owners

Tasks	Duration (days)	Owner(s)
Refinement of prototype ideas	1	Everyone (discussion)
Design on paper of the first prototype		
Identifying the different components of the prototype (motor, controlling chip, building materials, etc.) and designing each component in more detail (dispensing mechanism, sanitization station, etc.).	2	Everyone (divided)
Making the first prototype (cardboard, scrap)	5	Everyone
Test first prototype	3	Everyone
Revising the first prototype, making changes as necessary		
Deciding which component(s) needs to be prototyped further for Prototype II to ensure that our ideas will actually work	1	Everyone
Choosing materials for the second prototype	2	Everyone

Determining the total area/volume/mass of the prototype so we know how much material is needed to build the prototype	1	Yigit
Determining the cost of the prototype	2	Tian
Buying the materials	1	Anyone
Determining how the moving components will actually move. Determine the forces that will be generated to ensure that sufficient support is implemented	1	Symoom
Impact of mass of ingredients on the prototype and adjustments.	1	Sandeep
Build the second prototype	1	Everyone
Test the second prototype	2	Everyone
Determine how different components will fit together for Prototype III	2	Tian
Determining how the electrical components will actually be implemented	2	Symoom
Implementing the mechanism to the prototype with programming	2	Sandeep
Determining if we can optimize anything + help everyone finish in time	2	Yigit
Build the Prototype III	5	Everyone
Testing the prototype	3	Everyone
Improvements on the prototype and testing those improvements	4	Everyone

To make the information above more visual and less daunting, a Gantt Chart was made using the same information. Gantt Chart was useful as it automatically produced due dates based on dependencies, which helped everyone stay on task.



	Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
7		Deciding which component(s) needs to be prototyped further	1 day	Fri 01/03/19	Fri 01/03/19	6,5	Everyone
8		Choosing materials for the second prototype	0 days	Mon 04/03/19	Mon 04/03/19	7	Everyone
9		Determining the total area/volume/mass of	1 day	Mon 04/03/19	Mon 04/03/19	8	Yigit Gunes
10		Determining the cost of the prototype	2 days	Tue 05/03/19	Wed 06/03/19	9	Wang tian Tian
11		Buying the materials	1 day	Thu 07/03/19	Thu 07/03/19	10	Everyone
12		Determining how the moving components will actually move.	1 day	Mon 04/03/19	Mon 04/03/19	8	Symoom Saad
13		Determine the forces that will be generated	1 day	Tue 05/03/19	Tue 05/03/19	9	Symoom Saad
14		Impact of mass of ingredients on the prototype and adjustments.	1 day	Tue 05/03/19	Tue 05/03/19	9	Sandeep Shrestha
15		Build the second prototype	1 day	Fri 08/03/19	Fri 08/03/19	11,7,8,9,10,12,13,1	Everyone
16		Test the second prototype	0 days	Mon 11/03/19	Mon 11/03/19	15	Everyone
17		Determine how different components will fit together for Prototype III	2 days	Mon 11/03/19	Tue 12/03/19	16	Wang tian Tian

Figure 2 : First Tentative Schedule

Of course, we could not keep up with the schedule all the time, so we had to adjust and do extra work to compensate for any unfinished work. Some of the tasks took longer than expected because our estimations for how long certain tasks would take were incorrect. For example, wiring for Arduino Uno and generating 3D models for 3D printing took us an exceptionally long amount of time.

We also had to keep track of all the materials we bought since we only had a $100 budget. Table below shows all the materials we had to purchase, with their quantity and cost.

Table 5 : Bill of Materials

Material	Amount	Cost
MDF	2x	$2.5 per sheet
Metal sheet	1x	$3.13
Arduino Uno	1x	$13
Linear actuator	1x	$56.59
Relay	1x	$11
Power adaptor (12V)	1x	Borrowed from a friend
Total	$86.22	

We used a total of $86.22, which was under the budget of $100. The linear actuator ended up costing a lot due to our poor planning. We were running out of time, so instead of performing some calculations and ordering a reasonable actuator, we decided to buy a powerful one to ensure that it would work with our design.

6 Analysis

For our first prototype, we tried to optimize:

1. The shape of different components
2. The amount of materials needed to construct them and
3. The volume occupied by different components.

For the bulk portion, we compared cuboidal and cylindrical boxes of different volumes and compared their prices to pick the better design.

We first came up with an equation for each of the boxes (height vs surface area) to find their optimal length. Below is a sample graph for the graph equation for a cubic box containing 236.5 cm^3.

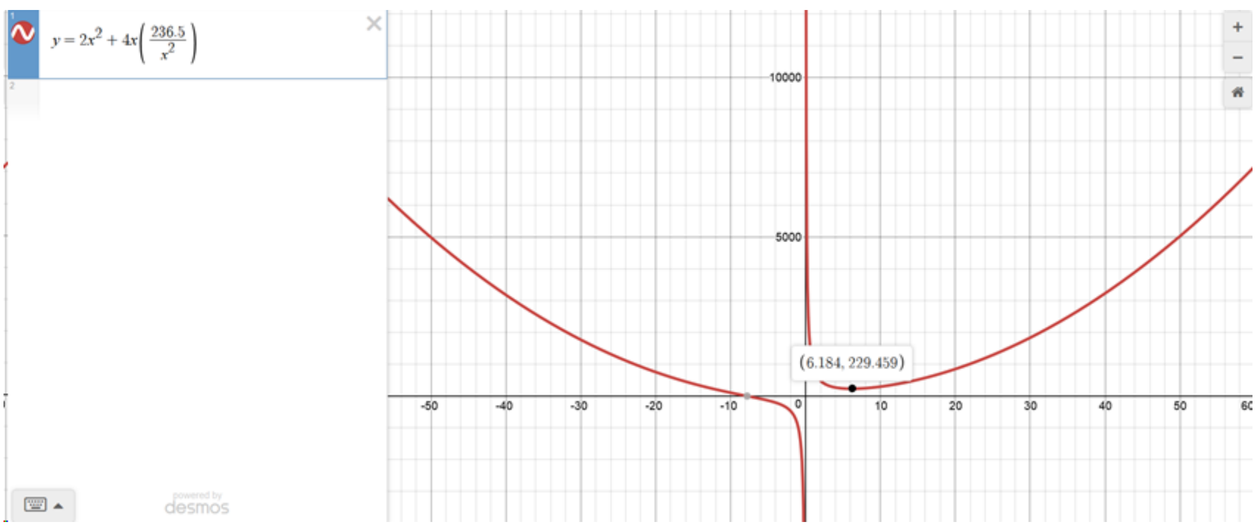


Figure 3 : Height vs surface area graph for a cuboid box of volume 236.5cm^3

We then organized information in Excel and calculated how much material will be needed to construct each container. Notice that some material is wasted to build a closed cylinder. This is since making a circle lid out of a rectangular sheet of metal will waste some material around it while carving.

Table 6 : Amount of material (cm^3) needed to construct each container

Optimized Cuboid				Optimized Cylinider				
Length	Height	TSA	Surface area wasted	Radius	Height	TSA	Surface area wasted	TSA Needed
6.184	6.1843233	191.2173	0	3.351	6.703975	176.4295	9.639228633	186.0687
6.27	6.2702065	196.5697	0	3.398	6.795483	181.3594	9.911518018	191.271
13.104	13.103121	858.528	0	7.101	14.20344	792.1259	43.28450458	835.4104
18.171	18.171618	1650.971	0	9.847	19.6967	1523.265	83.23410262	1606.499
23.205	23.206405	2692.491	0	12.576	25.14989	2484.138	135.76208	2619.9
14.354	14.353492	1030.157	0	7.779	15.55629	950.4499	51.94466727	1002.395
14.354	14.353492	1030.157	0	7.779	15.55629	950.4499	51.94466727	1002.395
5.36	5.3603252	143.655	0	2.905	5.808695	132.5361	7.244121057	139.7802
20.702	20.701267	2142.803	0	11.219	22.43693	1977.022	108.0442656	2085.066
9.043	9.0430122	408.8797	0	4.901	9.799838	377.2356	20.61877364	397.8544
4.523	4.5215627	102.2616	0	2.451	4.901232	94.3522	5.156797151	99.509

We then used that information, to calculate the cost of each container made up of stainless steel. Turns out, cylinders would cost cheaper to manufacture, but not by that much.

Table 7 : Cost of each container

Optimized Cuboid	Optimized Cylinider
Price	Price
4.116487	3.798138
4.231713	3.90427
18.48222	17.05273
35.54178	32.79254
57.9634	53.47803
22.17702	20.4611
22.17702	20.4611
3.092576	2.85321
46.12984	42.56094
8.80228	8.121052
2.201468	2.031195

Next, we tried to optimize the area the structure will occupy. To do this, we did some calculations and noticed that rectangular designs did not waste any space when placed next to one another. The cylinder design however wasted a lot of space between them. As a result, we decided to go for a rectangular design for the bulk portion at least.

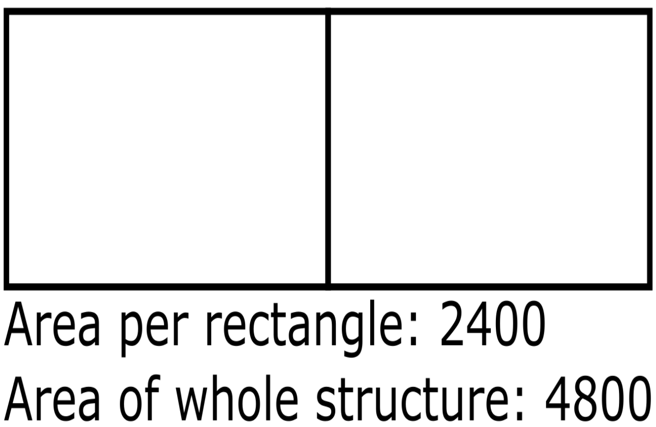


Figure 4 : Area needed for Rectangular Containers

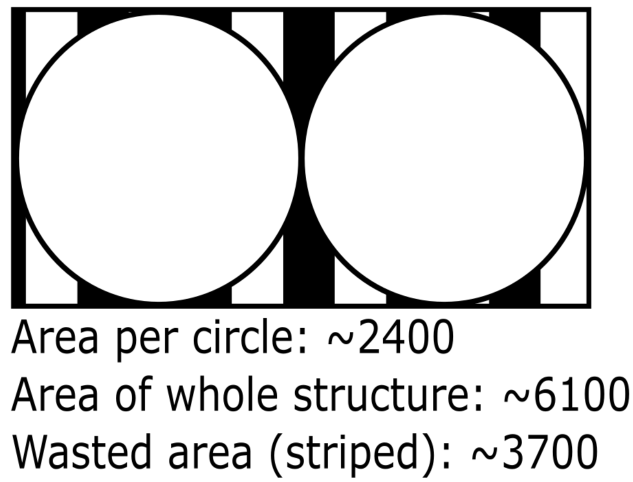


Figure 5 : Area needed for Circular Containers

Even though rectangular designs are more optimized for this scenario, the doors in our portioner would only work with cylinders. Thus, we decided to use cylinders for our portioning part at least.

7 Prototyping, Testing and Customer Validation.

7.1 Prototype I – Customer Review

For our initial customer feedback, we decided to go make a physical prototype to go with our analytical one described above. It was used to show how our first design for the portioning system would work.

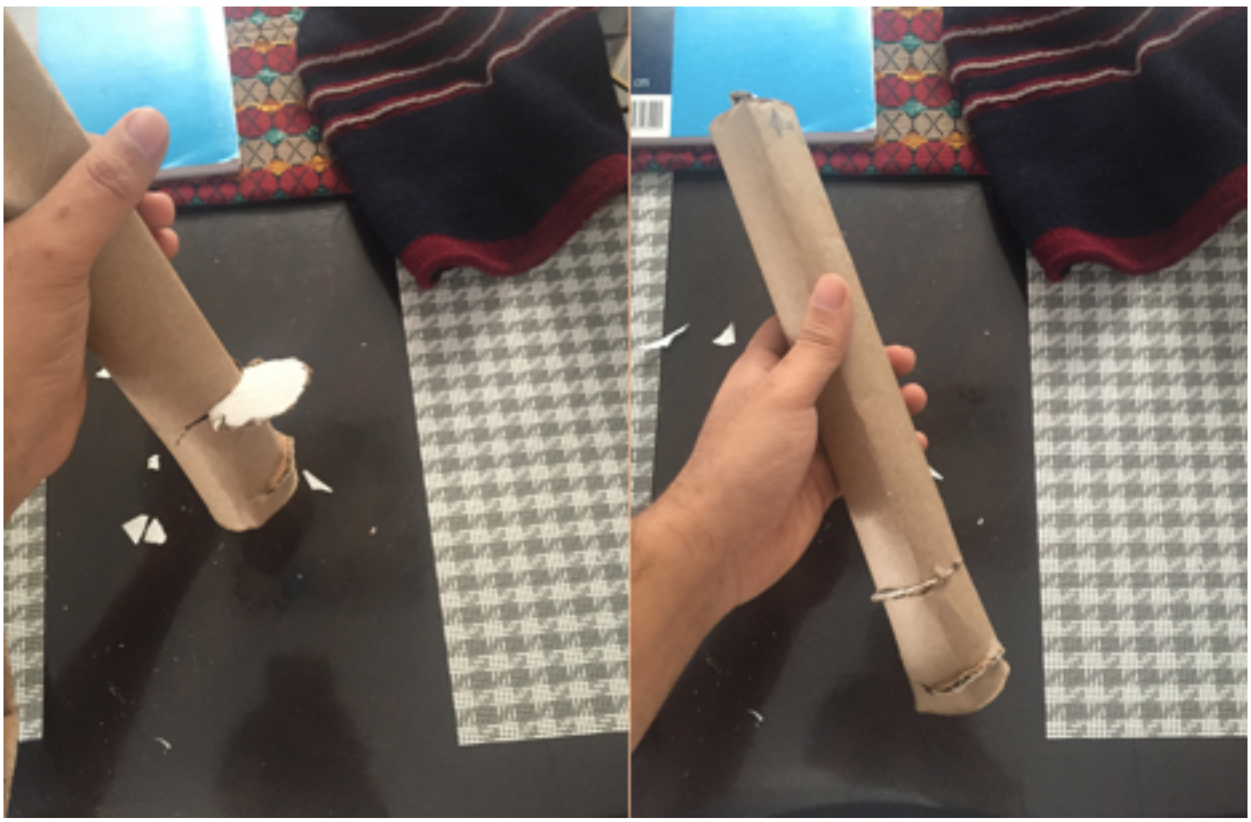


Figure 6 : Physical Prototype I of the portioning part

This prototype was presented to our customer in the second customer meeting. He had some concerns with our design idea, some of them being:

1. Ingredients getting stuck in the pipe system if pipes were indeed used.
2. The washability and replicability of our prototype.

3. The temperature of the ingredients placed directly above the cooking station.
4. The large number of moving parts.

To address some of the concerns that our client had, we decided to go back to our list of sketches and try to formulate a new plan for our distribution system. The figures below show a modified version of the first prototype, which is greatly improved upon, but has a same concept of using volume for measuring portions of ingredients:

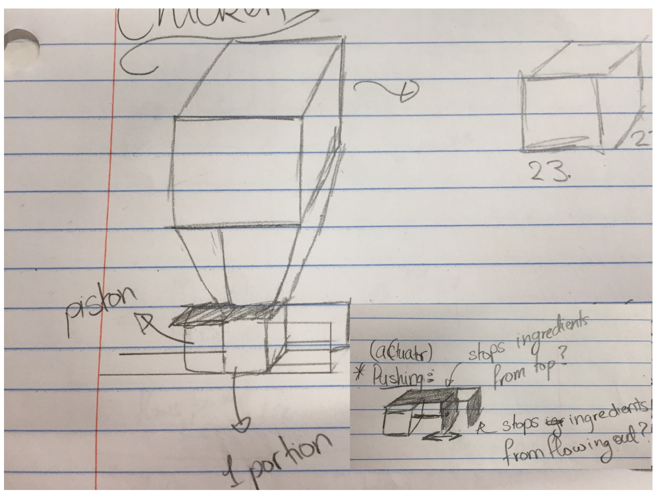


Figure 8 : Sketch of second design

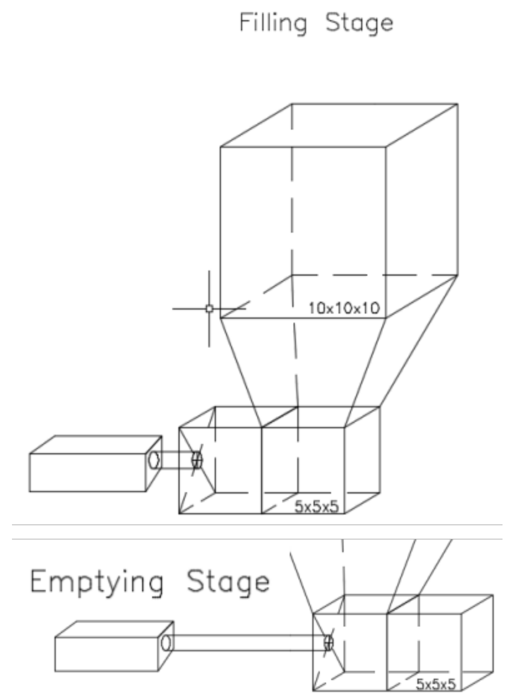


Figure 7 : Computer generated image of second design

As you can see from the pictures above, the new design was fully rectangular and uses only one motor. The bulk is fixed at the top and a funnel leads it to the portioning mechanism. The mechanism consists of two cubes and a linear actuator. The cube on the right has a hollow top and bottom, so the ingredients can fall using the gravity force. The cube on the left has six solid sides to prevent food from pass through. There are two main stages for this design:

1. Filling stage: When the hollow cube on the right is below the funnel, which causes the box to be filled with ingredient.

2. Emptying/Distributing stage: When the right cube is filled with ingredients, it is pushed to the right so the ingredients inside it can be distributed to an appropriate location. The left box covers up the bulk, so ingredients do not spill.

The mechanism switches between the two stages by extending and retracting the linear actuator to portion and distribute ingredients when needed.

7.2 Prototype II

To test out our design, we made a physical, focused prototype. Our Prototype II focused on the portioning part of the design, as we were not sure if the actuator could overcome friction to push box the ingredients and the boxes. We ordered an actuator and 3D printed a box and brackets for this prototype.

The linear actuator we ordered can push and pull using up to 750N of force, moves at 10 mm/s speed, and can extend to up to 5 cm. It also consumes 12V of energy, so we needed an adaptor if we wanted to use wall power. The figure below shows the linear actuator bought from amazon.com.

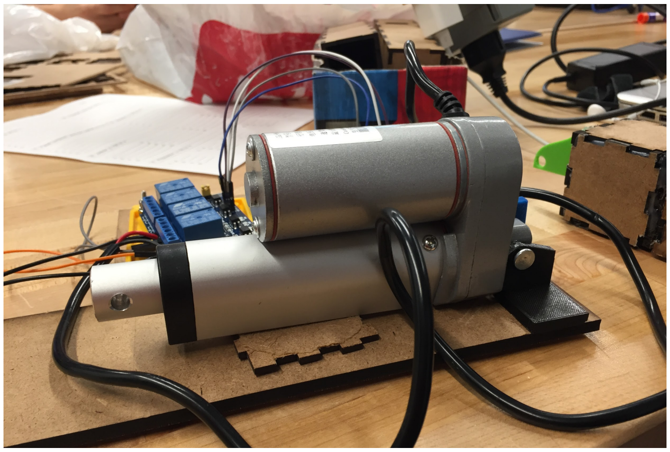


Figure 9 : Linear Actuator

Figure 10 : Testing Prototype II

The test was successful, and we found out that the actuator could easily overcome the friction force and dispense even the heavy ingredients.

7.3 Prototype III

After being reassured by our Prototype II, we decided to start constructing Prototype III. We decided to make a comprehensive, physical prototype to ensure that everything works well and is tested before the Design Day.

7.3.1 Portioner

First, we needed the portioning container to portion precise amount of ingredient for cooking. We decided to first make the box out of sheet metal, where the top and bottom faces are hollow to make it look more professional. The box did not work well because it was hard to bend the sheet to make a perfect cube. It turned out to be wobbly, as it was not level. Figure 7 below shows this problem.

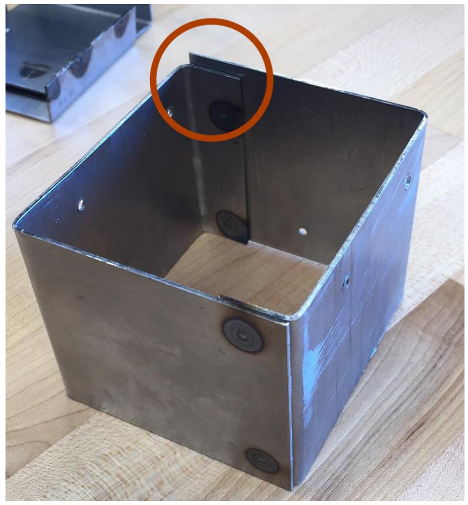


Figure 11 : Computer generated image of second design

After the steel box failed, we decided to consider different ways to make a box and pick the best one for the final design. The figures below show the different portioners that were tried and the final portioner that was used for the prototype.

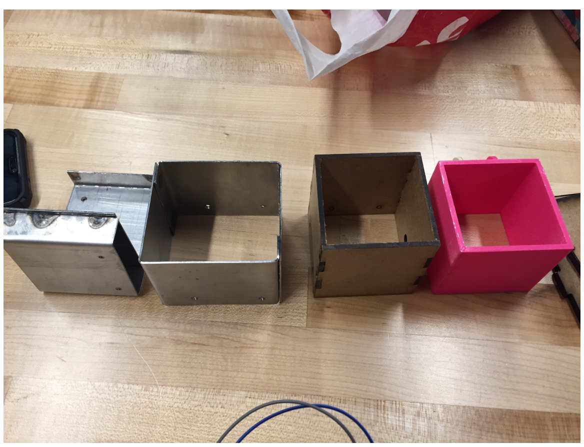


Figure 12 : First Portioner Prototypes

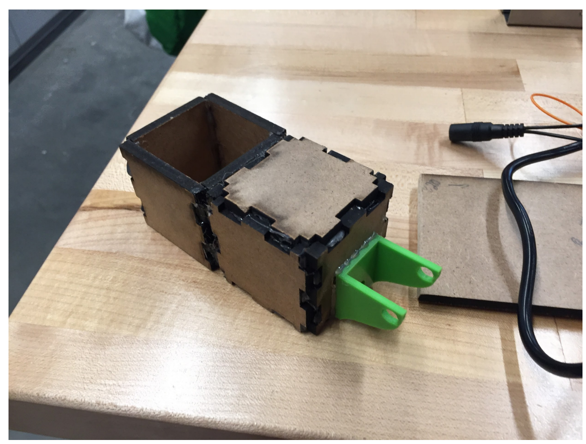


Figure 13 : Final Portioner Prototype

We tried 3D printing a box right after the steel box (Right most box in Figure 9). It did not work because the linear actuator would only extend by 5cm, and the 3D printed box was 6x6x6cm – it was too big. It takes many hours to 3D print a huge design and we did not have that time to spend, so we had to look for better alternatives.

We tried laser cutting next. It was significantly faster than 3D printing as the manufacturing time is greatly reduced. Figure 10 shows the final portioner prototype. It is 5x5x5cm box (~½ cup), so our prototype would be able to portion and distribute about half a cup of any solid ingredient.

The green bracket was 3D printed to attach to the linear actuator and the box together as the stock brackets were too large to be used.

7.3.2 Controlling the Linear Actuator

We used an Arduino Uno to control the actuator, and a relay to reverse the polarity of the circuit, i.e. complete the circuit to extend or retract the actuator as needed. It was powered using a 12 volts AC adaptor. The wiring looked something like this:

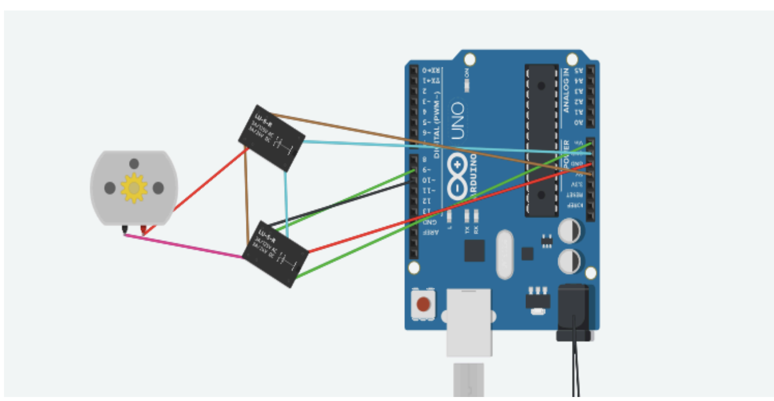


Figure 14 : Arduino Uno Wiring

This image was generated in Tinker Cad. As you can see, there is an Arduino Uno, two black components that represents the relays that we used, and a motor, which represents our linear actuator. The Arduino Uno controlled the motor; the coding for it can be found in the Appendix IV below. The actual relay wiring looked like this:

Figure 15 : Relay, without wiring

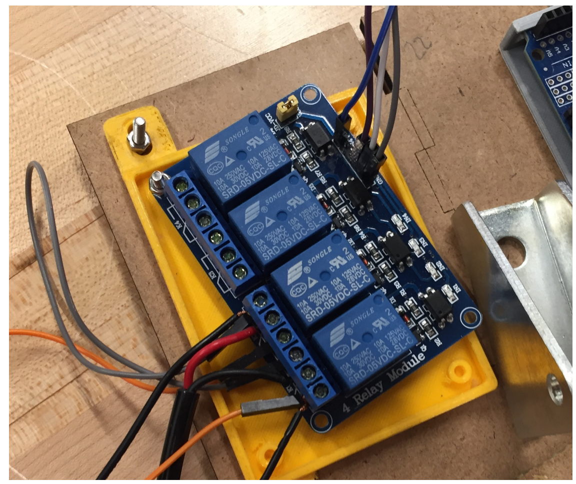


Figure 16 : Relay, with wiring

Here, relay helped control one thing: the direction of the current flow. When relay is 'off', then the COM is connected to NC. The relay is 'on' when the COM is connected to NO. There are two such connections, and it was setup such that if both are connected to the same thing, then the actuator won't move, and when one is 'on' and the other is 'off', diodes in the relay will help determine the direction of the current flow, and determine whether the actuator will move forward or backward.

7.3.3 Scrapped Ideas

We also wanted to implement wheels to reduce friction and save energy, but that would lead to a new problem. Wheels would leave space between the bottom edges of the box and the surface below it. Therefore, we decided to drop that idea.

Figure 17 : Computer generated image of second design

7.3.4 Testing Prototype III

Testing Prototype III was extremely useful as it gave us lots of feedback on how our mechanism could be improved.

A long trail of ingredient was being left behind by the portioner and ingredients kept getting stuck between the holes of the joints.

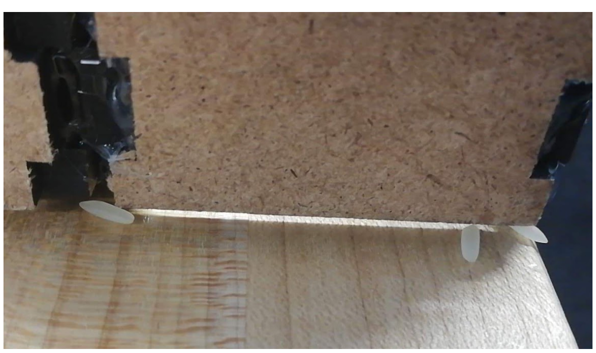


Figure 18 : Trial of Ingredient

Figure 19 : Food stuck between the joints

While these two problems might seem to be unrelated, they both stem from the same issue. We accidentally manufactured the boxes using a material which is thicker than the design specified. As a result, our portioner was leaving some space between the table and itself, allowing food to escape, and our portioner had empty spaces between its joints, allowing food to get stuck between. Making a new box with a material of correct thickness will definitely solve the issue.

Our biggest problem, however, was the landslide effect. Our portions were consistently bigger than they were supposed to be, and it was because the funnel was not fully closed during transition period.

Figure 20 : Inconsistent portions

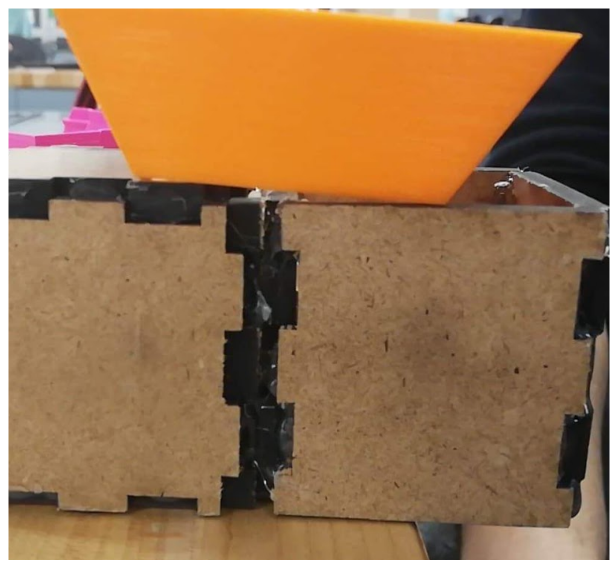


Figure 21 : Funnel not fully closed during transition

When the actuator was extending to dispense the food to the container, the opening in the funnel was not fully closed and as a result, food could flow freely through the hollow cube to the container until the closed cube closed the opening off completely.

To solve this problem, we would need another motorized mechanism (like a door), to close off the funnel opening while the hollow box is extending and retracting. Due to time constrains, however, this feature could not have been implemented.

8 Final Solution

After constructing and testing the portioning mechanism all set, we decided to make the rest of the system. It consisted of:

1. Bulk to hold the ingredient;
2. Support to support the bulk, and;
3. Base to hold the Arduino Uno, relay and linear actuator together

We made the base by laser cutting MDF and then drilling holes in appropriate places to screw the different components on the base. Brackets were made for the Arduino Uno and relay to prevent damage while screwing things in.

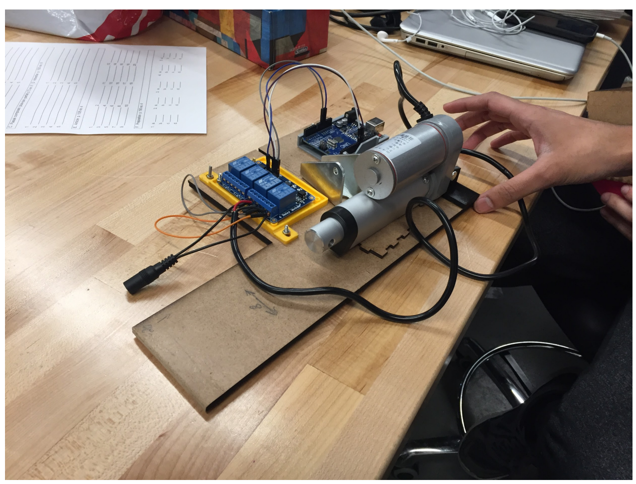


Figure 22 : Base for the Prototype

We made the supports by laser cutting MDF and cutting holes in correct places where the brackets in the bulk would go. The idea is that 4 "legs" would help the bulk stay upright and they would be connected via bracket attached on the bulk.

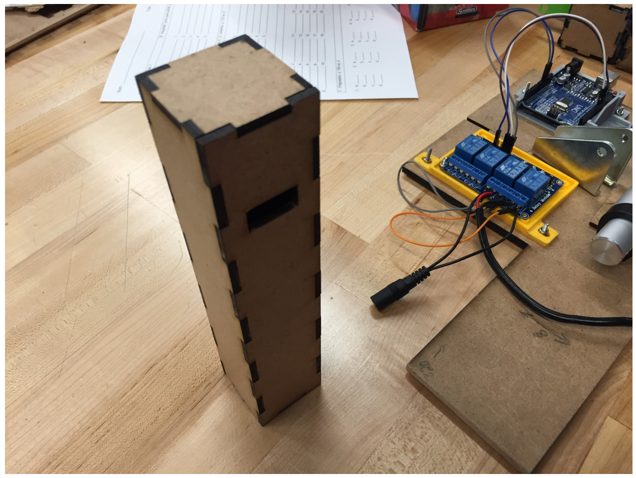


Figure 23 : Support for the Prototype

The bulk consisted of two parts: the bulk part and the funnel part. The bulk part was made by laser cutting MDF as its manufacturing and assembly time was significantly less than 3D printing. We decided to 3D print the funnel however, as laser cutting complex 3D structures proved to be quite a challenge. We attached the bulk and the funnel using hot glue.

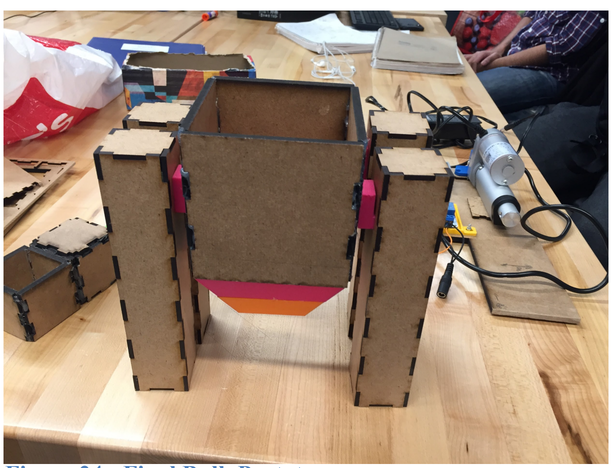


Figure 24 : Final Bulk Prototype

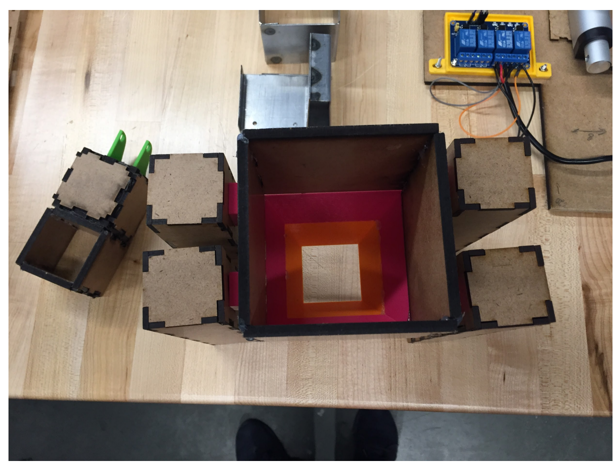


Figure 25 : Final Bulk Prototype, Top View

Below are some pictures from Design Day. Although it may not look like much, we put a lot of work in our design and are proud of our final product. While it is true that our design could use some improvements to perform even better, we feel like the current design tackles all the customer's problems and could be used to make the restaurant more efficient, after the motorized door for the funnel is implemented.

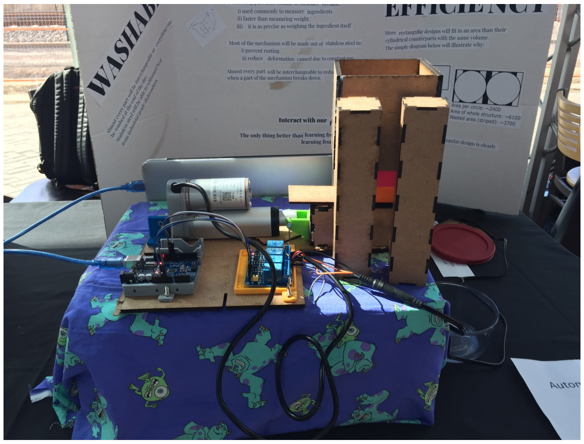


Figure 26 : Final Prototype (Design Day)

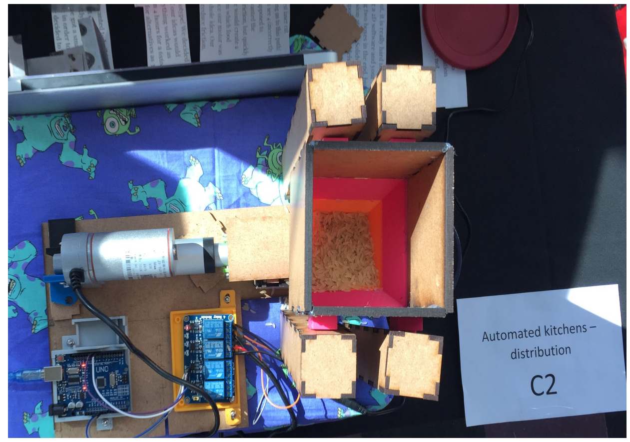


Figure 27 : Final Prototype (Design Day), Top View

9 Conclusions and Recommendations for Future Work

To conclude, we designed a machine that is capable of portioning bulk food for our client. While working on that design, we focused on 3 different features, which are as follows:

1. Washability: All the parts that we used in the project are removable to ensure that sanitization process is easier for employees.
2. Reliability: Our mechanism has only one moving part, the horizontal movements of a linear actuator. Our actuator can portion and distribute 1/4th cup of an ingredient in 11 seconds per cycle. Using only one mechanical part lets us minimize the percentage of failure of the mechanism while in service.
3. Space efficiency: We used rectangular designs to save the most amount of space when placing multiple mechanisms side by side.

Our prototype was satisfactory in terms of the three features listed above, but it was struggling to distribute precise amount of portions for each ingredient. This was due to the Landslide Effect (see 7.3.4 Testing Prototype III) and with the help of a motorized door, even this problem could be avoided completely.

10 Bibliography

1. “Culinary Excellence Elevated By Technology”, Spyce
2. Josh Constine, “Taste test: Burger robot startup Creator opens first restaurant”, Tech Crunch, July, 2015
3. ko_boku , “Nagoya's FuA-Men Robo Ramen”, Ressuns, August 31, 2009
4. Aqua-Calc, AVCalc LLc, 2019
5. Jake PA, “How To Use Relays To Control Linear Actuators”, Progressive Automations, August 21, 2015.

11 APPENDICES

11.1 APPENDIX I: User Manual

1. Capabilities: All parts of our product are washable and replaceable.
2. To wash or replace any part, unhook the hinges and do what needs to be done.
3. To combine the product, you don't need any glue, everything is connected physically. The whole system should look like Figure 26 and Figure 27 in Final Solution after the combination.
4. List of parts:

Linear Actuator (1x)

Portioner (1x)

Bulk (1x)

Supports (4x)

12V adaptor (1x)

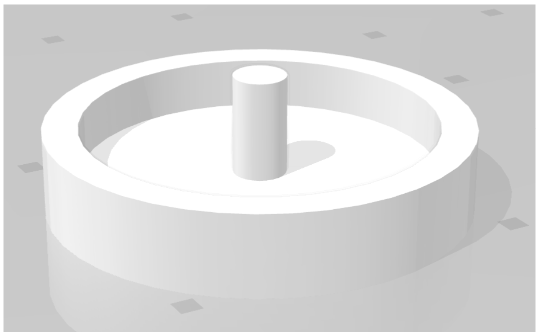
Arduino Uno (1x)

Relay (1x)

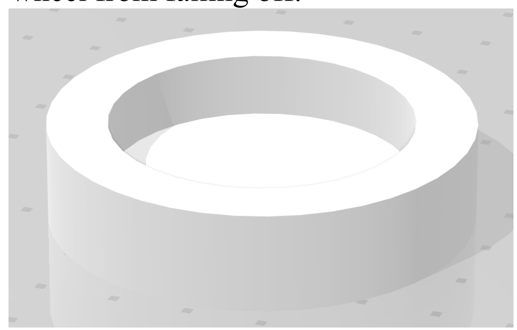
5. Use Section 7.3.2 (Controlling the Linear Actuator) in this document to do the wiring.
6. Upload the code from Appendix IV to the Arduino Uno. Edit as necessary.
7. Once the setup is completed, insert the ingredient in the bulk.
8. Press the red button on Arduino Uno to portion the ingredient.
9. Refer to this video to get a better understanding on how this product works.

11.2 APPENDIX II: Design Files

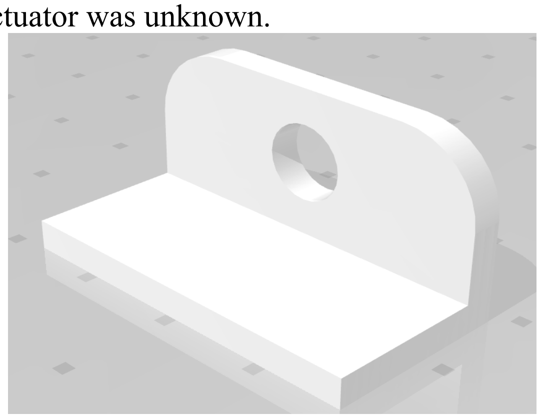
Wheel.stl: A 3D model of the wheel.



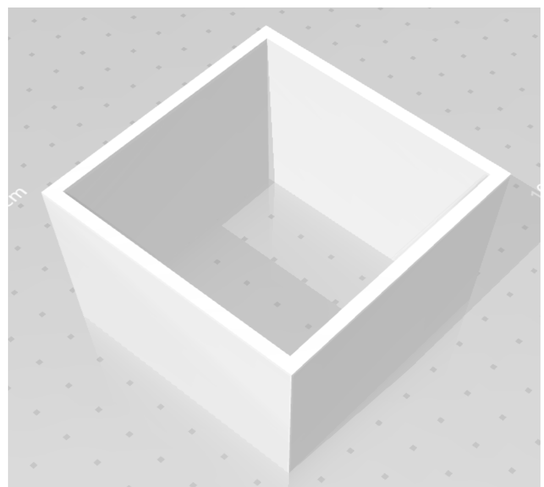
Cap.stl: File for a cap which would be attached to the axle of the wheel, to prevent the wheel from falling off.



bracket.stl: A preliminary design for the bracket. Two of them were to be printed to attached to the box. Was done using this method as the actual diameter of the linear actuator was unknown.

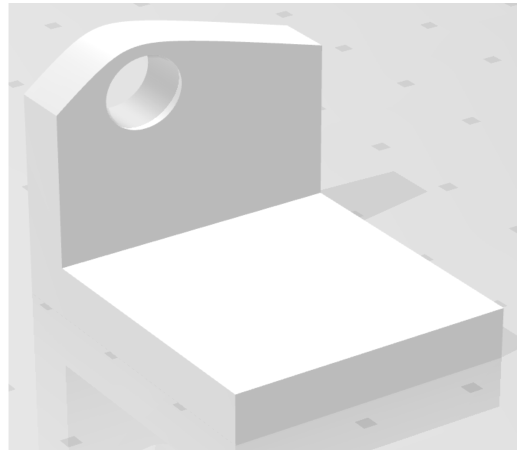


box.stl: The 3D model for the 6x6x6cm box.

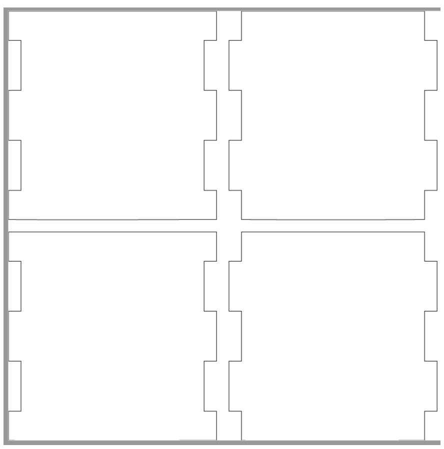


NewBraket.stl: The final bracket to attach the box to the actuator.

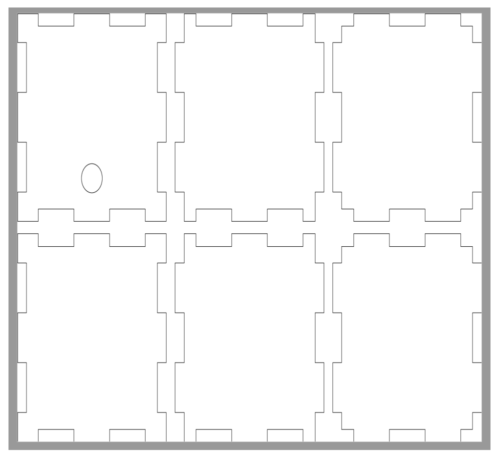
BracketForActuator.stl: The final bracket to attach the actuator to the base.



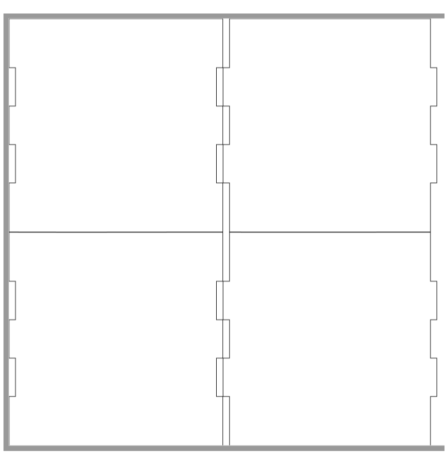
HollowBox.svg: The design for the hollow cube which has to be laser cut.



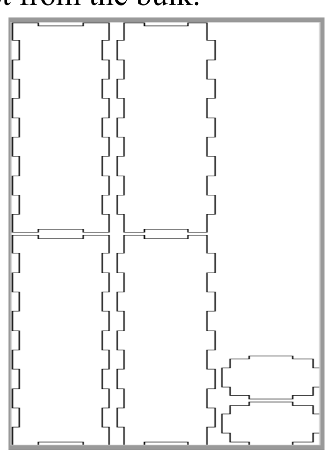
CompleteBox.svg: The design for the closed cube which must be laser cut.



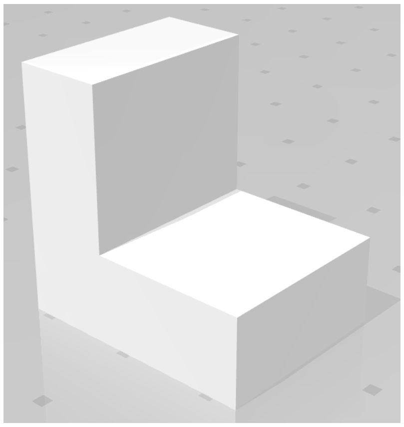
BulkBox10.svg: The design for the bulk box which is 10x10x10 cm.



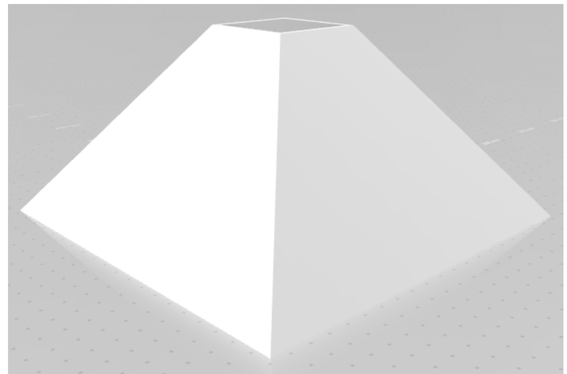
box.svg: The design for supports to hold up the bulk. It is missing the hole which accepts the bracket from the bulk.



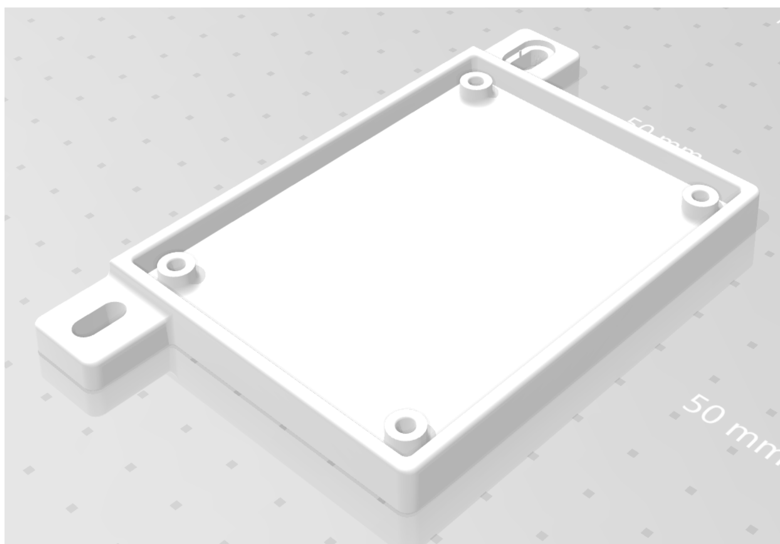
BracketForStructure.slt: Bracket which is to be attached to the bulk. It connects the bulk to the support.



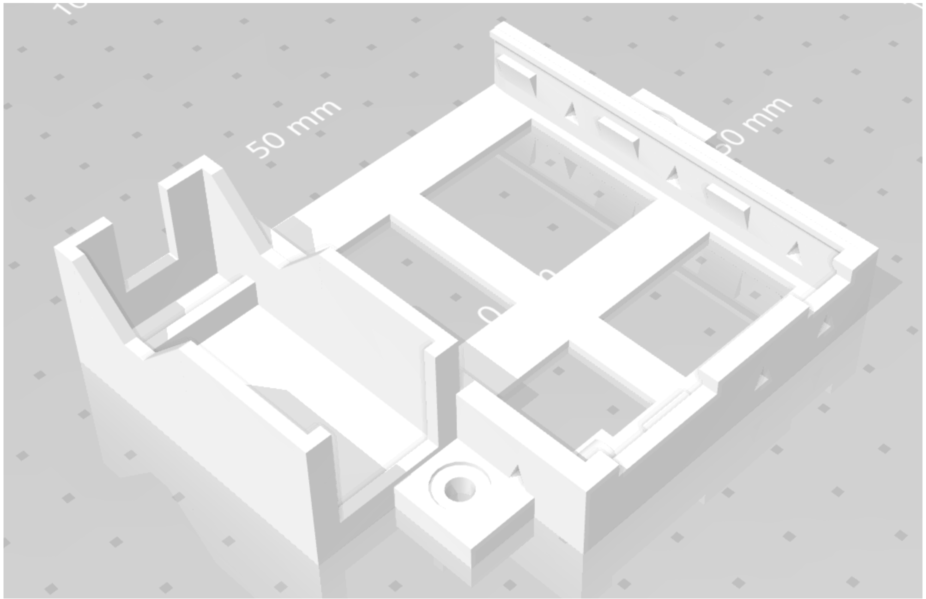
RectangularFunnel.slt: Funnel which connects the bulk to the portioner. 3D printed because laser cutting complex structures is hard.



RelayMount.slt: A mount for our 1x4 relay; the design was picked up from an open source design website.



BaseV5.slt: A mount for our Arduino UNO; the design was picked up from an open source design website.


50 mm

11.3 APPENDIX III: Sketches

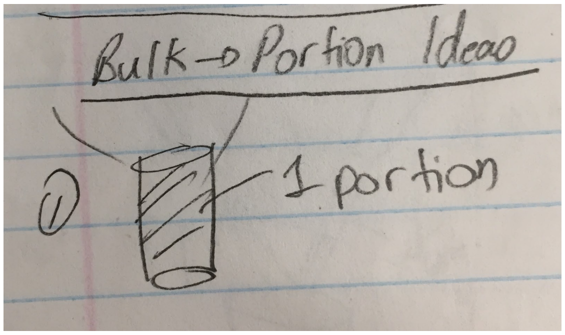


Figure 28 : Idea for Bulk to Portion using Pipes

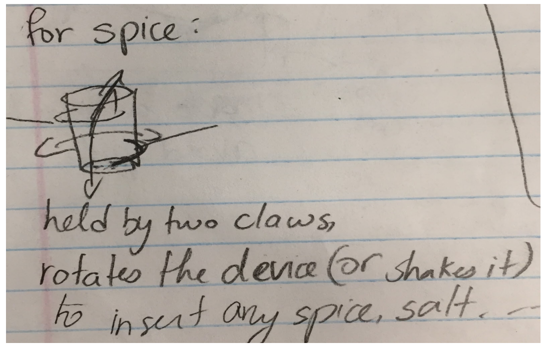


Figure 29 : Portioning spice and salt ideas

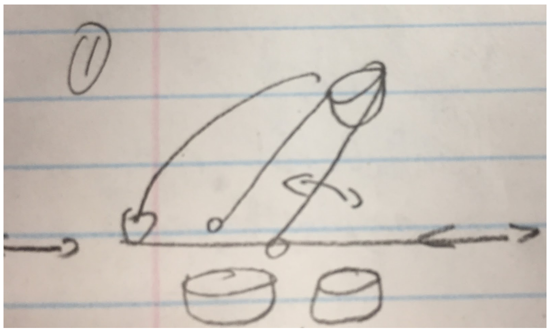


Figure 30 : Transporting Ingredients using Catapult

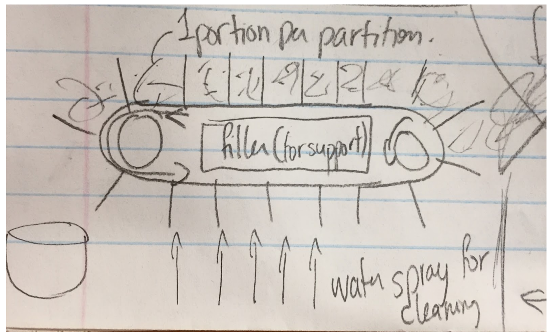


Figure 31 : Treadmill Idea for Portioning

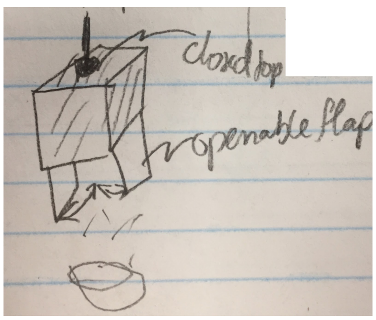


Figure 32 : Idea for distributing portioned ingredients

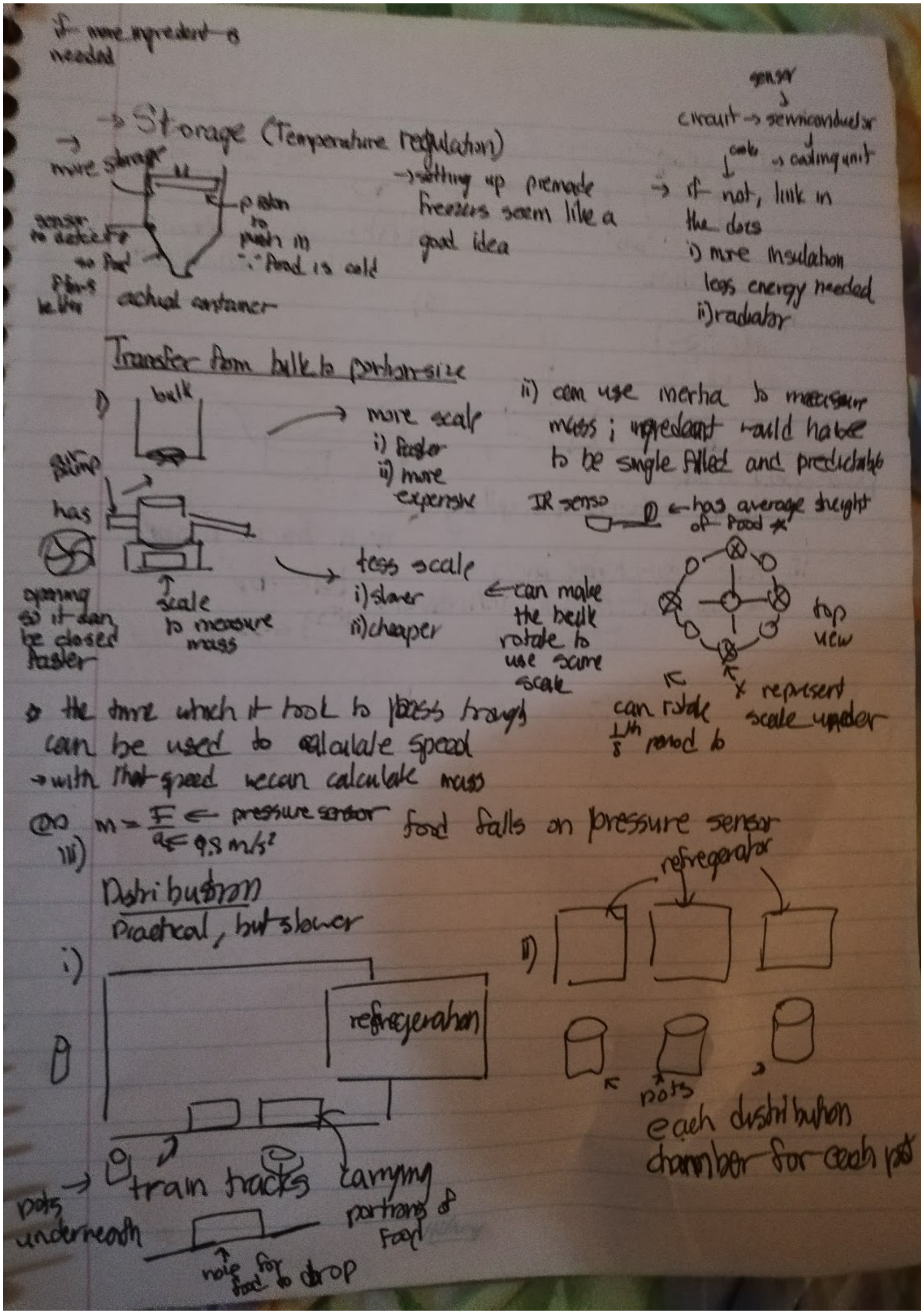


Figure 33 : Sketches for designs

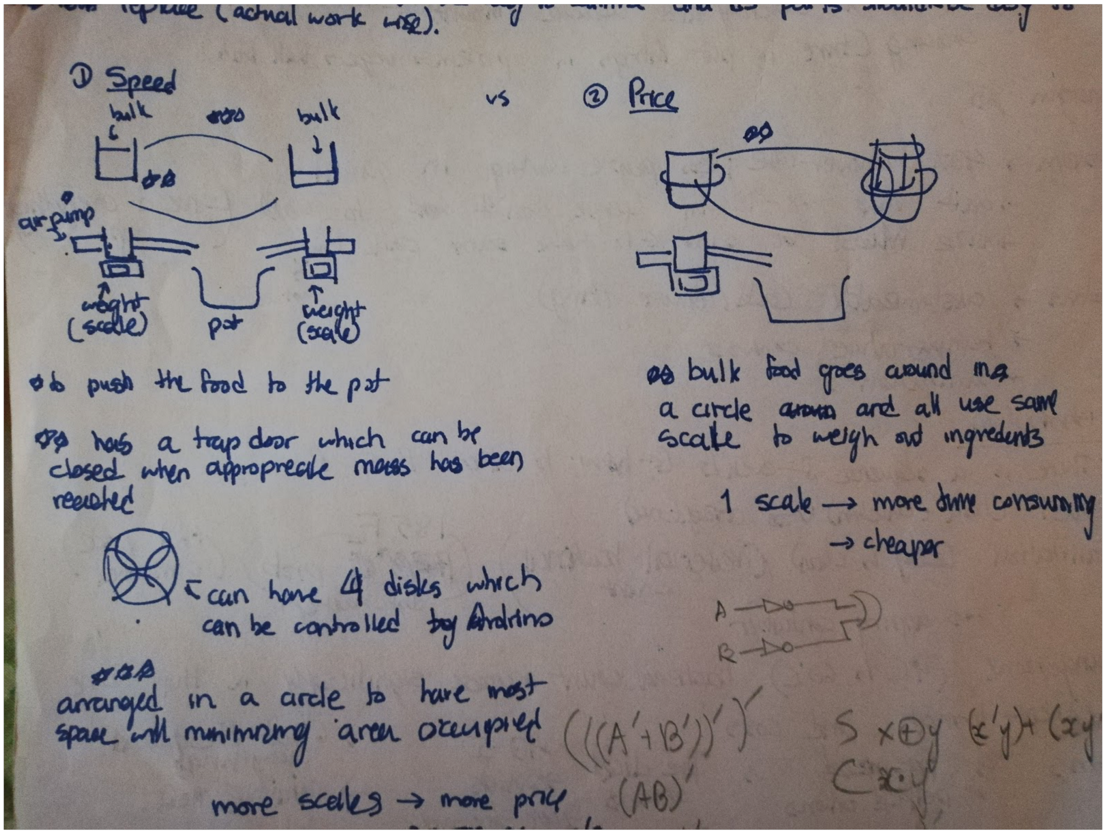


Figure 34 : More sketches for designs

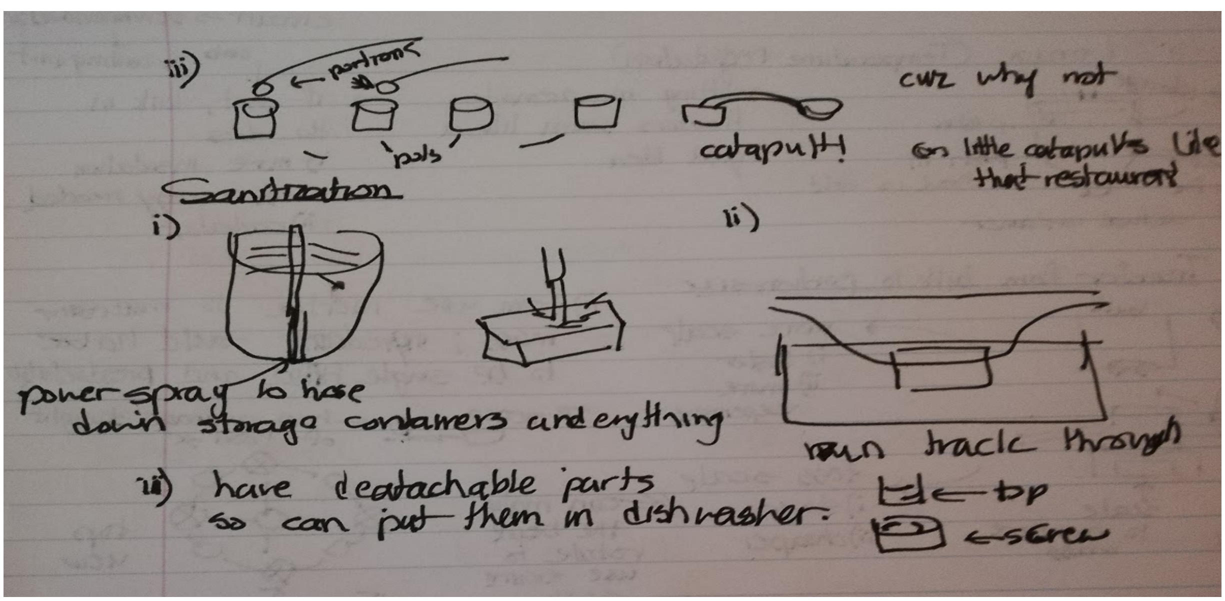


Figure 35 : Even more sketches for designs

11.4 Appendix IV – Coding

```
/**
Title: Coding for Portionator 2.0 by Group C2
Source: https://www.progressiveautomations.com/blogs/how-to/how-to-use-relays-to-control-linear-actuators
There are two versions of the code; one version is commented. The commented version loops infinitely, but if the code is run as is, the linear actuator should run 'loop' twice.
*/


int pinActuator1 = 9; //actuator will be set to pin #9
int pinActuator2 = 10;


void setup() {
  // put your setup code here, to run once:
  pinMode(pinActuator1, OUTPUT);
  pinMode(pinActuator2, OUTPUT);


  int i;
  int numberOfPortions = 2;//for now


  for(i=0;i<numberOfPortions;i++)
  {


  delay(1000);
  digitalWrite(pinActuator1, HIGH);
  digitalWrite(pinActuator2, LOW);
  delay(5000); //assuming the speed of the thing is 10mm/s
              //and it has to travel 5mm
              //so t= 5/10 = 0.5s = 500ms
  digitalWrite(pinActuator1, LOW);
  digitalWrite(pinActuator2, LOW);
  delay(500);//wait until everything drops. can always change.

  digitalWrite(pinActuator1, LOW);
  digitalWrite(pinActuator2, HIGH);
  delay(6000);
  }

}
```

```
void loop() {
 /* // put your main code here, to run repeatedly:


  digitalWrite(pinActuator1, HIGH);
  digitalWrite(pinActuator2, LOW);
  delay(6000); //assuming the speed of the thing is 10mm/s
               //and it has to travel 5mm
               //so t= 5/10 = 0.5s = 500ms
  digitalWrite(pinActuator1, LOW);
  digitalWrite(pinActuator2, LOW);
  delay(500);//wait until everything drops. can always change.

  digitalWrite(pinActuator1, LOW);
  digitalWrite(pinActuator2, HIGH);
  delay(6000);


  digitalWrite(pinActuator1, LOW);
  digitalWrite(pinActuator2, LOW);
  delay(500);
*/
}
```