INTRODUCTION TO INFORMATION TECHNOLOGY

***Analysis***

The local animal shelter is asking for a low cost, programmable automated pet feeder.

**Product requirements**

* Low cost
  + To make it low cost this system will use a rudimentary Arduino system made of servo motor, weight sensors, led lights, IR sensor, RTC real time clock, and alarms/buzzers. This excluding the case of the pet feeder (pet food, bowls, etc), with the materials from the table below the system should cost for one system would be $42.41

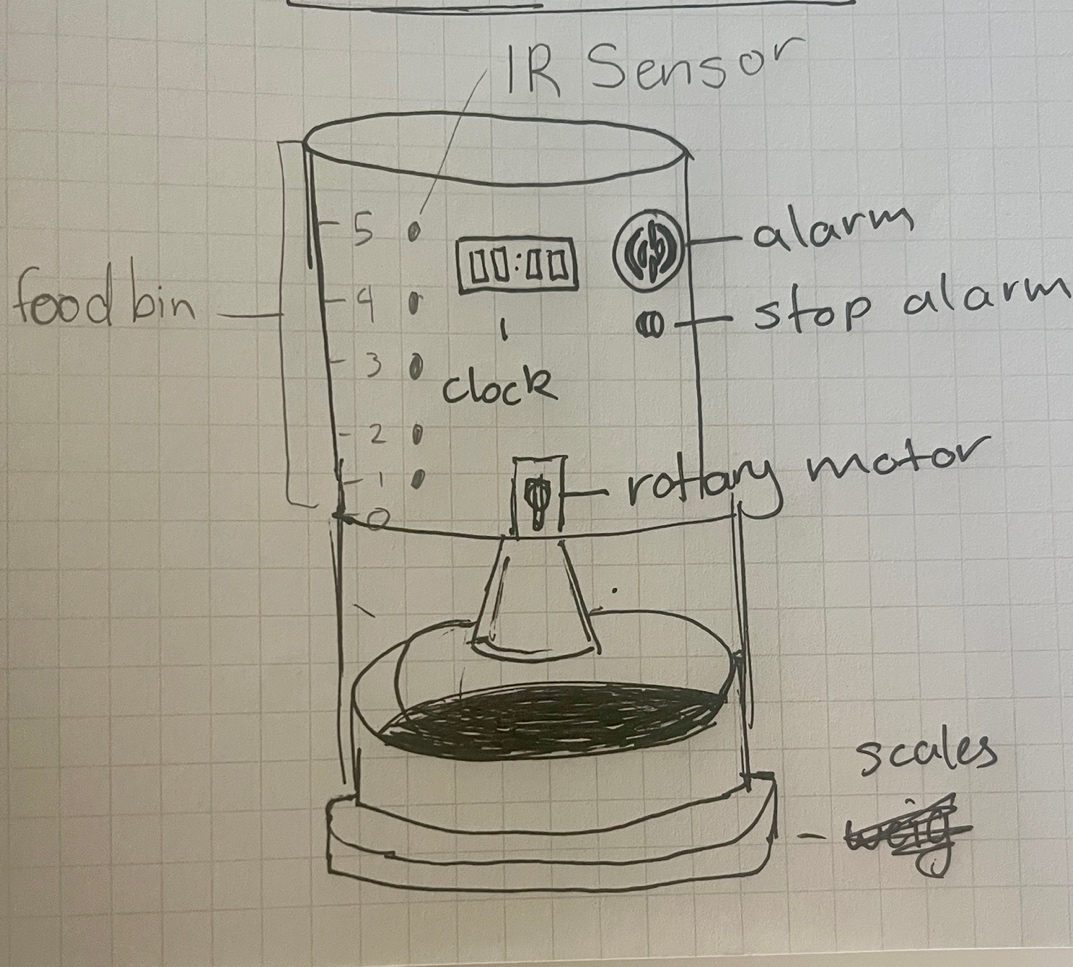
|  |  |
| --- | --- |
| part | price |
| Arduino | $11.10 |
| Weight sensor | $3.95 |
| Buzzer/alarm | $4.99 |
| RTC | $5.44 |
| light | $3.25 |
| servo motor | $11.95 |
| IR sensor | $1.73 |
| total | $42.41 |

* Programmable
  + The use of an Arduino makes the system programmable and adaptable to changing situations. You can change the time of feed or even the conditions of which the staff are alerted by the feeding system. Arduino is easy to use and cheap to replace if it breaks
* Automated pet feeder
  + The Arduino system will make it automated
* Dispense food for cats and dogs at scheduled times.
  + The system will need to be created into two separate areas or be replicated for two rooms. There might also need to be slight variation how much the server rotates to release the food due to size variations in each animal’s food.
  + The system will have two set times for food dispensing. At 8am and at 6pm
* Monitor whether food has been consumed or the amount of food that has been consumed.
  + The system will do this by weighing the food once it dispenses then weighing it again after 10 minutes. If the weight has changed the system will take note by how much and store that information for someone to access and take not of if needed.
  + If the weight hasn’t changed the system will alert the staff.
* Alert staff if there’s an issue (e.g., no food dispensed, food not eaten).
  + This will be specified in the code using the weight sensor at several points during the task.

**Assumptions/limitations**

* Limited memory
  + This could be an issue, but the Arduino could just be programmed to wipe the memory every 2 weeks.
* Different pet food types
  + Cats and dogs have different food types. Cats do usually eat wet food over dry but that’s an entirely different problem. The pebble size difference between cat and dog dry food could cause an issue. This would be fixed by adjusting the angle the servo is rotated and how long it is. Testing would be done to survey the time and angle and how much food is released to get the desired amount.

**Sketch**

****

**Organise and Describe the Data**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type | Name | Description | Sample values | Operational constraints |
| input | RTC | Tracks current time to trigger scheduled feedings | 08:00  18:00 | Must be synchronised and support 24h format |
| input | Weight sensor | Measures food weight in bowl before and after feeding | 0g, 200g, 50g | Accuracy ±5g used to detect dispensing and consumption |
| input | Food level sensor | Detects if food is available in the food bin | Level 1, level 5 | Must be checked before dispensing. Triggers refill alert |
| Output | Servo motor control | Rotates to dispense food | Rotate 90º, rotate 180º | Rotation angle based in portion size |
| Output | Alert system | Sends notification if issue detected | Led ON, buzzer, message sent | Accidently triggered |
| Output | Feeding log | Records timestamps and status of each feeding | 08:00 success, 08:00 alert | Limited storage |