Project Final Report

Project Identification: Ping-pong Design System

AERO 2001: Aerospace Graphical Design

Group # 7

David Ntwali, 101337251

Jacob Puvan, 101343322

Alexia Savage, 101332163

Kalea Lebenhagen, 101331675

Submission date: 03/31/2026

Table of Contents

1. Introduction

The objective of this design was to create an automated system that was capable of shooting 20 ping-pong balls from a designated starting square through a 6 x 6 in opening in a barrier and into a bucket within that square in under 60 seconds. This project mimicked many widely used engineering systems and what is required in order to manufacture these systems, as it is a small-scale automated transport system. This project demonstrates the challenges that designers face in terms of performance and manufacturing within the constraints of the problem at hand. The report will discuss the design requirements and constraints after explaining the operation of the system, materials used, and assembly process. The code and models used in this project will also be included in order to further explain this system, and finally, a summary and recommendations for improvement will be given.

2. Description of Design

The Department of Mechanical and Aerospace Engineering established performance criteria and physical constraints that the Pong-pass system had to satisfy. The main objective of the Pong-pass system is to ensure that the timing and propulsion of the system are working correctly. All the components are essential to ensure that the launch is done, all within 60 seconds. The design architecture, manufacturing processes, and requirements that characterize our final prototype are carefully considered.

2.1 Requirements and Constraints

Before operating the system, it must be contained within the starting square and the target square; the system cannot be larger than 2 ft x 2 ft, and not be higher than 1.0 feet off the ground. Once started, the system has the ability to move and be controlled electrically through wires or wirelessly, and the bucket must stay inside the target square and on the ground for a ball to be counted. A ball is to be considered inside the bucket if its entire diameter is inside the bucket's opening. Other operational constraints are that the balls can be put into the system by hand to the system or by automation. The material constraints include any alkaline, NiMh, or lithium-ion batteries as long as any part of the circuit does not exceed 12V. The total cost of the project must not exceed 100$.

2.2 Design Functionality (Flywheel & Arduino)

The structural integrity of this propulsion system was made possible through the collaboration of the key areas. The electrical distribution area was the central point of the electrical subsystem, which consisted of the power supply unit rated at 12V. This power supply unit was configured in parallel to enable the simultaneous powering of the Arduino Microcontroller and the Pulse Width Modulator. Although the output of the Microcontroller was rated at only 5V, this was enough for signal communication with the Servo Motor. As depicted in Figure 1 in the appendix, the control algorithm was developed in C++. This was done in the Arduino Integrated Development Environment to control the gate mechanism of the ball feed system.

The code was optimized to enable the consistent rhythm of 1.325 seconds per ball. This was done to guarantee the highest level of repeatability in the requirement of the 20 balls. When this code was analyzed over the entire range of the experiment with the 20 ping-pong balls, it was evident that the mechanical sequences took only 26.5 seconds out of the allotted 60 seconds. This translates into a theoretical value of 33.5 seconds, which was provided by the timer to boost the speed performance points as outlined in the competition rules. The integration of the manual PWM interface was the most important link in the chain that connected the battery and the flywheel motors. This subsystem made use of the potentiometer dial to enable the precise and real-time adjustment of the Revolutions Per Minute of the motors. This was important in the calibration of the exit velocity of the balls passing through the 6-inch aperture without interfering with the accuracy and trajectory of the balls.

2.3 Manufacturing and Assembly

The body of the system was built from wood, which includes the ramps, rails, and supporting beams, which were put together with hot glue, as shown in Figure 8. The ping-pong reservoir was made from cardboard, and the supports for the motors were made from two pieces of wood, being hot-glued on either side as well as hot-glued to the supporting base to ensure stability while operating. The two flywheels and door were 3D printed from polylactic acid; the design used to 3D print can be referenced in section 5 of the report. The flywheels were attached to the motors by applying a small amount of hot glue to the inside of the wheel to ensure the flywheels rotated with the motors. Another key component of the manufacturing was the use of rubber bands on the flywheels to provide friction in order for the balls to be launched by the wheels, as the smooth surface initially provided by the 3D printer did not allow the balls to be grabbed and launched at the speed the flywheels were rotating.

3. Summary and Recommendations

The design architecture, as well as the combination of all the components, demonstrated solid performance during the competition. A core strength of the system was the manual PWM interface, which allowed our team to accurately adjust the flywheel RPM to match the specific height and range requirements. As initially predicted, the RPM change was working accurately, granting us the flexibility needed to attain the target square. Furthermore, the servo-actuated door mechanism perfectly replicated the cycle intended, feeding one ball at a time. Thus, it allowed all 20 ping-pong balls to process the entire cycle without jamming or mechanical failure.

While the project was largely successful in its execution, some challenges were encountered regarding the height and trajectory consistency. The initial height calculations were predicated on a mathematically perfect alignment of the two flywheel motors. However, because the physical mounting was not perfect, the balls could not follow the exact theoretical path. When the motors were set at a higher RPM to reach the required height, the increase in kinetic energy would cause the balls to exceed the target range.

To bypass this problem, the design should incorporate an adjustable angle ramp. If the angle of the ramp can be adjusted, it becomes easier to optimize both the range and height. By treating initial speed and launch angle as independent variables, a Python script could be utilized, inputting a maximum height of 3.0 to 3.5 feet and a range of 4.0 feet to optimize parameters for a perfect result. Furthermore, the choice of an open ramp wasn’t ideal as the ball’s path would deviate. A closed ramp would have proven itself more efficient.

Finally, an orientation change for the motor position would’ve avoided energy loss. Because the flywheels were positioned horizontally, as shown in Figure 8, the balls hit the ramp before ascending, triggering the impulse-momentum principle in a way that created an impulse opposite to the direction of motion. While our use of filleted cardboard reduced this impact, orienting the flywheels parallel to the ramp's incline would minimize energy loss and result in a much smoother launch sequence.

4. References

[1] gpTeacher, “Ping Pong Ball Cannon - Arduino,” *YouTube*, Feb. 03, 2019. https://www.youtube.com/watch?v=cs02VoRqgPg (accessed Mar. 30, 2026).

[2] B. Farmer, “VEX TURNING POINT: FLY-WHEEL BALL LAUNCHER,” *www.youtube.com*, Jun. 07, 2018. https://www.youtube.com/watch?v=ZAiLxv0Xo8U

[3] G. Payne, “Arduino Ping Pong Ball Cannon,” *Hackster.io*, Feb. 03, 2019. https://www.hackster.io/GordPayne/arduino-ping-pong-ball-cannon-3c1fdd

[4] Department of Mechanical and Aerospace Engineering, “Design project description and deliverables,” 1BC.

5. Appendices (Drawings and Code)

AERO2001_Door.ino

```
#include <Servo.h>

Servo myServo; // Create servo object

const int servoPin = 9; // Servo pin
const int closedPos = 90; // Door closed
const int openPos = 0; // Door open

void setup() {
  myServo.attach(servoPin);
  myServo.write(closedPos); // Start when it's closed
}

void loop() {
  // 1. Open the door
  myServo.write(openPos);
  delay(325); // Keep door open for 0.325 seconds to let the ball pass

  // 2. CLose the door
  myServo.write(closedPos);

  // 3. Wait for the 1 seconds
  delay(1000);

  // The entire process is a 1.325s cycle
}
```

Figure 1: Servo Motor Code Controlling the Door

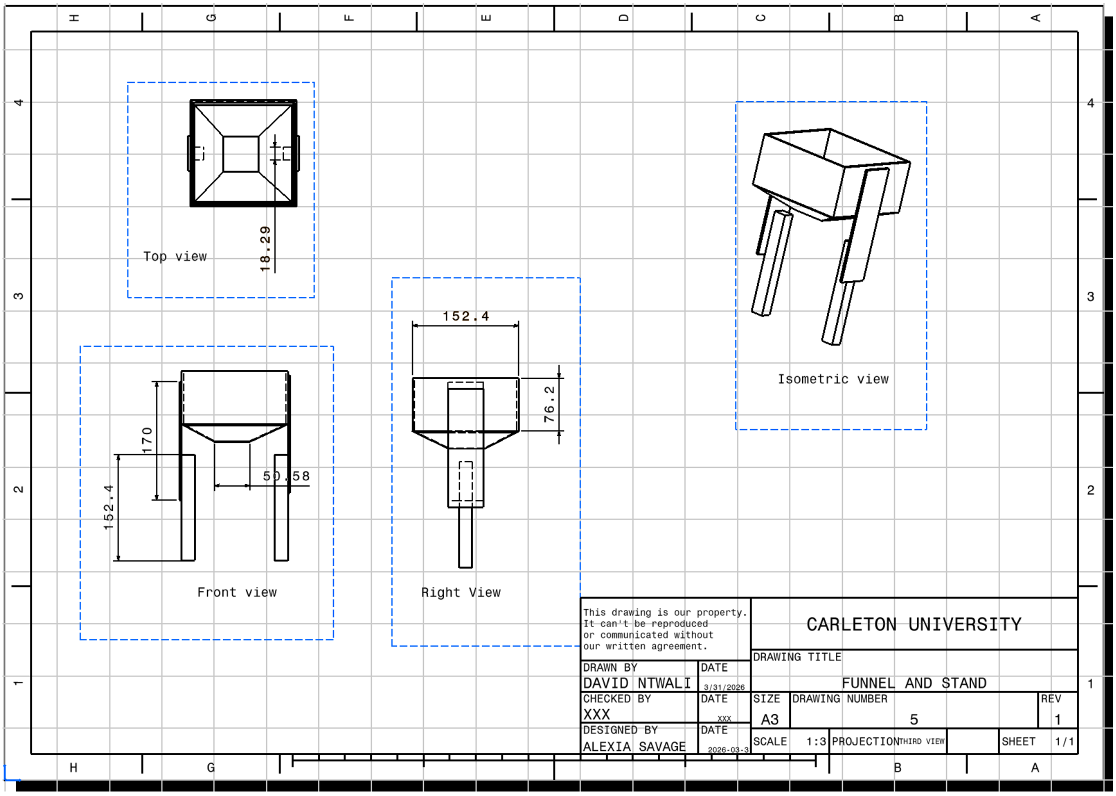

Figure 2: Funnel Bucket & Stand

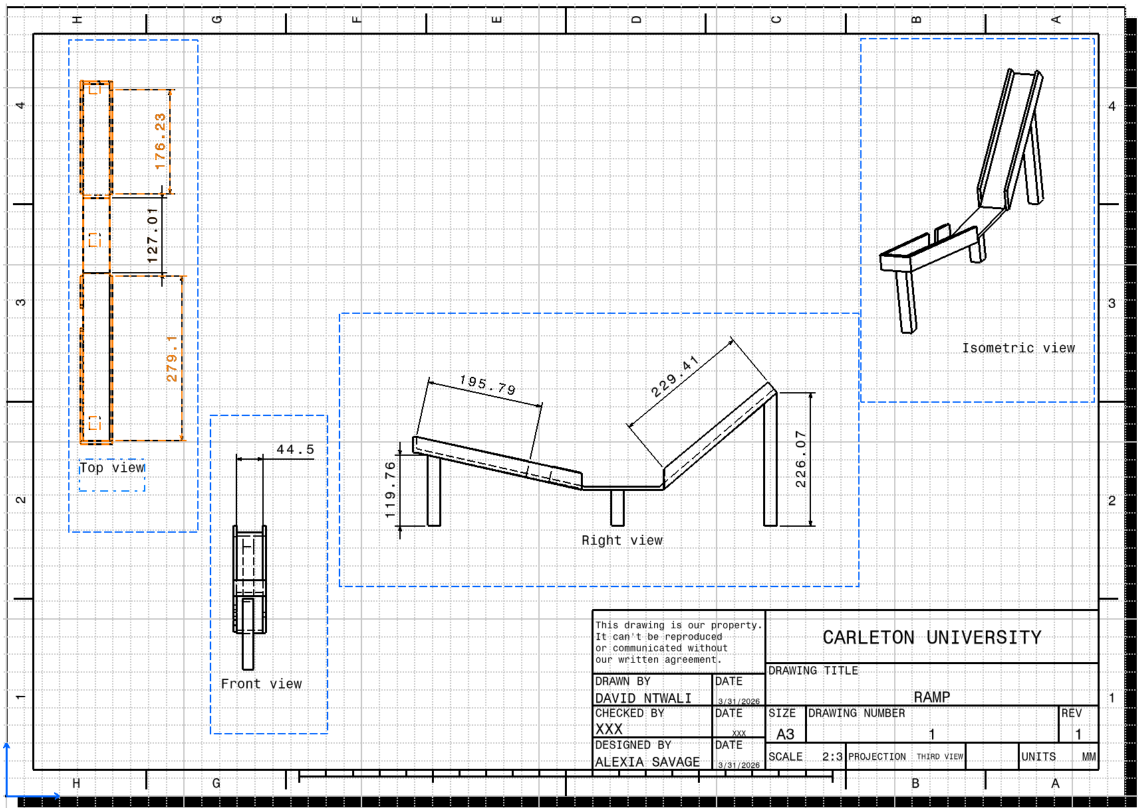

Figure 3: Ramp & Supports

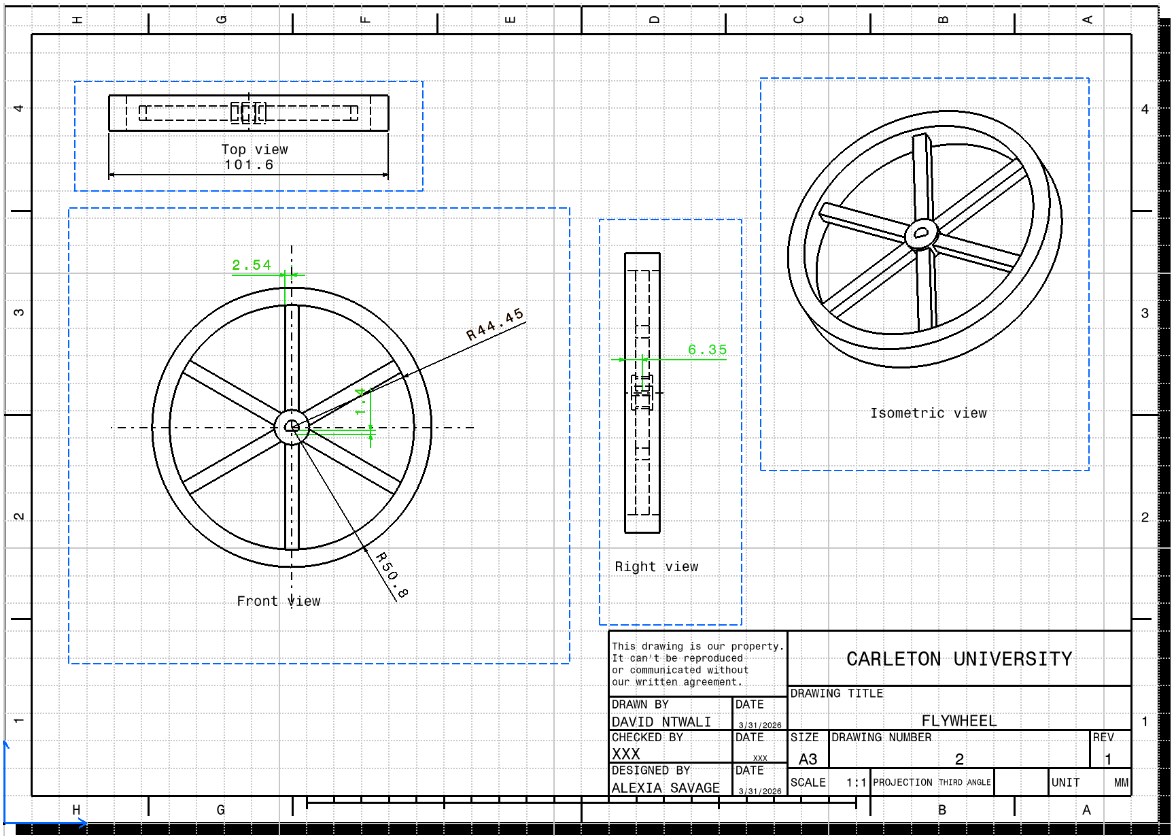

Figure 4: Flywheel

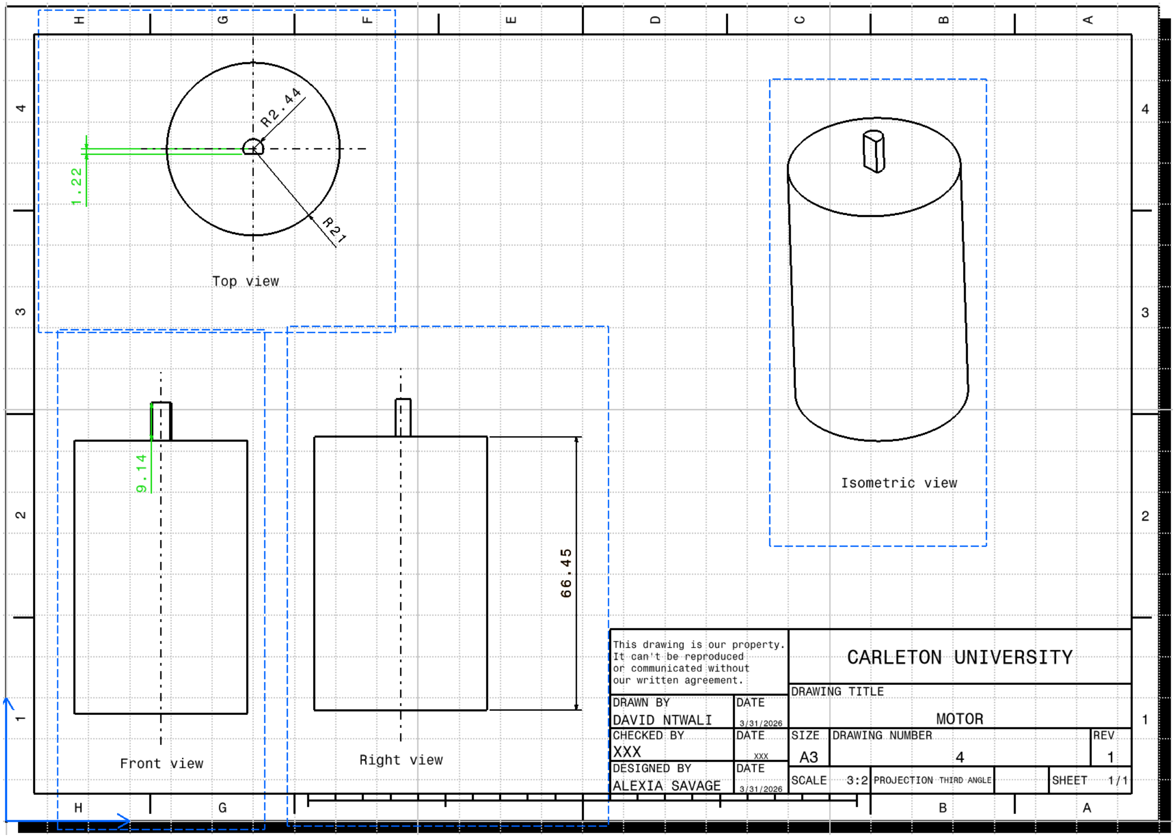

Figure 5:Motor

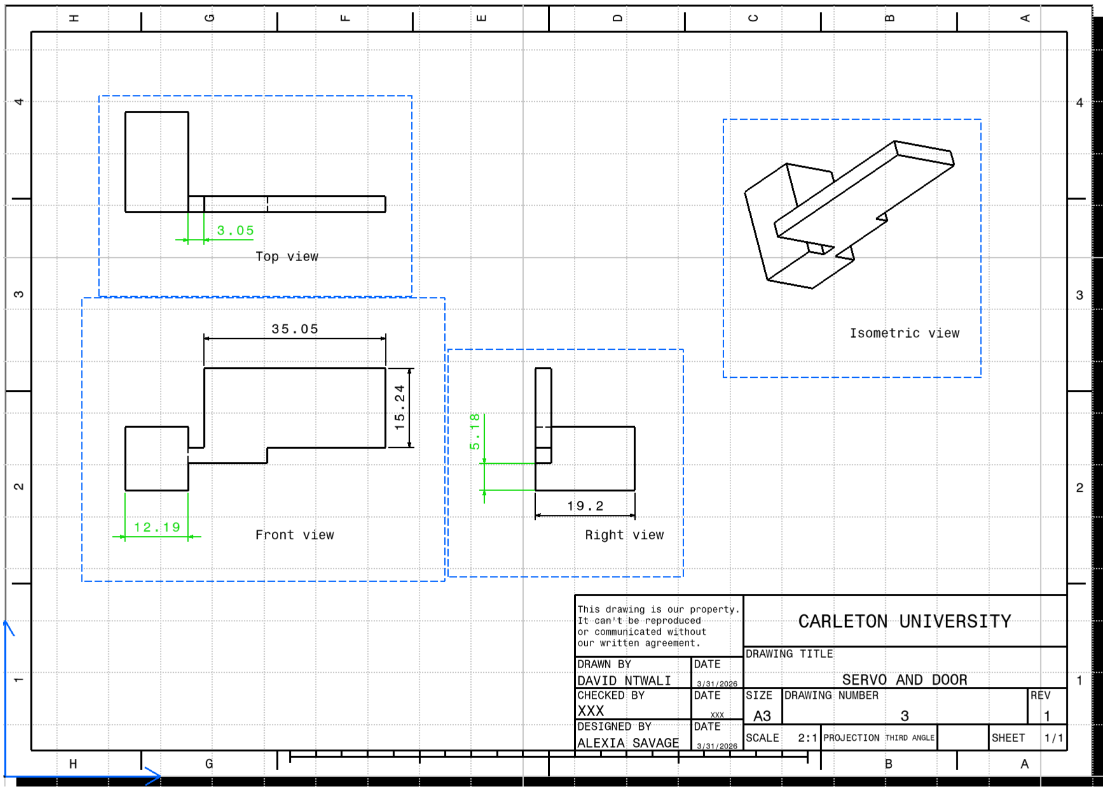

Figure 6: Servo Motor & Door

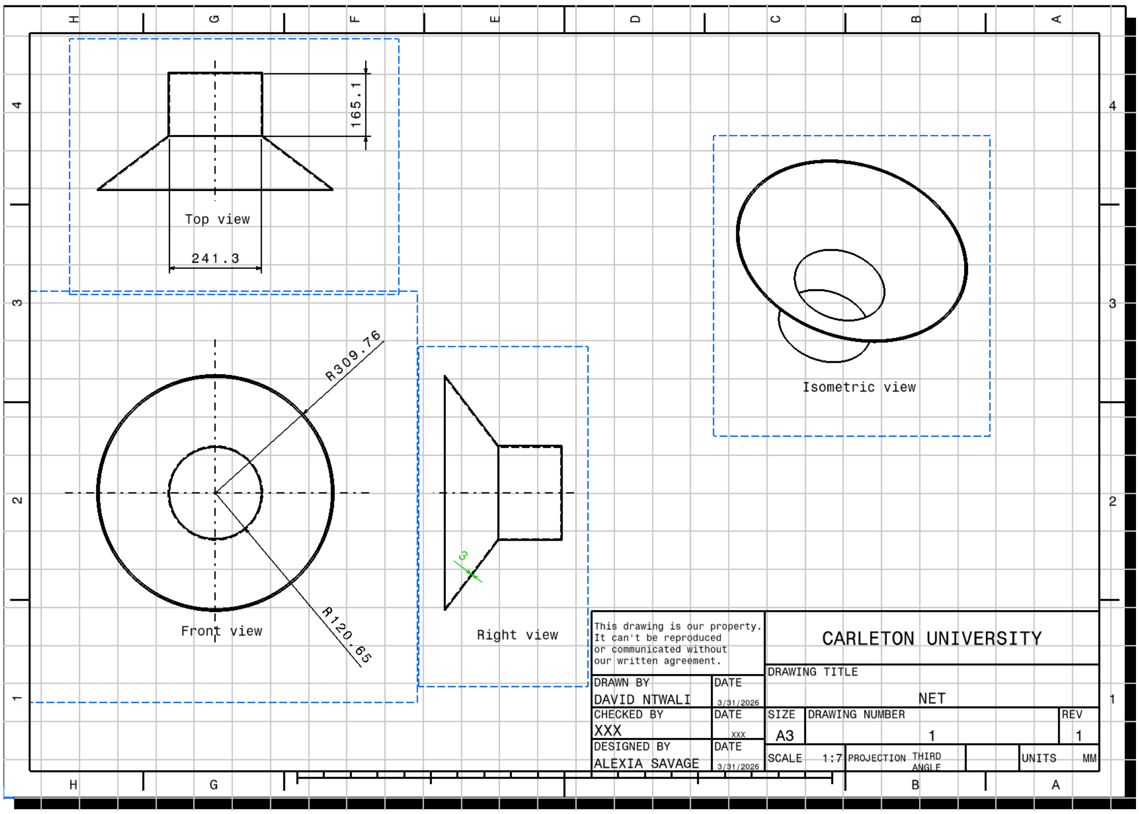

Figure 7: Net

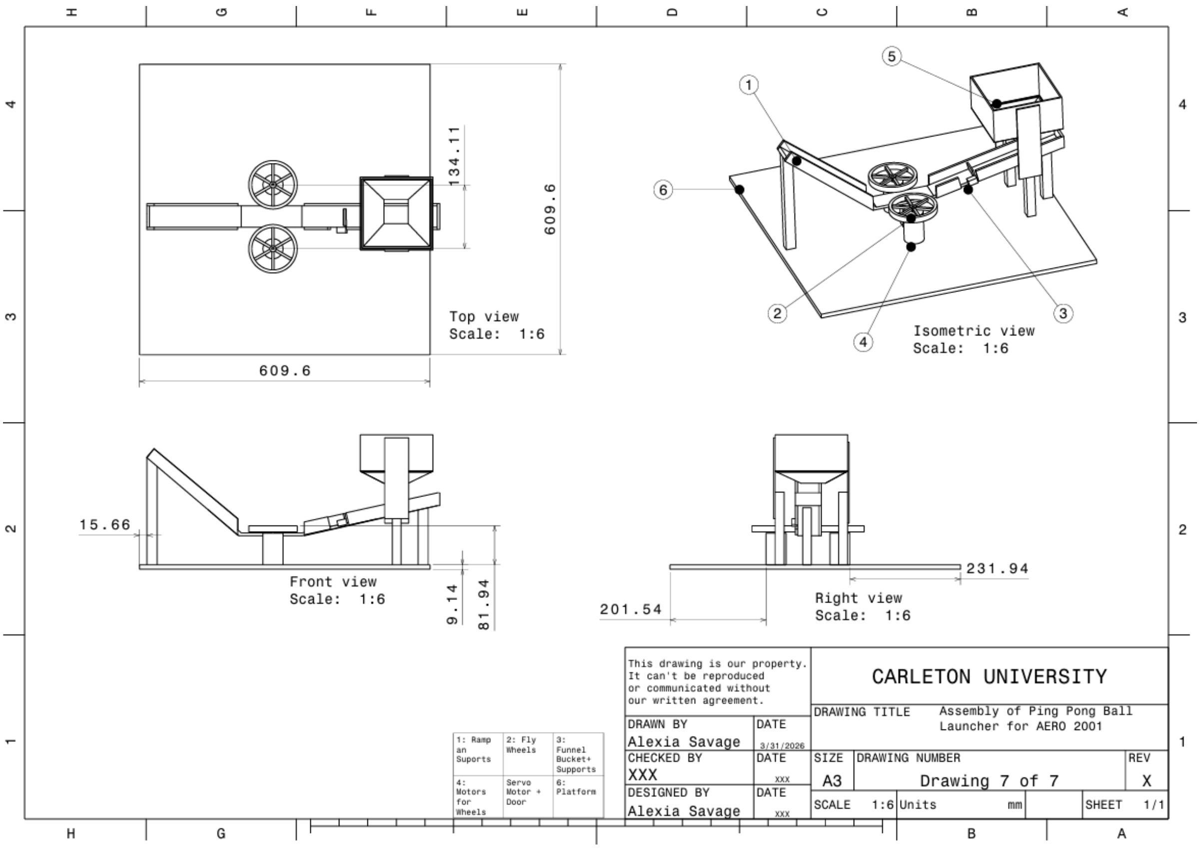

Figure 8: Assembly of Ping Pong Launcher