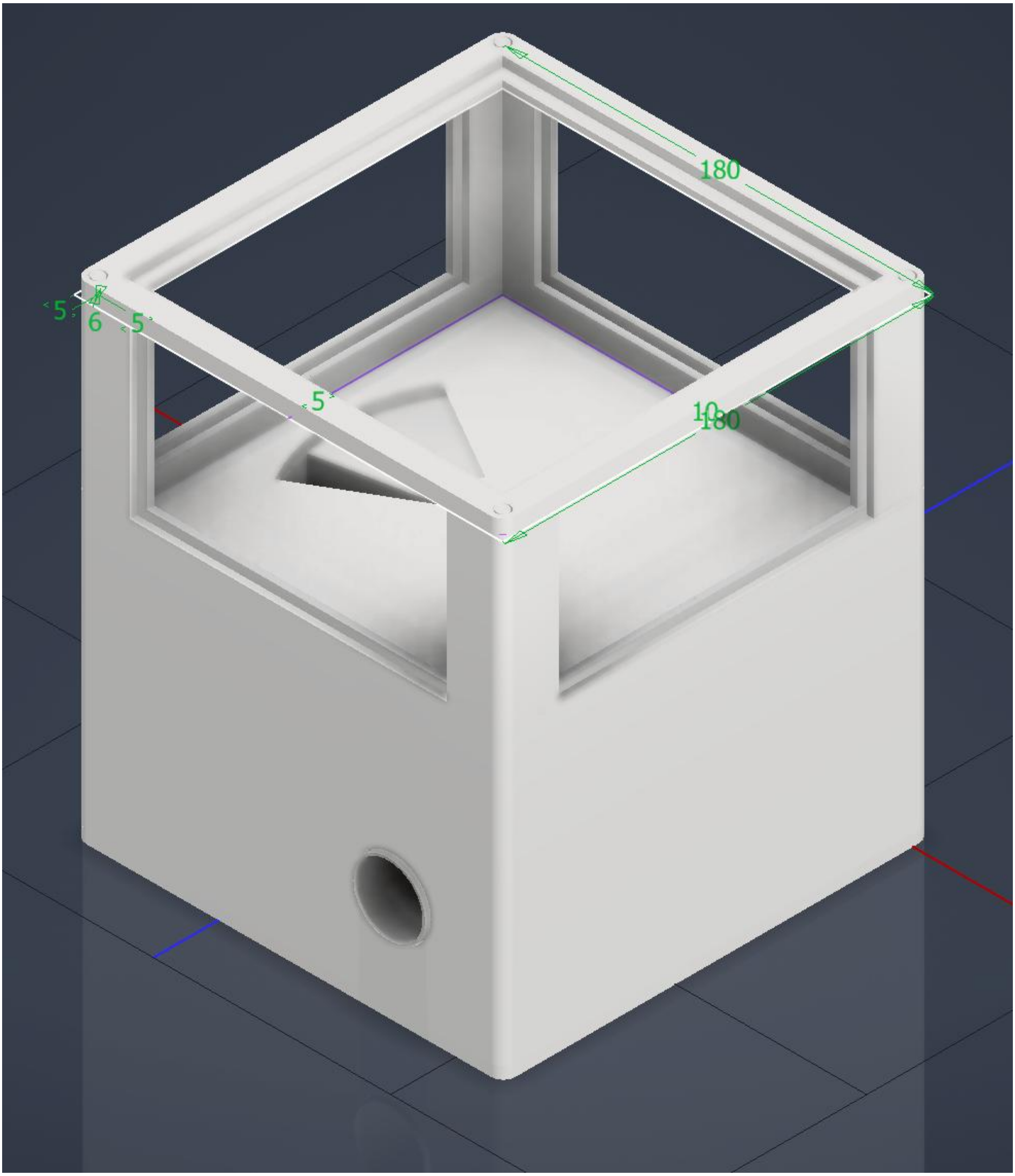

Creating an automatic pet feeder for my dog

Abstract

This project explores the design and development of an automatic dog feeder with the aim of reducing stress for my family during the morning and ensuring accurate portion control. I conducted research into existing feeder designs, nutritional requirements of dogs, suitable materials and electronic components. For the food I chose dry-expanded kibble due to its low moisture content, and consistent pellet size, allowing for easy dispensing.

For the microcontroller I used a Raspberry Pi Pico which using Bluetooth connects to a mobile app built with React Native. The housing and internal components were designed in Autodesk Inventor and manufactured using 3D printing, with laser cut acrylic panels used in some sections for food visibility. Testing showed an average portion output on a single rotation of about 50.5g with a ±2.5g variation. Mechanical reliability testing showed it to be dependable with only minor blockages which were cleared by tipping the machine, suggesting small design improvements required for future versions.

Introduction

Meet my dog

This is Copper [Figure 1]: he is a cockapoo and currently around seven months old. Ever since my family and I brought him home, his diet has been one of our main concerns. From stopping him eating grass and leaves in the garden to choosing the right dog food in shops, feeding him has often been a challenge. For pets, food is also one of the main ways they interact with the world around them. Due to this, it's important they get the right amount and type of food to support them in their development.

The Problem

In my house, the morning is normally quite busy, with everybody rushing to get ready for work and school. Feeding Copper is one of many tasks squeezed into the morning routine and often one of the most time-consuming. A product which could feed him automatically without intervention would reduce the stress.

Another challenge is making sure Copper gets the right amount of food every time. Since we put his food directly in the bowl without measuring, it's difficult to have consistent portions. This product would make sure he gets the same amount each time, maintaining a balanced diet.

Project Aims

There are six main things I want my product to achieve:

- It should be **user friendly,** with the controls being simple and intuitive to use, allowing every member of my family to change settings with ease.
- The amount of food dispensed per meal should be **easily adjustable**, allowing the portion size to increase as copper grows.
- The **feeding schedule should be customisable**, allowing the user to change the number and the timing of meals throughout the day.
- The mechanisms should operate with **limited noise** so that Copper is not disturbed during the day.
- The product should be **easy to install** requiring little to no technical knowledge to set up.
- The feeder should be **visually appealing** and blend in with the kitchen environment.

Research

Ideas and Influences for my Design

I took inspiration for my design by looking at existing products, Brunel University's *Journeys Fuelled by Ideas* (2013) highlights many student designs. One design that caught my attention was a toothpaste dispenser [Figure 2]. Unlike other dispensers, it is wall-mounted, keeping it out of the way which is an approach I like and would consider for my artefact. It uses four screws, allowing for easy installation, which is something I included as an aim. Additionally, its clean, cylindrical shape has no protruding parts keeping it minimalistic and out of the way another quality that I aimed to achieve.

The Spruce Pets. (2025) showcases nine of the most popular cat feeders [Figure 3]. Overall, all of them go for a basic black or white colour scheme; this is likely so they can work in many household environments. The majority of the products have some form of buffer with the bowl in front of it, I will likely do a similar thing as it seems to be a very tried and tested design. I also included a feeder one of my friends has [Figure 4] which also follows a similar design however this one has two bowls as they have two cats.

Feeding requirements of dogs

Nutrient Requirements of Dogs and Cats (2006) [Figure 5] provide a detailed overview of the different dietary needs of pets, helping me choose the appropriate food that I can easily store and dispense. According to the National Research Council (NRC) (2006, p. 656)," Dogs differ from cats in that they are not strict carnivores," which means they can tolerate a wider variety of ingredients, including plant-based ingredients. This is important for my feeder because it shows that a dog does not require perishable raw meat in every meal, allowing the feeder to use dry food while still being nutritious enough

NRC, they go on to highlight two main categories of pet food, canned and dry. Canned foods are "high in moisture (usually 74-78 percent)" making them prone to spoilage and bacterial growth once opened. Their wet, sticky texture also makes them difficult to dispense mechanically. This means that although canned food can be nutritious, it's not particularly good at long term storage without refrigeration, this is something that would be difficult to do in my feeder so something I want to avoid.

Dry pet food can be divided into several groups based on moisture content: dry-expanded, semi-moist, and soft-expanded. The NRC presents dry-expanded food as the most stable option. It "typically contains 10–12% final moisture content” and is usually made from “cereal grains, soybean products and animal by-products.” Most dry-expanded foods are produced through extrusion cooking, which both cooks the ingredients and shapes them into pellets, giving each piece a consistent size and shape. Extrusion also gelatinises the starch, improving digestibility. The consistent size is useful for the design of my feeder as it ensures reliable portioning and reduces the risk of blockages in the dispensing mechanism. The low moisture content also means the food can sit in the machine for long periods without spoiling

The NRC also includes a table showing the energy requirements between breeds [Figure 5]. Although Cockapoos are not listed choosing a breed of similar size provides a reasonable estimate of their nutritional needs.

Material Analysis

When I first considered what material to use for my dog feeder, plastic immediately came to mind. This was due its common use in the kitchen and food related products, however, it is important to use the right type, one which is durable, food safe and environmentally friendly.

According to Monica from Global Pets (2023), “besides the feel of the material, pets are particularly sensitive to smell.” This suggests that using strong chemicals, such as heavy varnishes or harsh cleaning products, could discourage a pet from interacting with the feeder. For Copper, it is important that the product feels safe and approachable so that he is comfortable using it.

Monica also highlights that water-repellent materials as being commonplace in kennels because they are easier to clean ensuring hygiene. In contrast, furry materials which have become more popular in recently years tend to trap mites and dirt, making them harder to keep clean. Designing the product with removable parts would further improve ease of cleaning. Using durable, washable materials would also extend the product’s lifespan, allowing owners to maintain it in good condition over time.

Animal Friends Insurance (2024) give many examples of pet safe materials such as “organic cotton, natural rubber, bamboo, hemp, and jute”. These materials provide sensory benefits to animals as they can scratch and chew them without injury or damage. Hemp and jute are typically quite common in cat trees and scratching posts. They also highlight different chemicals that are toxic to both pets and humans, which should be avoided.

- **PVCs** - These are soft and easy to work with but often contain toxic additives "(e.g. phthalates)" which should not be consumed.
- **Formaldehydes** - A group of chemicals known to cause cancer. Urea-formaldehyde is used as the glue used in MDF, a type of manufactured board.
- **BPA** (bisphenol A) - Commonly used in food and drinks containers, it has recently been banned by the EU in food contact materials due to it disrupting hormones within the body.
- **Arsenic** - Sometimes "found in squishy toys that contain foam", again known to cause cancer.
- **Rawhide** - Contain harmful chemicals from processing and creating of the product, including many already listed above, making it unsafe for pets.

Animal Friends recommend looking for safety and sustainability labels on products [Figure 6].

McCubbin, K. (2025) from Greenshop Paints explains that pets often much more sensitive to chemicals within the home compared to humans. This is because of their "sensitive senses of smell" and the fact they spend a lot more time close to the floor. "Their delicate respiratory systems can suffer from chemical exposure". Truly safe pet products should contain "low or zero Volatile Organic Compounds (VOCs)" and be free from formaldehydes as previously stated.

Dogs may also chew on painted surfaces, potentially ingesting harmful chemicals in the process. Finishes should include pet safe ingredients such as "natural wood oils and resins", which can be used in place of "conventional tins you'll find in most DIY stores" which are normally petrochemical based.

Development

Electronics

Choice of Microcontroller

When choosing a microcontroller for my feeder, I am considering the Arduino [Figure 8] and the Raspberry Pi Pico [Figure 7]. According to Xecor (2024) the Arduino is "renowned for its user-friendly hardware and software" and has an intuitive IDE, making it ideal for beginners. It also "offers dual power options either via a dedicated power jack or its USB port," allowing it to run from mains or battery. Arduino has 14 digital I/O pins and an expansive ecosystem of shields, which can add extra functionality.

However, the Pico has more processing power, which can “handle complex tasks in the background” such as the timekeeping for dispensing food. It supports 26 multifunction GPIO pins and multiple communication protocols like “I2C, SPI and UART,” allowing easy connection to motor controllers and other modules. The Pico accepts “1.8–5.5V” for power making it more power-efficient compared the Arduino which uses “7-14V”, it also has “two officially endorsed languages, C and MicroPython”. While third-party expansion boards “hats” exist for the Pico, they are less widely available than Arduino shields. Overall, “all these functionalities are packed into a $4 board,” making the Pico a budget-friendly and powerful option. The Pico is also smaller, and more lightweight and smaller allowing it to fit in tighter spaces

While the Arduino is user-friendly and well-supported, the Raspberry Pi Pico’s greater processing power, low energy consumption, and additional GPIO pins make it the better choice for the feeder, as it is compact and low cost while still being able to handle complex tasks.

User Interface

For the user interface, I am using a phone app. This makes the system easy to access and configure without needing to physically go to the feeder. The app communicates with a microcontroller inside the feeder, which monitors the time and controls the motors that dispense food. The type of connection is important because the setup process must be simple. I originally planned to connect the feeder to my Wi-Fi, but this would require entering passwords and selecting networks, increasing complexity. Instead, I chose Bluetooth. It is widely supported on phones and requires minimal configuration. The phone connects directly to the feeder’s controller with no middleman [Figure 9].

To develop the app, I am using React Native, which allows apps for both iOS and Android to be built from a single codebase. It is also a framework that I have a lot of experience with, making it well suited for this project. For Bluetooth communication, I am using the react-native-ble-plx library to scan for and connect to a device named “Dogfeeder”. On the microcontroller side, an HM-10 module handles BLE communication.

After setting up the project and installing the required libraries, I modified the AndroidManifest.xml file to include the necessary Bluetooth permissions. This allows the app to scan, connect, and communicate with the feeder [Figure 10].

The initial version of the app contains three buttons [Figure 11]:

- **Connect** – Scans for a device named “DogFeeder” and connects using the HM-10’s service and characteristic IDs.
- **Disconnect** – Disconnects from the device and updates the connection status.
- **Dispense Food** – Sends a test message (“Hello World”) to the microcontroller via Bluetooth.

In future versions, the app will send the current time to keep the feeder’s clock accurate and allow feeding schedules to be uploaded.

Keeping Time

To make sure the feeder dispenses food at the right time I need to use a Real Time Clock (RTC), a specialized chip that just keeps track of time. It can count leap-years and knows how many days are in a month, but it doesn't take care of Daylight Savings Time (because it changes from place to place). They also contain a coin battery to make sure it keeps counting even if the power lost, lasting up to 5 years.

RTCs must be set to the correct time before they can be used, otherwise they start counting from a predetermined point. Whenever a phone connects to the feeder via Bluetooth, it can update the RTC using the phone’s current time. Since the phone gets its time from the internet, this ensures the RTC always stays accurate and up to date. RTCs are fairly accurate, normally keeping a tolerance of 1 to 2 seconds

After looking at many RTCs from multiple companies on electronics websites, I came across the Adafruit PCF8523 [Figure 11] from PiHut. It comes on a breakout board, making it easy to connect to my microcontroller and runs on a relatively low voltage of 3.3 to 5V.

Computer Aided Design

Autodesk or Inventor

Almudhaf’s (2020) study compares Autodesk Inventor and SolidWorks focusing on the usability of each, the learning curve, their file support and use in industry. SolidWorks is described as more intuitive with students finding it “easy to use” allowing designs to be produced in a shorter timeframe, this is likely due the inclusion of “integrated training features and training programs”. Inventor has “a much steeper learning curve” requiring up to 50 hours of training before being confident with the software according to Almudhaf

However, "Inventor can be used with a 3D printer conjunctionally" since my final artefact body is planned to be 3D printed the easy transfer of files between it as the printer is essential. Both programs are windows only software which shouldn't pose an issue as I do have a Windows laptop and do not plan on using a Mac. Inventor is also free for students unlike SolidWorks which does have a reduced licence cost, remaining as its normal £70 a year.

SolidWorks excels in computer simulation and analysis. For example, it offers finite element analysis (FEA) to evaluate stress, strain, and deformation in parts and assemblies, as well as computational fluid dynamics (CFD) to simulate fluid flow and heat transfer. While Inventor also provides some of these tools, they are generally more basic and less refined.

Overall, Inventor is the best choice for my project. My design isn't highly complex, so I don't need the advanced tools that SolidWorks provides. I already have experience using Inventor, and the free student version makes it the most affordable. It provides all the tools I need and allows easy file preparation for 3D printing, which I plan to use for making the parts.

Motor Joint

To allow the motor to spin the mechanism I need a part that can connect to the motor. That holds onto the shaft while still being easily removable in the case there is a blockage or it needs replacing. Instead of finding all the dimensions myself, I found an online datasheet for the motor I'm using [Figure 12], which allowed me to get exact measurements. I used Autodesk to create an initial design, however after printing it I found that the fit was too tight this is due to the slightly inaccuracies of the 3D printer and the expansion of the plastic which I did not account for. It took four iterations [Figure 13] to achieve a design that had a reasonably tight fit while still being removable.

Electronics Base

To keep all the components organised and secure, I designed a base to hold them. I first created separate designs for each component using datasheets I found online [Figure 14], allowing me to test the tolerances and ensure they fitted correctly before committing to a longer print [Figure 15]. Once satisfied, I combined the designs into a single model and began printing. However, during the print the printer's hot end began leaking plastic [Figure 16], forcing me to stop and cancel the print after several hours. I had to disassemble the hot end to clear the blockage, but fortunately none of the parts were damaged and I was able to reassemble it successfully. While examining the failed print, I also noticed a void in the centre of the model that would have caused the print to fail anyway. I corrected this by modifying the CAD design and then successfully printing it [Figure 17].

Dispensing Mechanism

To dispense the food, I plan to use a mechanism like the ones used in gumball machines and other sweet dispensers. Owen J. (2018) demonstrates how these machines work in a video, showing a design that uses a single rotating disk [Figure 18]. The disk contains slots that allow food to fall into them as it rotates. As the mechanism continues to spin, the food is carried around the disk and eventually falls into a chute that leads to the exit.

In some designs, the size of the slots can be adjusted to control the amount of food dispensed. While this could be a useful feature, in my design I can simply rotate the motor multiple times if I want Copper to receive more food. To create this mechanism, I started with the motor joint from my earlier design and added a disk on top of it [Figure 19]. I then created three holes in the disk so that food can fall into them and be dispensed as the disk rotates.

Main Housing

Initially, I was going to print the main housing as a single part [Figure 20], however after doing some research and observations from printing other parts, I discovered I may face warping issues. According to Simplify3D (2026) while initial layers may print correctly “later on the part begins to curl and deform” [Figure 21] They go on to say that this commonly occurs “when printing very large or very long parts” which my design is both.

To reduce the risk of warping and improve overall print reliability, I decided to split the housing into multiple smaller parts. These can then slide together to form the complete structure. Having multiple will also make assembly easier as the internal panels for the dispensing mechanism can be inserted without needing additional opening or slots, improving the overall strength. In total, I used four of these corner components to form the housing [Figure 22].

Prusa3D Forum (2023) also reinforced many of the recommendations made by Simplify3D. In addition, they suggested using products such as “Layerneer Bed Weld” [Figure 23] to improve bed adhesion. This is adhesive applied to the print bed to help the part stick more securely during printing. In practice, this can be something as simple as a standard Pritt Stick. I chose to use this method to ensure the print achieved strong adhesion to the bed and to reduce the risk of warping further.

Funnel and Pipe

After the food leaves the dispensing disk, it needs to go into Copper's bowl in a controlled and reliable way. To achieve this, I designed a funnel and pipe which directs the food downwards into the bowl. It is positioned directly below the rotating disk [Figure 24] so that any food falling from the slots is immediately captured. The funnel [Figure 25] gradually narrows into a circle pipe which helps concentrate the food into a single direction. When designing these components, it's important that the slope of the pipe is not too shallow, so pieces of kibble don't become stuck or accumulate on the sides.

To evaluate the strength of the hopper before printing it, I carried out Finite Element Analyis (FEA). I simulated pressure against all internal surfaces to simulate the flow of dog food pressing against the walls as it flowed through the component [Figure 26]. The material I used for simulation ABS which can be 3D printed. Most of the hopper remained in low stress (shown was blue) with only small concentrations of stress around of the corners and edges. The 100 MPa applied is far higher than the hopper would experience in normal use meaning the part exceeds practical limits.

Food Area

Above the dispensing mechanism, there needs to be a storage area to hold the food. This must be relatively large so that it can store multiple portions, reducing the need for frequent refilling. It also needs to provide visual feedback to the user so they can easily see how much food remains in the hopper and know when it needs to be refilled.

To achieve this, clear acrylic sheets are used on each side of the container, allowing the user to easily see inside. This level of transparency would not be possible with standard 3D-printed parts, making acrylic a more suitable material choice for this component.

The acrylic panels were manufactured using a laser cutter [Figure 28]. The laser cutter allowed the sheets to be cut very precisely to the required dimensions, ensuring they fit accurately within the design. To hold the panels together, I 3D-printed corner pieces [Figure 27]. These components were printed to match the dimensions of the acrylic panels and act as structural supports, allowing the panels to slot together securely.

.

Testing

Portion Accuracy Testing

One of the aims of the project was to ensure consistent portion sizes. To test this, I programmed the feeder to complete one rotation of the rotating disk. The food that was released was then measured using digital kitchen scales and I repeated this ten times.

Trial	Weight (g)
1	51g
2	49g
3	52g
4	50g
5	53g
6	48g
7	52g
8	50g
9	51g
10	49g

Mean: 50.5g **Range: 48g – 53g**

The results show a variation of ±2.5g from the mean. This level of variation is acceptable for dry dog food, as commercial feeders, I often rely on timed motor control rather than weight sensing. However, the feeder does not directly measure weight, meaning accuracy depends on consistent kibble size and density.

Mechanical Reliability Testing

The dispensing mechanism was assessed over thirty consecutive rotations to assess:

- Risk of jamming or blocking
- Motor overheating
- Consistency of rotation

During testing, some of the food became stuck in in the dispensing chute instead of falling directly into the bowl. This occurred when many pieces tried to go down at the same time or the kibble did not gain enough speed.

The issue was minor and was resolved by tipping the feeder slightly forward to allow the food to fall through. Although this did not cause mechanical failure, it highlights a small limitation in the current chute design. Future improvements could include widening or increasing the angle of the chute to improve food flow and reduce the risk of blockages.

Bluetooth Connectivity Testing

To evaluate user accessibility connection tests were conducted at varying distances indoors.

Distance	**Connection Success**	**Time to Connect**
1m	Yes	2 seconds
3m	Yes	3 seconds
5m	Yes	4 seconds
8m (through wall)	Intermittent	7 seconds

The Bluetooth connection was reliable within normal kitchen-to-room distances. However, connectivity became unstable through walls at greater distances. This confirms that Bluetooth is suitable for local control, but does not allow remote feeding outside the home, unlike Wi-Fi-based feeders on the market. The choice to prioritise simplicity over remote connectivity keeps with the original design aim of easy setup.

Noise Level Observation

While the feeder was operating, I assessed the noise it produced. The device generated a low motor sound along with slight scratching from the dispensing wheel against the plastic base, lasting 2–3 seconds per rotation. Copper did not display any signs of distress or avoidance behaviour during testing.

Although a decibel meter was not used, the sound level was comparable to that of an electric toothbrush. In future development, using a decibel meter would provide more objective and measurable data.

Conclusion

The aim of this project was to design and manufacture an automatic dog feeder that could reliably dispense food for my dog Copper while reducing the stress of feeding during busy mornings. Overall, the project was successful, and most aspects of the design specification were achieved.

One of the main aims of the feeder was for it to be user-friendly and easy for my family to use. This was achieved using a mobile app, which both of my parents can install on their phones. The feeder connects to the phone via Bluetooth, meaning no passwords or setup are required. This simplifies the process significantly when compared to alternative methods such as Wi-Fi.

Another important requirement was that the feeder should dispense consistent but adjustable portion sizes. Testing showed that the mechanism, inspired by Owen J.'s gumball machine design, worked reliably with slight variation in the amount of food dispensed. This level of accuracy is acceptable for dry dog food. Portion sizes can be adjusted by rotating the mechanism multiple times, allowing it to increase as copper grows.

The feeder also meets the aim of producing minimal noise. Testing demonstrated that the mechanism only produced a short sound like an electric toothbrush, suggesting that it is suitable for everyday use within a household environment.

The use of Autodesk Inventor played a significant role in the development of the final product. It allowed me to create and assemble complex parts such as the dispensing mechanism and hopper before manufacturing them with a 3D printer. Iterative prototyping was an important part of the process, particularly when developing the motor joint where several versions were required before achieving the correct tolerances.

However, testing also identified some limitations with the current design. Occasional blockages occur in the chute when a large amount of food tries to go down at the same time. While these were minor and could be resolved having the machine at a slight angle. Redesigning the chute at a steeper angel could improve reliability.

Further Research

Currently the feeder dispenses food using timed motor rotations rather than directly measuring the amount of food released. This means the accuracy of a portion depends on the size of kibble pieces and the initial location of the rotating wheel. In future versions I could add a load cell [Figure 29] below the bowl to determine how much food is in it. By connecting this component to the microcontroller, the motor could stop once the desired weight has been reached.

The current product is primarily made from 3D printed components will this has made development a lot easier and quicker however this might not be best suited for larger scale for production. Using injection moulded parts could allow the product to be produced quicker and with higher levels of strength, while reducing costs when manufacturing in large quantities.

Furthermore, I could have designed a PCB (Printed Circuit Board) [Figure 29] allowing the micro and motor controller to sit on the same board, making assembly easier as the different components don't need to be connected with jumper cables and breadboards [Figure 30].

Bibliography

Brunel University & Ashikordi, V. (2013). *Journeys fuelled by ideas*.

The Spruce Pets. (2025). *The 9 Best Automatic Cat Feeders, Tested in Our Lab*. [Online] Available at: https://www.thesprucepets.com/best-automatic-cat-feeders-4175145

National Research Council, *Division on Earth, Committee on Animal Nutrition, Subcommittee on Dog and Cat Nutrition. (2006). Nutrient requirements of dogs and cats. National Academies Press.*

Ferrigno, M. (2023). *Pet-friendly materials*. [Online] Available at: https://globalpetindustry.com/article/pet-friendly-materials/

Animal Friends Insurance (AFI). (2024). *What makes pet products eco-friendly and pet-safe?* [Online] Available at: https://www.animalfriends.co.uk/save-animals-with-us/tails-of-uk-wildlife/tails-of-uk-wildlife-blogs/what-makes-pet-products-eco-friendly-and-pet-safe/

McCubbin, K. (2025). *Pet safe paint – dog safe, cat and animal friendly decorating*. [Online] Greenshop Paints. Available at: https://greenshoppaints.co.uk/blogs/blog/pet-safe-paint-dog-cat

Linn. (2025). *Can TPR material pass food-grade testing?* [Online] Available at: https://www.tpe123.com/can-tpr-material-pass-food-grade-testing/

React.dev. (2022). *Creating a React app – React*. [Online] Available at: https://react.dev/learn/creating-a-react-app

Adafruit. (2017). *Adafruit PCF8523 real time clock*. [Online] Adafruit Learning System. Available at: https://learn.adafruit.com/adafruit-pcf8523-real-time-clock/real-time-clock

Almudhaf. (2020) Design of Automotive engine components by cad software. Autodesk Inventor and SolidWorks a comparative study

Xecor.com. (2024). *Raspberry Pi Pico vs. Arduino: How to Choose the Right Microcontrollers*. [Online] Available at: https://www.xecor.com/blog/raspberry-pi-pico-vs-arduino

Owen, J. (2018). *How does a Gumball Machine work? YouTube*. Available at: https://www.youtube.com/watch?v=Q3ZeUNDg4fQ

Simplify3D. (2026). *Warping*. Available at: https://www.simplify3d.com/resources/print-quality-troubleshooting/warping/

Prusa3D Forum (2023) *Large flat print is warping*. Available at: https://forum.prusa3d.com/forum/original-prusa-i3-mk3s-mk3-how-do-i-print-this-printing-help/large-flat-print-is-warping/

Glossary

Gelatinizes - *When a substance, usually* starch, absorbs water and swells upon heating, forming a thick, gel-like consistency. It's the process by which starch transitions from a granular to a viscous, semi-solid state.

Integrated Development Environment (IDE) – A software application that provides a comprehensive set of tools for programming. These include a code editor and compiler, debugger and sometimes features like code completion. It allows developers to write, test and debug code in a single area.