Design Project 2 – Get A Grip

ENGINEER 1P13 – Integrated Cornerstone Design Projects in Engineering

Tutorial 7

Team Thurs-16

ISABEL ANDERSON (anderi3)

ELAINE OCAMPO (ocampm6)

REEM BASIOUNY (basiounr)

AKINNIYI CHIDIEBUBE (akinniyc)

LUAY ALABED ALKADER (alabedal)

Submitted: December 6th, 2023

Course Instructors: Dr. McDonald, Dr. Doyle, Dr. Ebrahimi, Dr. Fleisig, Dr. Hassan, Dr. Zurob

Table of Contents

Academic Integrity Statement

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Isabel Anderson 400509424


____

(Student Signature) *

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Reem Basiouny 400509539


____

(Student Signature) *

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Elaine Ocampo 400509437


____

(Student Signature) *

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

LUAY ALABED ALKADER 400526433


_________ ________________

(Student Signature) *

The student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University.

Akinniyi Chidiebube 400501611

_____________ _________
(Student Signature) *

Executive Summary

The task for this project, labeled "Get a Grip", takes inspiration from the increasing need for robotic arm assistance during surgical procedures [1]. As a part of this process, our team was approached with the challenge of designing a container that can be used to hold a surgical tool and partake in the tool's sterilization, as well as designing a program code that will pick up, transfer, and drop off different containers to their assigned autoclave location [1]. The autoclaves need to be activated to start the sterilization process, and then deactivated when it is done. Our solution to this challenge includes a container design that can fit in our assigned footprint shape, and that securely holds our assigned surgical tool for sterilization. Our container design includes small holes in the bottom and sides to allow water and steam to pass through the container and reach the tool inside. We also included a lid with holes to keep the surgical tool from exiting the container. Inside the base of the container, there are two small pegs that are designed to keep the tool from moving within the container during sterilization. The computing component of our solution includes a code used to control the Q-Arm. First, our program spawns a new container every loop for six different containers in a random order. The code then goes on to include functions that can pick up the container, allows the use of the right potentiometer to rotate the arm towards the corresponding-coloured autoclave, identify the color and size of the container, and then drop off the container to the correct location based on the previous information. The program also uses the left potentiometer to indicate whether an autoclave drawer needs to be opened or not depending on its size. We also include print statements within each function to instruct any user on how to operate the Q-Arm with the potentiometers. After we identified all of the functions, they are included in a specific order within our main function so the program can run simply by calling the main function. The program will then terminate after six runs of the code, indicating that all six containers have been dropped off at their corresponding locations. These two components form a possible solution for the use of Q-Arms in the healthcare field. Our solution presents a suitable container that will aid in the sterilization process, as well as a program that can be used to minimize time when organizing and sterilizing specific tools.

Main Body

Summary of Design Objectives

Project "Get A Grip" is geared towards achieving two main objectives aimed at optimizing project efficiency and the main objective which is transporting the container safely. To complete these tasks, we consider a variety of objectives, constraints, and functions throughout our design process *(Table 1)*. Firstly, the initial main objective is entrusted with the modeling team, tasked with developing a 3D-modeled container that is both lightweight and medical-grade. The container is designed to picked up for sterilization processes and seamlessly transport it to an autoclave using a robotic Q-Arm. This involves implementing an efficient sterilization procedure, crafting a secure form capable of precisely holding surgical tools, and ensuring adherence to specified solid model criteria. as well as autoclave footprint restrictions and design limits. The limitations on the weight to be less than or equal to 350g, and limitations placed by the 3D printer requiring that features of the container must be greater than 2mm in size [1].

Subsequently, the second main objective involves the computing team in creating a simple computer code for the Q-arm. This program is essential for executing multiple commands involved in transferring of the container. The application plays a critical role in ensuring the precise execution of orders, facilitating the smooth transfer of the container to the autoclave for sterilization without any disruptions, and maintaining strict adherence to safety requirements throughout the process [1]. Specifically, the coded needed to be able to loop through six differently coloured and sized containers placing them in the correct autoclave with the proper use of left and right potentiometers [1].

Objectives	Constraints	Functions
Container should securely hold surgical tools.	Container must have weight less than or equal to 350g	Container has the ability to be held.
Program code should be time efficient and effective.	The container must sit in the given footprint.	Arm picks up and moves container to the correct autoclave.
The grip of the arm should be well balanced and strong enough to hold the container weight.	Program must interact with Q-labs and use a potentiometer.	The code and the containers must be compatible with each other.
Minimal excess movement of the arm.	Tools must fit properly in the container.	The container can be closed once the tool is inside.
Objects inside the container can be sterilized within it.	Must have a total print time after scaling of less than two hours, and no part can be smaller than 2mm.	Tools have minimal movement inside container.

Table 1. Finalized list of objectives, constraints, and functions

Background and Research Summary

In project two "Get a Grip", the goal was to design a container that would hold and protect a medical tool for sterilization in an autoclave [1]. The other aspect of the project was creating code for a robotic arm (Q-arm) to sort and move containers. It could move based on joints modeled similarly to those of a human arm, with a base, shoulder, elbow, and gripper. The motion of the arm needed to be done step by step in order to not overlap commands and have the motion be smooth and the grip in the physical demonstration not to be too hard so as not to damage the tool. These could be moved individually using angle measurements, or the gripper effector of the arm could be sent to a specific coordinate location. The Q-arm was coded to pick up, move and drop off containers in specific autoclaves. In the simulation environment, there were three different coloured autoclaves green, red, and blue, of which each had a small and a large drop off location for the container. The unique drop off location was dependent on the colour and size of the container. The code took potentiometer readings as inputs for the rotation degrees and worked as switches to help determine the size of the container. Additionally, the code needed to check if the manual input matched the information it was sensing to see if an adjustment needed to be made. The container that was designed differed from the simulation containers, its shape needed to minimize mass, protect, and effectively hold the tool which was a pair of medical scissors [1]. At the same time, holes were needed to allow heat and water to pass through for sterilization in an autoclave. In real applications, autoclaves are machines used to sterilize medical tools, they often use wet heat (steam) to cleanse the object of germs and bacteria; because of this, the container would need to be made from a stiff material with a high melting point, a common choice is stainless steel [2]. When designing a container with a specific purpose such as for "Get a Grip," it is important to consider all options before choosing a design, as more than one idea will work, but some will be more optimal than others. The use of a robotic arm in real application would be similar to that of the simulation, it could help streamline process of sorting that would otherwise need to be done manually. In the modern world, variations of robotic arms are used from everything from space (the Canadarm) to medicine, and more [3].

Description of Proposed Solution

The functionality of the design calls a main function which includes a for loop that calls three predefined functions: pickup, rotate base, and drop off (*Figure 1*). Before the main function calls any of those functions, it starts off by shuffling a list of container IDs from 1-6. This randomizes the order in which the containers will spawn. While there is still a container to spawn, the main function calls the pickup function where the Q-arm will go to the container pick up location, pick up the container and return it to the home position. After the Q-arm has gone through the pickup function, the main function calls the rotate base function where the user can use the right potentiometer to rotate the base if and only if the left potentiometer is set to 0. The user will then set the left potentiometer to 50 to indicate the desired rotation is acquired. The main function then calls the drop off function where the program will check if the autoclave colour matches the container colour and if it is a small or large container. If it is a small container, the Q-arm will drop the container off on top of the corresponding autoclave, and if it is large, the program will activate the autoclaves, open the corresponding autoclave's drawer where the Q-arm can now drop the container off inside, which now allows the program to close the drawer and deactivate the autoclaves. The main function loops through this whole process for all 6 containers, then returns home and terminates after it is all done.

```
def main():

    """
    Function: main()

    Purpose: To pickup, move and drop off containers generated at random into their correct autoclaves.

    Inputs: container - a random number from 1-6

            potentiometer readings - moved by hand and shown as percentages (floats form 0-1)

    Outputs: arm movement

    Authors: Isabel Anderson, Reem Basiouny, Elaine Ocampo
    Last Update: November 16, 2023
    """
#generate a number from 1-6 for the container ID then spawn that container.
    container_ids = [1,2,3,4,5,6]

    random.shuffle(container_ids)

    print(container_ids)

    for i in range(6):
        container = container_ids[i]
        time.sleep(3)
        arm.spawn_cage(container)
        print("The container has id number", container)
        time.sleep(2)
#Pick up the container
        pick_up()
        time.sleep(0.5)
        print("Set left potentiometer to 0")
        while potentiometer.left() != 0.0:
            time.sleep(2)
        print("You may now use the right potentimeter to rotate the arm, when you are done, the left potentiometer to 50%")
#rotate the base to the correponding autoclave
        rotate_base()
        while (potentiometer.left() != 0.5):

            time.sleep(2)
        print("if the container is large set left potentiometer to 100, if it is small set between 50 and 100")
#drop off the container
        drop_off(container)
        time.sleep(2)
        print("Thank you container has been placed, please reset both potentiometers to 50%")
```

Figure 1. Main Function in the code

The proposed container is in the shape of a cuboid, truncated on two opposite sides to appear more triangular and similar to the tool; it was designed this way because it was intended to house a surgical clamp, which commonly has a wide base and a small tip (*Figure 2*). The container is just as high as the tool. Although the shape is tapered, it maintains two parallel faces at all opposite ends to give the arm places to hold onto. It has two extruded cylinders in the interior for equipment security. It has multiple openings (holes) along its sides to facilitate the process of sterilization and uses a hinge system to open and close the container.

The container constraints were all met as it weighed less than 350 grams, every feature was equal to or greater than 2mm, it secured the equipment, and it could fit into its footprint.

Figure 2. Final 3D printed container design

Strengths and Limitations of Design

The simulation code runs smoothly without errors and loops through all six containers, placing them correctly. In the drop off function, if the potentiometer input does not match what the code has stored for correct size and colour, it will ask for new potentiometer readings, even allowing you to re-rotate be base if needed, until the corresponding values are given (*Figures 3 and 4)*. This helps to eliminate errors in placement which would be incredibly important for real world application. The arm is coded to sense if it is within 15cm of the correct autoclave location and automatically correct itself, this eliminates human error in rotation, and ensures a safe and repeatable deposit of the container. However, the arm moves slowly, this is needed as it otherwise combines movements, leading to the wrong output. This is not a large issue in the simulation, but when applied on a large scale a couple seconds quickly escalate into hours. The potentiometer can also be finicky and occasionally cause abrupt movement or not respond immediately. Additionally, when transferring from the simulation to physical environment, numbers and coordinates that worked for one don't always apply to the other, this could again be an issue when applied on a wider scale with many different arms and locations.

```python
def drop_off(ContainerID):
    """
    Function: drop_off()

    Pupose: This function angles the arm and drops it in the correct autoclave, as well as opening and closing drawers if needed.
    Inputs: ContainerID - an integer from 1-6
            Potentiometer readings - input by hand and come in the form of a float from 0-1.
    Outputs: The movement of the arm
    Author: Isabel Anderson
    Last Update: December 1, 2023
    """
    item_can_drop = False
    arm.activate_autoclaves()
#See if the container is large or small and reach the drop position for the autoclave of the correct colour and size with the drawer being opened if need be.
    while not item_can_drop:
        if (potentiometer.left() >0.5 and potentiometer.left() <1) and (ContainerID < 4): #Checks if the container is small
            if (ContainerID == 2 and arm.check_autoclave('green') == True): # small green container position
                arm.move_arm(-0.587, 0.226, 0.39)
                item_can_drop = True
            elif (ContainerID == 1 and arm.check_autoclave('red')==True): #small red container position
                arm.move_arm(0.0, 0.612, 0.302)
                item_can_drop = True
            elif (ContainerID == 3 and arm.check_autoclave('blue') == True): #small blue container position
                arm.move_arm(0.0, -0.635, 0.34)
                item_can_drop = True
            else:
                print("error")
                time.sleep(2)
                print("Please set left potentiometer to 0 then rotate the base again to the correct autoclave, when you are done please set left potentiometer to 50%")
                rotate_base()
                item_can_drop = False

        elif potentiometer.left() == 1 and ContainerID > 3: #Checks if the container is large so it can also open the corresponding drawer

            if (ContainerID == 4 and arm.check_autoclave('red')==True): #large red container position
                arm.open_autoclave('red', True)
                arm.move_arm(0.0, 0.40, 0.21)
                item_can_drop = True
            elif (ContainerID == 5 and arm.check_autoclave('green') == True): #large green container position
                arm.open_autoclave('green', True)
                arm.move_arm(-0.408, 0.159, 0.196)
                item_can_drop = True
```

Figure 3. Drop Off function

```python
            elif (ContainerID == 6 and arm.check_autoclave('blue') == True): #large blue container position
                arm.open_autoclave('blue', True)
                arm.move_arm(0.0, -0.397, 0.178)
                item_can_drop = True
            else:
                print("error")
                time.sleep(2)
                print("Please set left potentiometer to 0 then rotate the base again to the correct autoclave, when you are done please set left potentiometer to 50%")
                rotate_base()
                item_can_drop = False
#Drop off the container and return arm to home
    time.sleep(2)
    arm.control_gripper(-45)
    time.sleep(2)
    print("The object has been dropped off, returning to home")
    arm.home()
#Check to see if a drawer needs to be closed and close the corresponding one

    if ContainerID >3:
        if ContainerID == 4:
            arm.open_autoclave('red', False)
        elif ContainerID == 5:
            arm.open_autoclave('green', False)
        elif ContainerID == 6:
            arm.open_autoclave('blue', False)
        else:
            print("error")
    arm.deactivate_autoclaves()
```

Figure 4. Drop Off function continued

On the modelling side, the container, should be able to securely hold the specific tool in place during the relocation process. Openings are present to enable the container to act efficiently for sterilization, and due to its shape and the cylinders within that aid in tool security, the tool's movements are minimised. The shape of the container has also proven to be compatible with the Q-Arm. The shape also significantly reduces the amount of material needed to produce a single unit, minimising the cost of production and printing time. The hinge system also allows the container to be closed after the tool has been placed inside, making it easier to transport the container with the tool inside.

The container is only able to transport one tool at a time to ensure that the shape fits within the footprint and minimizes material-related costs. Additionally, because of the container's width and low centre of mass, the equipment security could be compromised if it were to encounter a slight turning force while being transported. The shape of the container makes it rather difficult to pick up, although it's still very possible. The container also has no direct system for sterilizing the bottom of the tool unless the surface the container is placed on also has openings. The thickness of components is just barely acceptable – it was designed this way to maximize the interior volume while minimizing exterior area to ensure that the container still fits within the footprint – making the container more susceptible to failure when under the influence of lateral, unaccounted gravity loads or unprecedented force; if a miscalculation occurs in the Q-Arm code, the container will likely be crushed.

Summary of Contributions

Table 2. Summary of Contributions Table

Isabel Anderson	– Created the final Gantt chart – Created the drop off function – Contributed to the main function. – Submitted the coding team deliverables and milestones. – Wrote the background research summary and the coding section of strengths and limitations.
Elaine Ocampo	– Created pick up function in the code – Contributed to main function in the code – Filled out the meeting minutes for weekly TA meetings – Kept a logbook of additional meeting – Wrote the computing part of description of proposed solution
Reem Basiouny	– Created preliminary Gant Chart – Made manager agendas for weekly TA meetings – Created rotate base function in code – Contributed to main function in code – Wrote executive summary
Akinniyi Chidiebube	– Submission of Modelling Sub-team's deliverables – Wrote the modelling section of “Strengths and Limitations” – Made the container designs – CAD Modelling of the container – Contributed to Printing the 3D container
Luay Alkader	– Created The objective summary. – Contributed to designing the container. – Grammer check of all the paragraphs and topics. – Contributed to Printing the 3D container

Reference List

[1] McMaster University Faculty of Engineering, “P2 Project Module: Get a Grip,” Avenue to Learn. [Accessed: December. 6th, 2023].

[2] "EFFECTIVE STAINLESS-STEEL USAGE IN MEDICAL ENVIRONMENTS," Unified Alloys, [Online]. Available: https://www.unifiedalloys.com/blog/medical-stainless-steel. [Accessed December. 3rd 2023].

[3] "BEST APPLICATIONS OF ROBOTIC ARMS," UNIVERSAL ROBOTS, 15 November 2022. [Online]. Available: https://www.universal-robots.com/in/blog/applications-of-robotic-arms/. [Accessed: December. 4th, 2023]

Appendices

Appendix A: Project Schedule

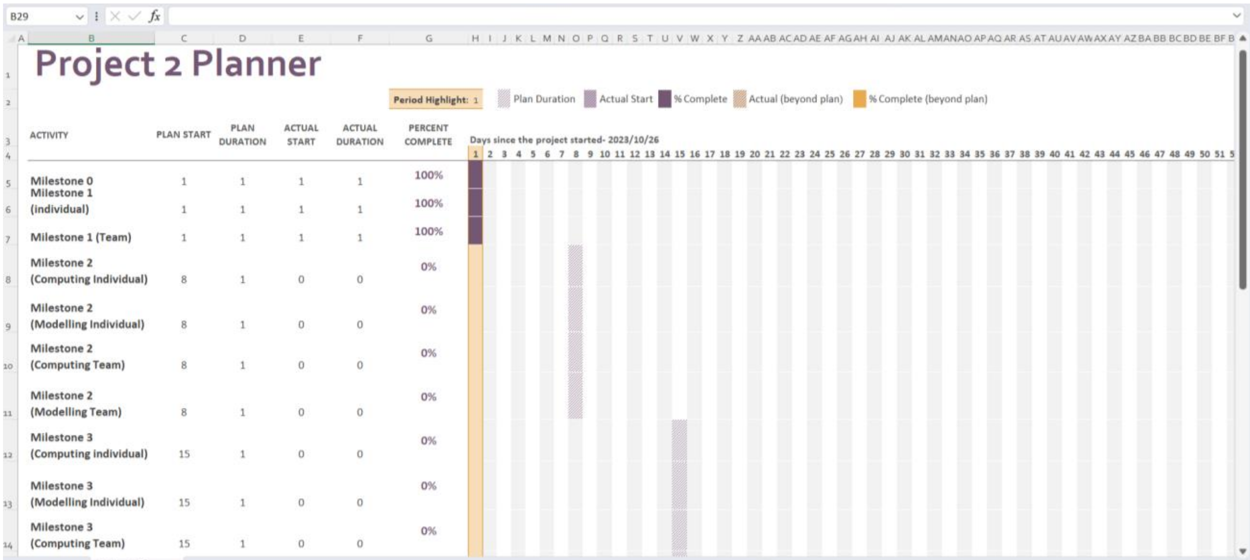


Figure 2. Preliminary Gantt Chart

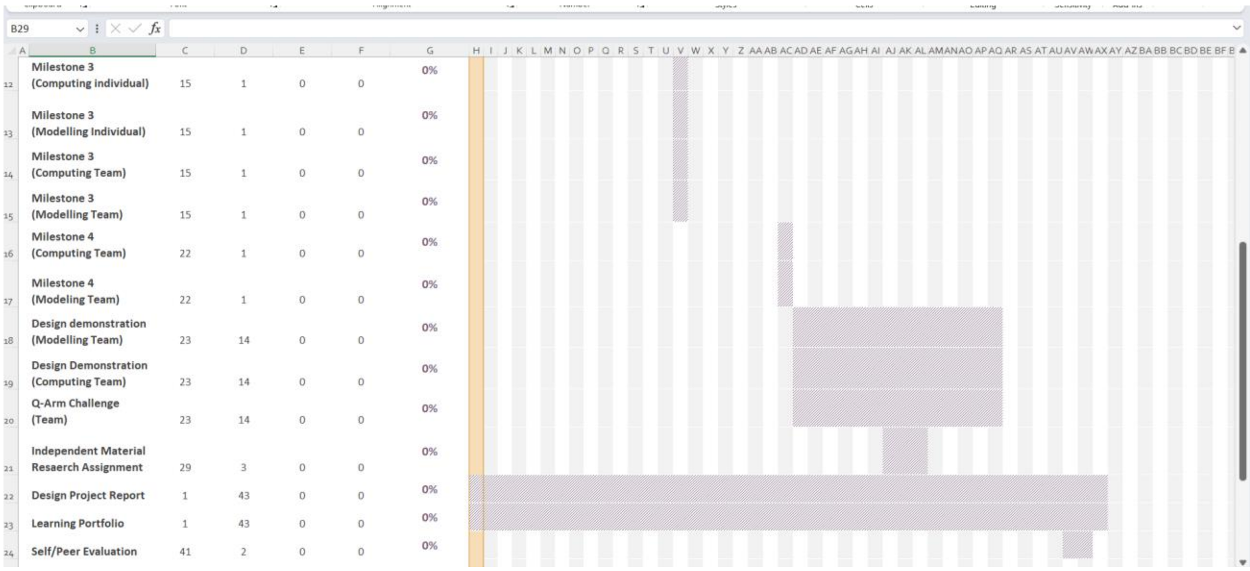


Figure 3 Preliminary Gantt Chart continued

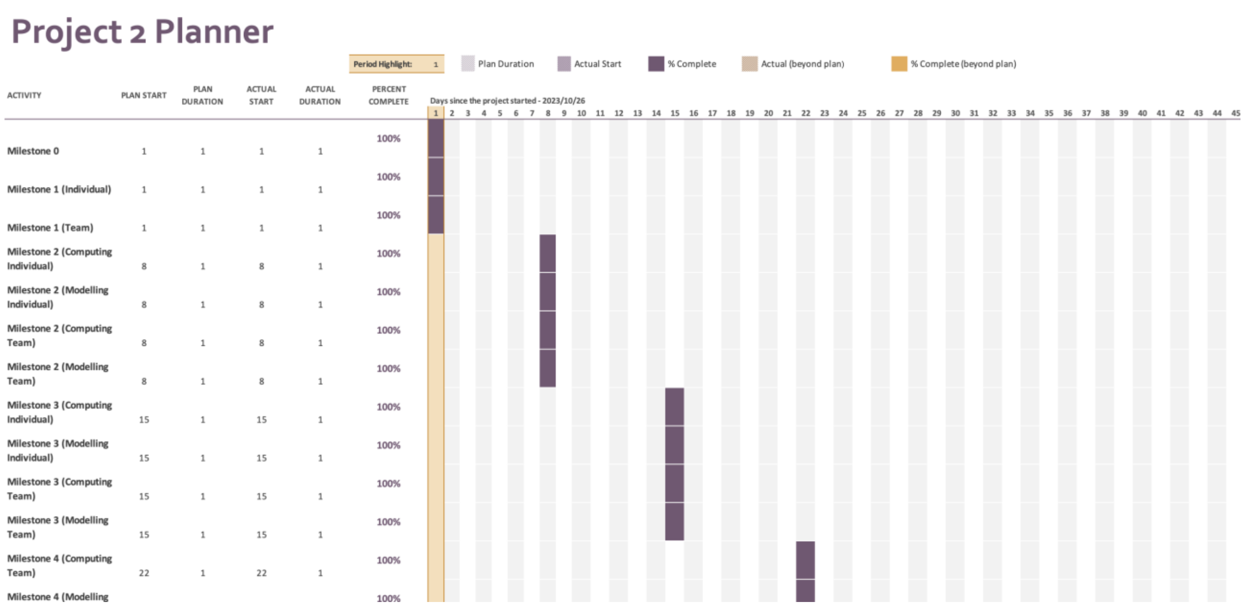


Figure 4. Final Gantt Chart

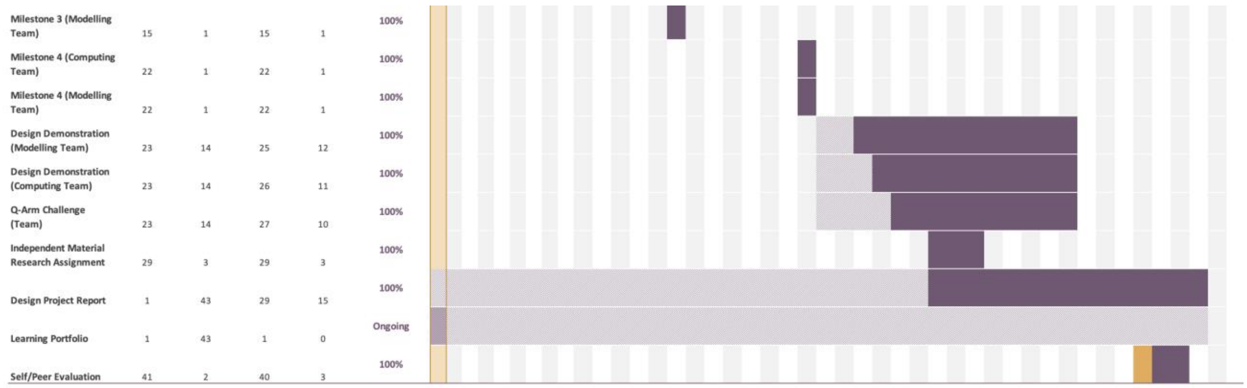


Figure 5. Final Gantt Chart continued

Coordinator Logs

Meeting Date	Reason
Tuesday, November 7 (Meeting outside DS)	Write pseudocode for our program
Tuesday, November 14 (Meeting outside DS)	Write code and define functions for our program

Tuesday, November 7 (Meeting outside DS):

- Wrote pseudocode for picking up the container and continuing or terminating the program depending on whether there are still more containers to sort and sterilize.

Tuesday, November 14 (Meeting outside DS):

- Turned our pseudocode into actual code and wrote the rest of our program by defining a pickup function, a rotate base function, and a drop off function.
- Split up the work between everyone and decided on who worked on which function.

Figure 6. Logbook of Additional Meetings and Discussions

Appendix B: Scheduled Weekly Meetings

Agenda Items

1. Attendance and recap
   - Last time we broke down the project into objectives functions and means and used those to layout our individual milestones
   - Prior to DS we had to do a workflow and prototype model, we all did it, did anyone have any questions about their specific one
2. We didn't really have a prototype yet from last time to reflect on and change since it was our first DS, but we can discuss the conceptualizing of both sub teams, and how they are built off of, and/or differ from the functions and means we highlighted in our last milestone
3. Action items for this meeting, (we also need to see what tool and footprint we are assigned)
4. Final notes: we seem to be on track of everything, are there any questions? – Check if our models and flow charts look right.

Meeting Minutes

1. Attendance and Recap
   a. Shared our workflow and prototype model with everyone and asked questions about each other's
2. Discussion
   a. Discussed what each sub team has done so far and what needs to get done in the future.
3. Action Items for this meeting
   a. Checked on Avenue our assigned environment, tool, and footprint
4. TA Check-in
   a. Asked a TA if our workflows and models were good and received positive feedback with some minor changes to make.

Figure 7. November 2 Agenda Items and Meeting Minutes

Agenda Items

1. Attendance
2. Check up- we all completed Stage 1 of Milestone 3 (specific pseudocode and initial design of final sterilization container)
3. Ask T.A to look over our functions from milestone 2 stage 2 (how the potentiometers work with them), the pseudocode we just did (stage 1 milestone 3), and modeling teams design so far.
4. Sub-team check up on each other's progress
5. Check Action Items

Meeting Minutes

1. Attendance
   a. Everyone is here
2. Check-up
   a. The initial design of the container looks good
   b. The pseudo code had to be edited a little
3. TA Check-in
   a. Functions looked good, just ensure the functions include the potentiometers
4. Sub-team check-in
   a. Modelling sub-team updated the computing sub-team on how the container is looking, and the computing sub-team showed the modelling sub-team what functions were written
5. Action Items
   a. Make sure Milestone 3 is completed and submitted by the end of the day

Figure 8. November 9 Agenda Items and Meeting Minutes

Agenda Items

1. Attendance
2. Check up- we all completed Stage 1 of Milestone 4
3. Ask T.A to look over our functions and ask about main function workings, modeling teams design.
4. Sub-team check up on each other's progress
5. Check Action Items

Meeting Minutes

1. Attendance
    a. Everyone is here
2. Check up
    a. Everyone completed Stage 1 of Milestone 4 which was completed prior to DS
3. TA Check up
    a. A TA took a look at the modelling sub-team's design and asked about the mass and why certain decisions were made.
    b. Was not approved for 3D printing, made suggestions like change the shape and height of the container to minimize the material and printing time.
    c. A TA looked over the simulation code and if it could run with no errors.
4. Sub-team check up
    a. Updated each other on the TA's feedback
5. Action Items
    a. Submit Milestone 4 by the end of the day

Figure 9. November 16 Agenda Items and Meeting Minutes

Agenda Items

1. Attendance
2. Check up- made improvements to both the code and the final cad model.
3. Ask TA for feedback on new code and do follow up meeting on container design
4. Action Items

Meeting Minutes

1. Attendance
    a. Everyone is here
2. Check up
    a. Got the go ahead to 3D print as modifications were made to the final CAD model
    b. Added to our code so it is able to loop through all 6 containers in a random order
3. TA Check up
    a. A TA watched as we ran our program and she saw that we were able to loop our code and sort a large container appropriately
4. Action Items
    a. Finalize our code and make any adjustments if needed
    b. Finish 3D printing

Figure 10. November 23 Agenda Items and Meeting Minutes

Appendix C: Comprehensive List of Sources

[1] McMaster University Faculty of Engineering, “P2 Project Module: Get a Grip,” Avenue to Learn. [Accessed: December. 6th, 2023]

[2] "EFFECTIVE STAINLESS-STEEL USAGE IN MEDICAL ENVIRONMENTS," Unified Alloys, [Online]. Available: https://www.unifiedalloys.com/blog/medical-stainless-steel. [Accessed: December. 3rd, 2023].

[3] "BEST APPLICATIONS OF ROBOTIC ARMS," UNIVERSAL ROBOTS, 15 November 2022. [Online]. Available: https://www.universal-robots.com/in/blog/applications-of-robotic-arms/. [Accessed: December. 4th, 2023]

[4] Ansys GRANTA EduPack software, ANSYS, Inc., Cambridge, UK, 2023 (www.ansys.com/materials)

[5] "Autoclave Manufacturer: Avoiding Metal Contamination," Beta Star Life Science Equipment, [Online]. Available: https://www.betastar.com/autoclave-manufacturer-avoiding-metal-contamination/#:~:text=This%20prevents%20any%20corrosion%20or,properties. [Accessed: December. 3rd, 2023].

[6] L. V. R. Thomas Duda, "3D Metal Printing Technology," ScienceDirect, 22 December 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896316325496?via%3Dihub. [Accessed: December. 3rd, 2023].

[7] "WHAT CAN AND CANNOT BE AUTOCLAVED," Beta Star Life Science Equipment, [Online]. Available: https://www.betastar.com/what-can-cannot-be-autoclaved/. [[Accessed: December. 3rd, 2023].

[8] "Carbon Steel vs Stainless Steel," Reliance Foundry, [Online]. Available: https://www.reliance-foundry.com/blog/carbon-steel-versus-stainless-steel. [Accessed: December. 3rd, 2023].

[9] "Advantages of stainless-steel tanks versus carbon steel tanks," Herpasa, [Online]. Available: https://herpasa.com/en/blog/2/23/advantages-of-stainless-steel-tanks-versus-carbon-steel-tanks. [Accessed: December. 3rd, 2023].

[10] "Titanium vs Steel, What’s the Difference," CNCLATHING, 17 September 2020. [Online]. Available: https://www.cnclathing.com/guide/titanium-vs-steel-whats-the-difference-titanium-and-steel-comparison-cnclathing. [Accessed: December. 3rd, 2023].

[11] "Stainless Steel or Aluminum? Material Considerations for Sterilization Cases & Trays," 8 2022 CRETEX. [Online]. Available: https://www.cretexmedical.com/blog/2022/06/08/stainless-steel-or-aluminum-material-considerations-for-sterilization-cases-trays/. [Accessed: December. 3rd, 2023].

[12] "Autodesk Inventor Professional 2023," Autodesk, 2023.

[13] Quanser, "Quanser Interactive Labs," 2023.

[14] "Stiffness vs Strength, How to Differentiate Them," CNCLATHING, 18 May 2021. [Online]. Available: https://www.cnclathing.com/guide/stiffness-vs-strength-how-to-differentiate-them-difference-between-strength-and-stiffness-cnclathing. [Accessed: November. 27th, 2023].

[15] T. A. &. H. VILLASENOR, "Engineering Fundamentals Refresh: Strength vs. Stiffness vs. Hardness," 2 February 2023. [Online]. Available: https://www.fictiv.com/articles/engineering-fundamentals-refresh-strength-vs-stiffness-vs-hardness. [Accessed: November. 27th, 2023].

[16] "Strength Versus Stiffness," 2018. [Online]. Available: https://courses.cit.cornell.edu/mclaskey/cee3710/stimulation3/strengthStiffness2018.html. [Accessed: November. 27th, 2023].

[17] "Stainless Steel or Aluminum," CRETEX, 8 June 2022. [Online]. Available: https://www.cretexmedical.com/blog/2022/06/08/stainless-steel-or-aluminum-material-considerations-for-sterilization-cases-trays/#:~:text=The%20primary%20metals%20used%20to%20manufacture%20sterilization%20cases,can%20offer%20different%20advantages%20during.

[18] "Functions I" class notes for ENGINEER 1P13, McMaster University, Fall, 2023

[19] "Functions II" class notes for ENGINEER 1P13, McMaster University, Fall, 2023

[20] "Sequences, Lists, Repetition" class notes for ENGINEER 1P13, McMaster University, Fall, 2023

[21] "Conditionals" class notes for ENGINEER 1P13, McMaster University, Fall, 2023

[22] "Assembly Modelling I," class notes for ENGINEER 1P13, McMaster University, Fall 2020

[23] Mcmaster University, "Python Library", Avenue to Learn, 2023.

Additional Documentation

Final Code

```
import random

#Note, for red set right potentiometer to around 77, for green anound 97, and for blue around 23

def pick_up():

    """
    Function: pick_up()

    Purpose: This function moves the arm to the spawn location, picks it up, and brings it back to the home position.
    Inputs: Coordinates of the spawn location and the home location

    Outputs: Moves the Q-arm back to the home position with the container, ready to be transferred

    Author: Elaine Ocampo

    Last Update: November 16, 2023

    """
    arm.move_arm(0.567,0.05,-0.006)
    time.sleep(2)
    arm.control_gripper(45)
    time.sleep(2)
    arm.move_arm(0.406, 0.0, 0.483)
    time.sleep(2)

def rotate_base():

    """
    Function: rotate_base
    Puprose: This function takes in left potentiometer values and allow the right potentiometer to function accordingly.
    Input: Left potentiometer values - 0.0 and 0.5
    Output: right potentiometer can rotate base
    Authors: Reem Basiouny
    Last Update: November 16, 2023

    """

    item_can_dropped = False
    old_reading = potentiometer.right()
    while not item_can_dropped:
#Can manually use the right potentiometer to rotate base
```

Figure 11. Final code

```
        if potentiometer.left() == 0.0:
            new_reading = potentiometer.right()
            delta = new_reading - old_reading
            x = 340*delta
            arm.rotate_base(x)
            time.sleep(2)
            old_reading = new_reading

        elif potentiometer.left() == 0.5:
#code can move to drop_off function
            item_can_dropped = True
            print("the item is ready to drop please set left potentiometer to 50%")

def drop_off(ContainerID):
    """
    Function: drop_off()

    Pupose: This function angles the arm and drops it in the correct autoclave, as well as opening and closing drawers if needed.
    Inputs: ContainerID - an integer from 1-6
            Potentiometer readings - input by hand and come in the form of a float from 0-1.
    Outputs: The movement of the arm
    Author: Isabel Anderson
    Last Update: December 1, 2023
    """
    item_can_drop = False
    arm.activate_autoclaves()
#See if the container is large or small and reach the drop position for the autoclave of the correct colour and size with the drawer being opened if need be.
    while not item_can_drop:
        if (potentiometer.left() >0.5 and potentiometer.left() <1) and (ContainerID < 4): #Checks if the container is small
            if (ContainerID == 2 and arm.check_autoclave('green') == True): # small green container position
                arm.move_arm(-0.587, 0.226, 0.39)
                item_can_drop = True
            elif (ContainerID == 1 and arm.check_autoclave('red')==True): #small red container position
                arm.move_arm(0.0, 0.612, 0.302)
                item_can_drop = True
            elif (ContainerID == 3 and arm.check_autoclave('blue') == True): #small blue container position
                arm.move_arm(0.0, -0.635, 0.34)
                item_can_drop = True
            else:
                print("error")
                time.sleep(2)
                print("Please set left potentiometer to 0 then rotate the base again to the correct autoclave, when you are done please set left potentiometer to 50%")
                rotate_base()
                item_can_drop = False
```

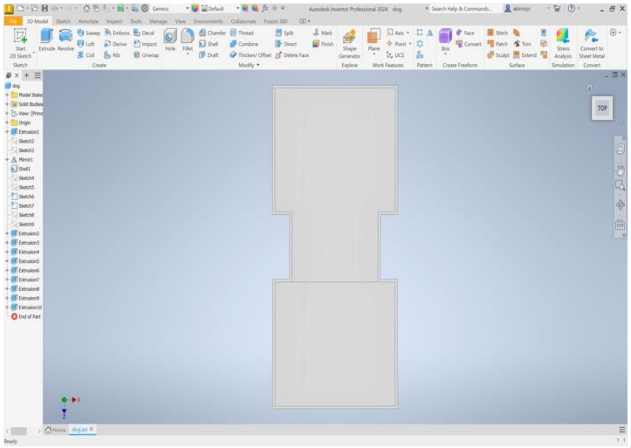
Figure 12. Final code continued

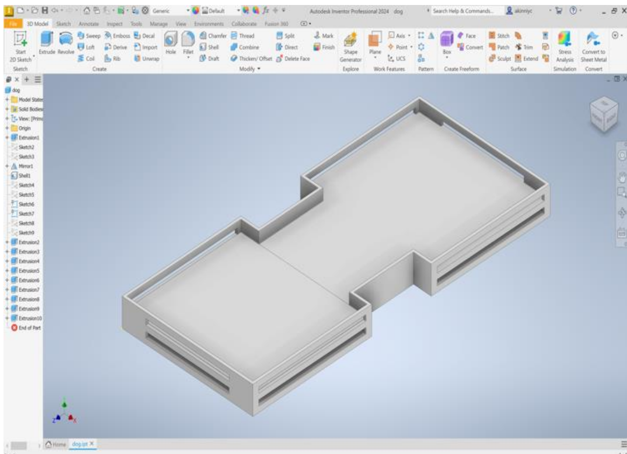
```python
        elif potentiometer.left() == 1 and ContainerID > 3: #Checks if the container is large so it can also open the corresponding drawer

            if (ContainerID == 4 and arm.check_autoclave('red')==True): #large red container position
                arm.open_autoclave('red', True)
                arm.move_arm(0.0, 0.40, 0.21)
                item_can_drop = True
            elif (ContainerID == 5 and arm.check_autoclave('green') == True): #large green container position
                arm.open_autoclave('green', True)
                arm.move_arm(-0.408, 0.159, 0.196)
                item_can_drop = True

            elif (ContainerID == 6 and arm.check_autoclave('blue') == True): #large blue container position
                arm.open_autoclave('blue', True)
                arm.move_arm(0.0, -0.397, 0.178)
                item_can_drop = True
            else:
                print("error")
                time.sleep(2)
                print("Please set left potentiometer to 0 then rotate the base again to the correct autoclave, when you are done please set left potentiometer to 50%")
                rotate_base()
                item_can_drop = False
#Drop off the container and return arm to home
    time.sleep(2)
    arm.control_gripper(-45)
    time.sleep(2)
    print("The object has been dropped off, returning to home")
    arm.home()
#Check to see if a drawer needs to be closed and close the corresponding one

    if ContainerID >3:
        if ContainerID == 4:
            arm.open_autoclave('red', False)
        elif ContainerID == 5:
            arm.open_autoclave('green', False)
        elif ContainerID == 6:
            arm.open_autoclave('blue', False)
        else:
            print("error")
    arm.deactivate_autoclaves()
```

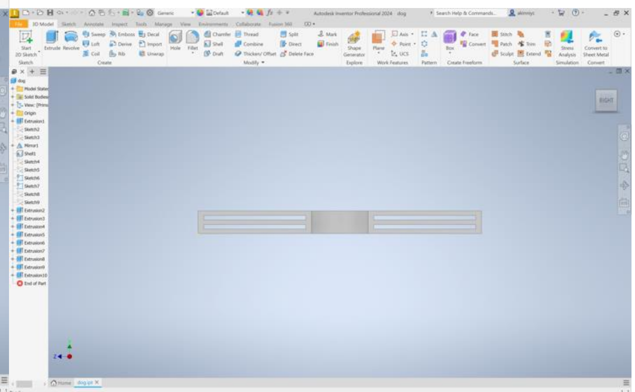
Figure 13. Final code continued once more

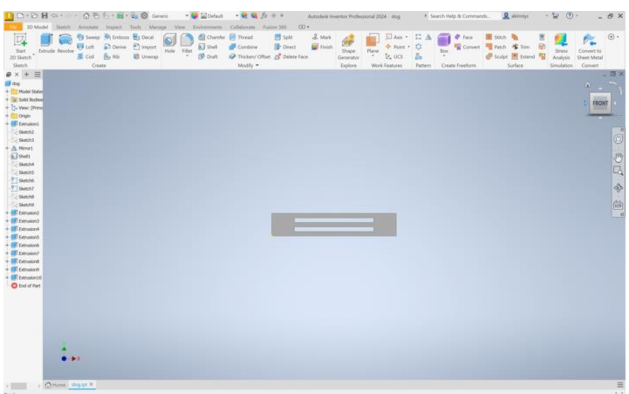
```
def main():

    """
    Function: main()

    Purpose: To pickup, move and drop off containers generated at random into their correct autoclaves.

    Inputs: container - a random number from 1-6

            potentiometer readings - moved by hand and shown as percentages (floats form 0-1)

    Outputs: arm movement

    Authors: Isabel Anderson, Reem Basiouny, Elaine Ocampo
    Last Update: November 16, 2023
    """
#generate a number from 1-6 for the container ID then spawn that container.
    container_ids = [1,2,3,4,5,6]

    random.shuffle(container_ids)

    print(container_ids)

    for i in range(6):
        container = container_ids[i]
        time.sleep(3)
        arm.spawn_cage(container)
        print("The container has id number", container)
        time.sleep(2)
#Pick up the container
        pick_up()
        time.sleep(0.5)
        print("Set left potentiometer to 0")
        while potentiometer.left() != 0.0:
            time.sleep(2)
        print("You may now use the right potentimeter to rotate the arm, when you are done, the left potentiometer to 50%")
#rotate the base to the correponding autoclave
        rotate_base()
        while (potentiometer.left() != 0.5):

            time.sleep(2)
        print("if the container is large set left potentiometer to 100, if it is small set between 50 and 100")
#drop off the container
        drop_off(container)
        time.sleep(2)
        print("Thank you container has been placed, please reset both potentiometers to 50%")
```

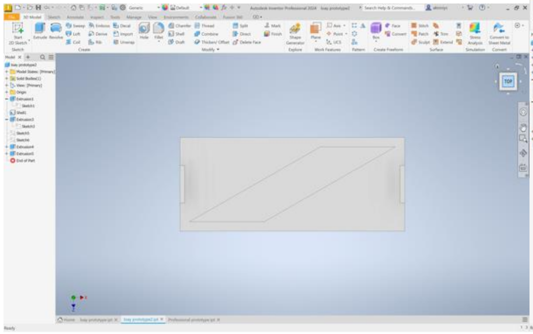
Initial Models

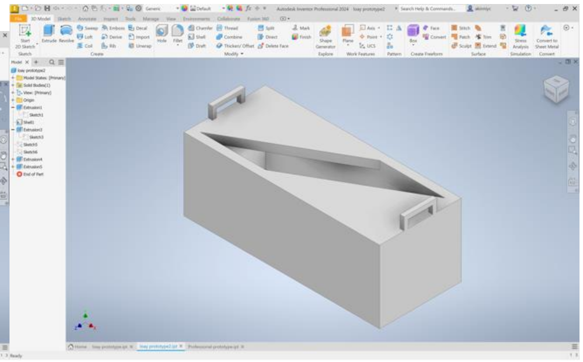


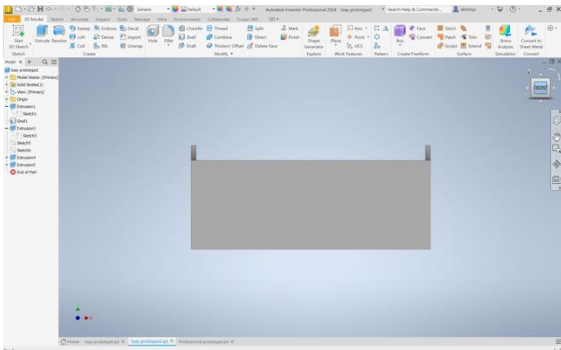


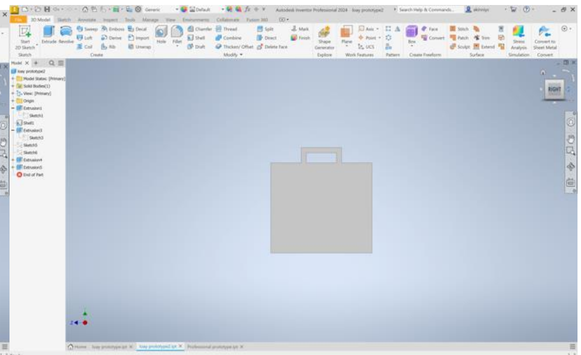




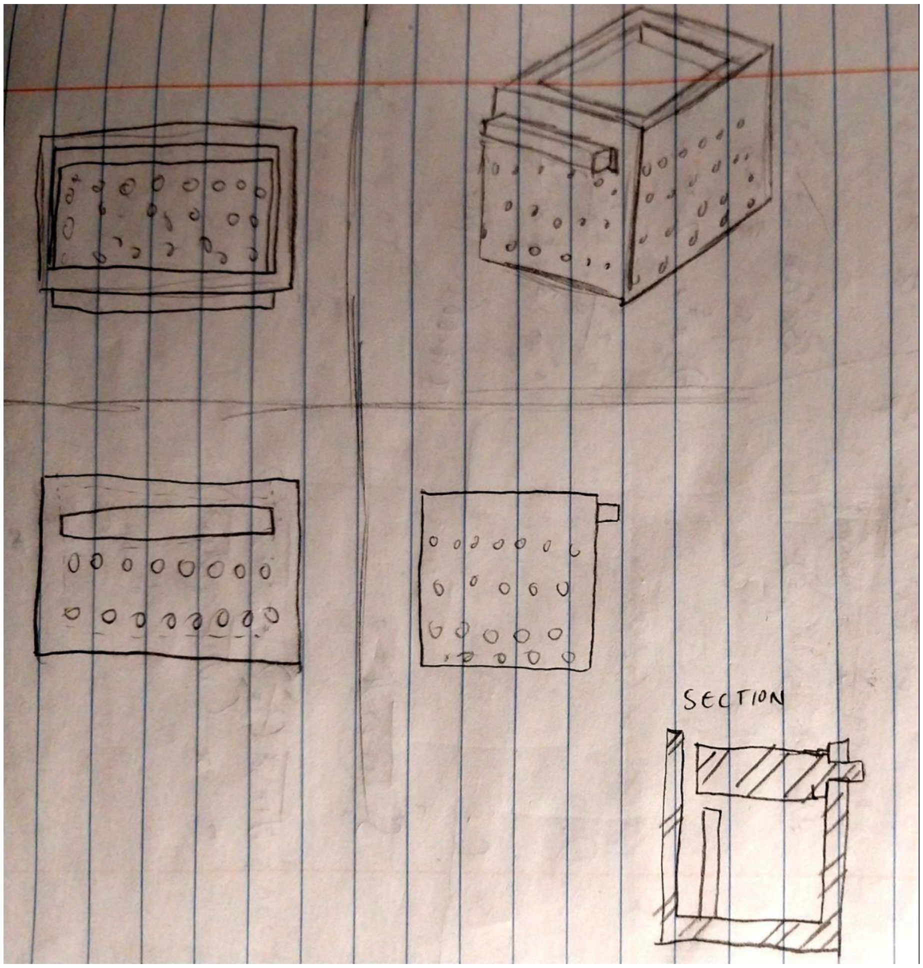


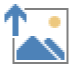




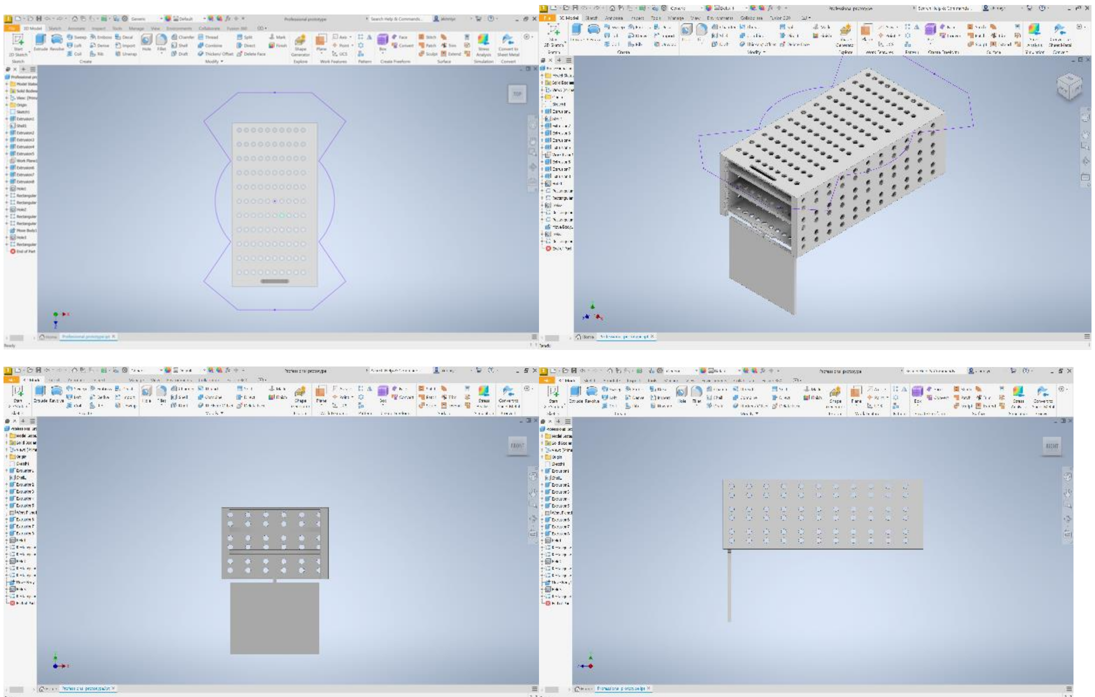


Preliminary Designs

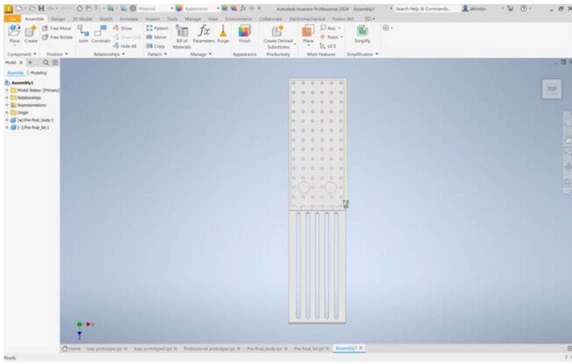
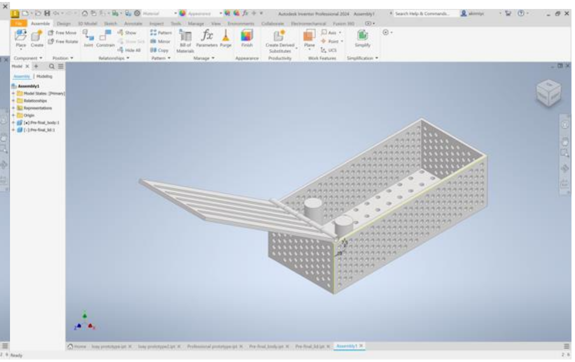
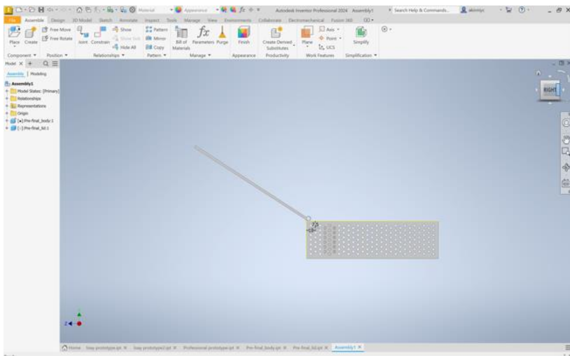
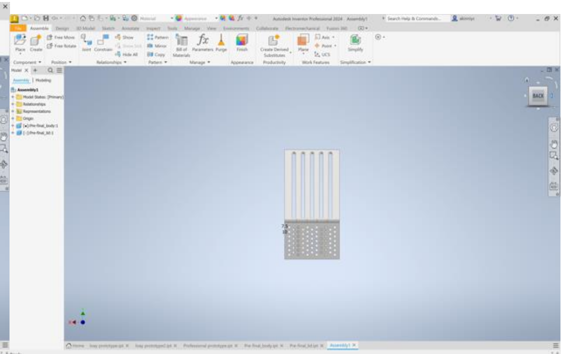




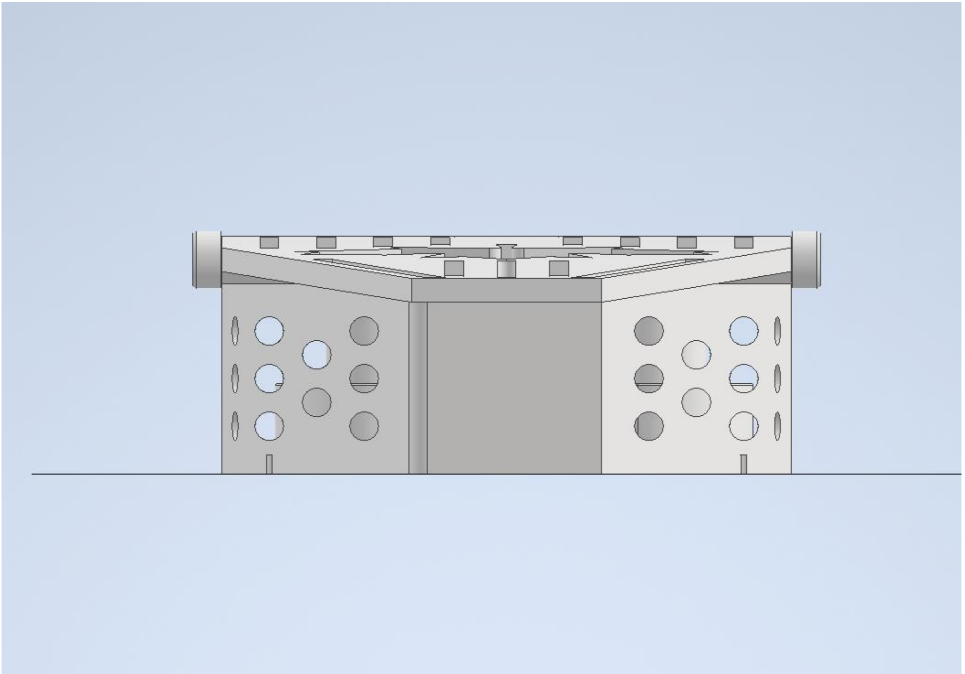
1.

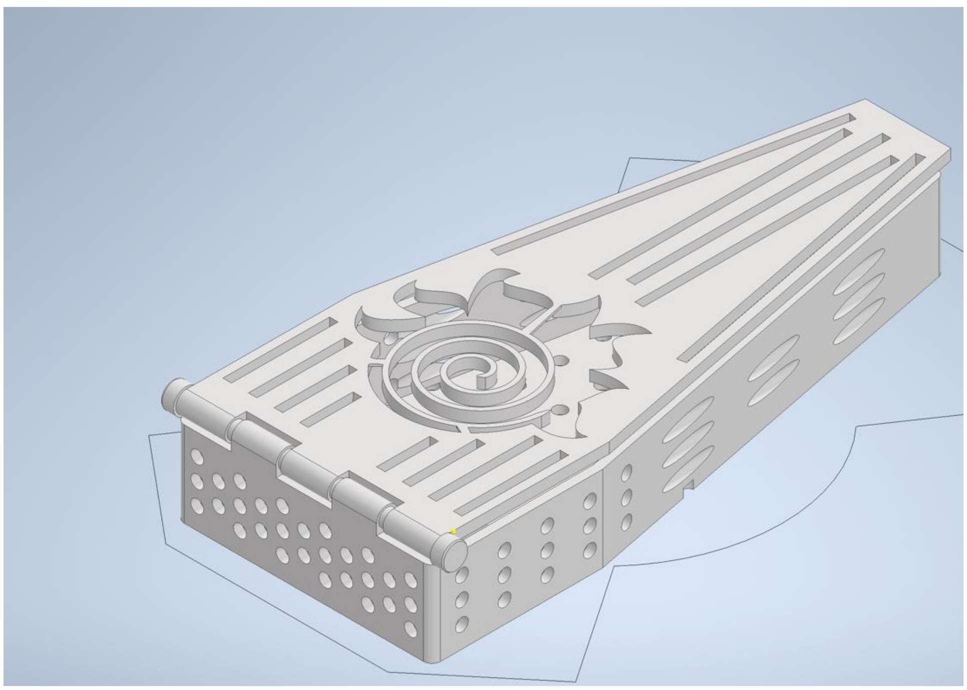


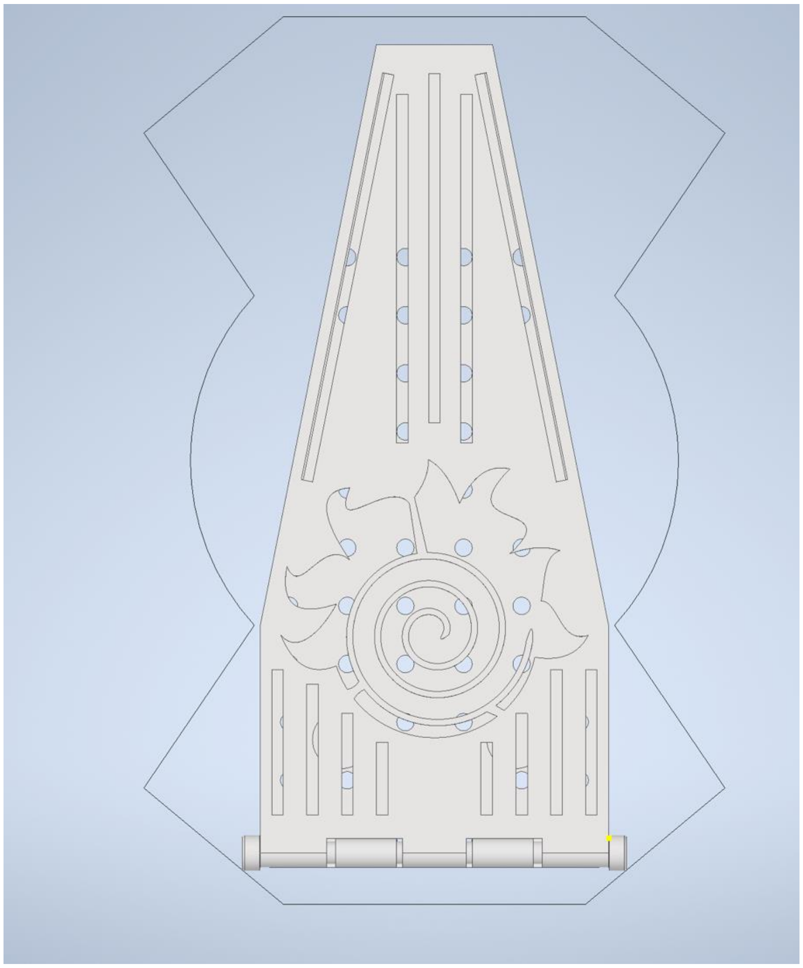
2.

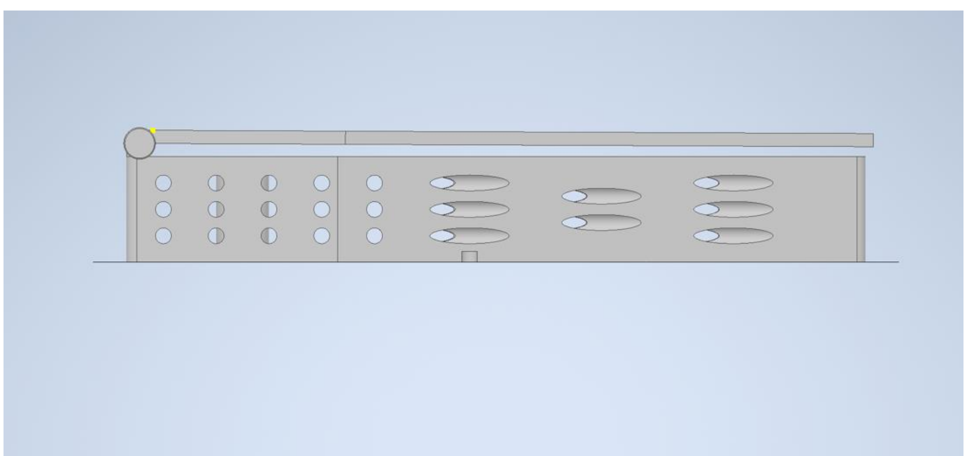





FINAL DESIGN









Engineering Drawings

Appendix D: Design Studio Worksheets