<u>Dry Tap Bottler</u>

Senior EMET Project Design Report

John Butler
Devin O'Hare

Penn State Altoona

Table of Contents:

I Introduction

A. Background and literature search

The Dry Tap bottling project was an idea which was independently conceived by Jack and Devin and initially intended for personal use. Both had brewing as a personal hobby and had in the past looked into the prospects of beverage packaging for their products. The majority findings were that the small scale brewer and home brewer market were largely uncatted to in terms of packaging systems, with existing systems being gated behind prohibitive cost and being overbuilt for smaller needs.

From this the goals of the project became to create a functioning automated bottling system which could complete the primary functions as follows:

- Clean the bottle
- Purge the bottle
- Fill the bottle
- Cap the bottle

As well as cater to the desired market as described through the following features:

- Low cost (within budget)
- Small size
    - Within 2ft x 4ft footprint
    - Weight movable by average person

B. Executive Summary

The automated bottling system is a system intended to output filled and capped bottles of any height.

The function requirements:

1. Load empty glass bottles into the system
2. Purge the bottles
3. Fill the bottles

4. Cap the bottles
5. Unload the finished product

The performance requirements:

The automated bottling system needs the ability to start and stop with the push of a button. It should be able to manually step through the system, as well as step through automatically. The system will be controlled by a Raspberry Pi through a web-based HMI, which will allow for the manipulation of the system by the user.

The system must bottle faster than one can bottle manually.

The system should be able to fill bottles to 12oz. ± 1oz. The system should maintain the majority of carbonation. It should utilize standard keg and compressor quick connect fittings to allow for ease of hookup.

The caps are to be of standard size to allow for uniformity.

The process will require the bottles to be manually loaded and unloaded from the system.

The design requirements:

The completed system will fit in a 2'x4' footprint, and weigh less than 40 lbs to allow for portability.

The safety requirements:

Wiring mustn't be in the bottling area, as the area is expected to have a certain amount of liquid in its proximity at any given time.

Due to the nature of the system, all materials used must be food grade. This includes materials like:

- HDPE
- Stainless Steel
- Aluminum
- etc.

C. Material Science Considerations

The aluminium 1010 extrusion was chosen relatively early in the design in order to build the main frame of the device and supply all major structural support. This was done for a few reasons, the largest being ease of assembly and affinity to water when compared to alternatives. Frame components would just need to be cut and assembled using pre purchased screws and brackets from 80/20.

The rails which would form the tracks for the bottles to flow though were initially to be made of ¼ in teflon sheets. This was in order to assure the smallest friction from the bottles to the bottle line, as it was feared that indexing would be an issue if there was resistance. After consideration HDPE was chosen as the material to use for these rails. This was because of the substantially lower cost and newfound confidence in the available variability in the line indexing.

The capping bells were to be made of stainless steel initially. This was because it was believed that they would need to endure long periods in contact with potentially corrosive fluids and that in order to ensure their longevity this was a necessity. Eventually it was decided that this was not extraordinarily important and 1020 steel was used. The only real necessity of this choice was that the hardness of the material be greater than the caps as to not cut and deform the bells. 1020 steels meets this requirement and is far easier to machine and available as scrap in the machine shop. To resist corrosion the final bells were painted.

D. Project Considerations

Dry Tap would need to:

- fit within the confines of the economic viability to a low capital and low throughput beverage producers, as mentioned before
- Need to be machinable by Devin and Jack in the machine shop on campus, and potentially assemblable using home tools.
- Be usable by a trained operator without major concern for bodily harm
- Not make mention of the intended use of the project to the program advisors and remind them of the existence of such and evil and reprehensible thing as alcohol.

II Research

The initial research for the project was done in the selection process and to determine the potential profitability. The lowest cost automated bottler that was found in the research for this project was a model from Meheen Manufacturing, which was priced to about $40,000. (Figure A-1) While this model and others like it (Figure A-2) are good options for established companies looking to automate previously hand packaged product, the cost is too high for the average microbrewer.

We had the benefit of not being the first group to embark on this project. We had much of Railroad City Automation's final project report that we were able to look over. Mike Hicks was a member of this group, and he was able to help us in the early stages of the project.

III Design Objectives and Specifications

A. Design Objectives

The major motivations for the Dry Tap bottler were to accommodate the undersupplied microbrewer market with an affordable and convenient automation device for intermediate packaging, ie bottling. This would be accomplished by producing a product with the same functionality with a reduced price.

The confines of the budget provided to us in the making of the product was $600 per person, so in our case $1,200 in total. If a product of comparable quality can be produced for this cost, then the project would be considered a success.

B. Performance Requirements (Design Specifications)

The system must start and stop with the push of a button. It should be controlled either by a manual jog, or a built in automatic system. The system will be controlled by a Raspberry Pi. The Pi will be accessible from a web-based HMI, which will allow for the manipulation of the system by the user.

The system must bottle faster than one can bottle manually.

The system should be able to fill bottles to 12 oz ± 1 oz. The system should maintain the majority of carbonation. It should utilize standard keg and compressor quick connect fittings to allow for ease of hookup.

IV Detailed Design Description

A. Conceptual Design Generation

Our goal was to create a portable system. Because of this, we had the idea of minimization in mind from the beginning. We also believed that indexing, or positioning the bottles, would be our biggest issue. As a result of that most of our early designing went into fleshing out the indexing design.

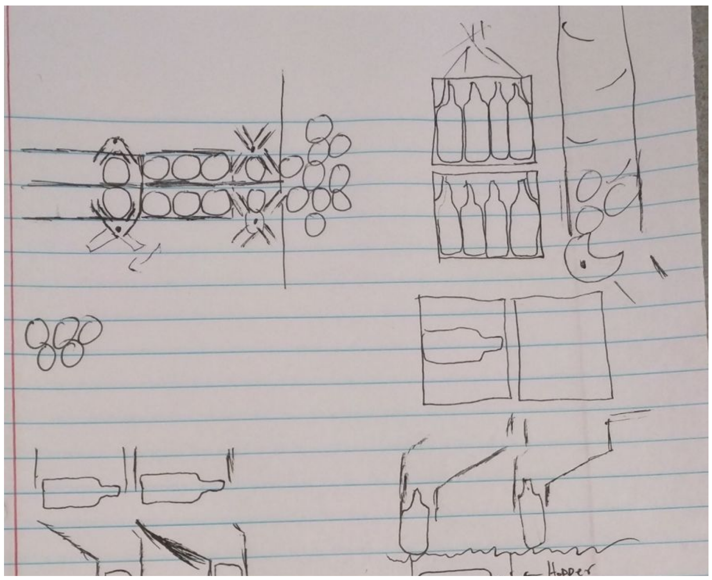

Figure 1: Early Design Ideas

In this sketch in the top right was an idea for a reserve of bottles in a large tower. It would be folded sheet metal into a box, with rips on the inside forming a distinct path for bottles. The bottom shows the means that bottles would be dispensed with a rotating cylinder which would be controlled with an electric motor. Bottles would be feed down a shaft which would reorient them to upright for further processing. (1)

On the left is an early idea for the indexing in line. Shown are 3 stations in the center represented by the 3 circles in the middle. These would have been the filling station the cap dispensing station and the cap pressing station. On either side of this are two sets of rotaries which would push the line of bottles on command and press the bottles into a specific position. These would be controlled with electric motors. (2)

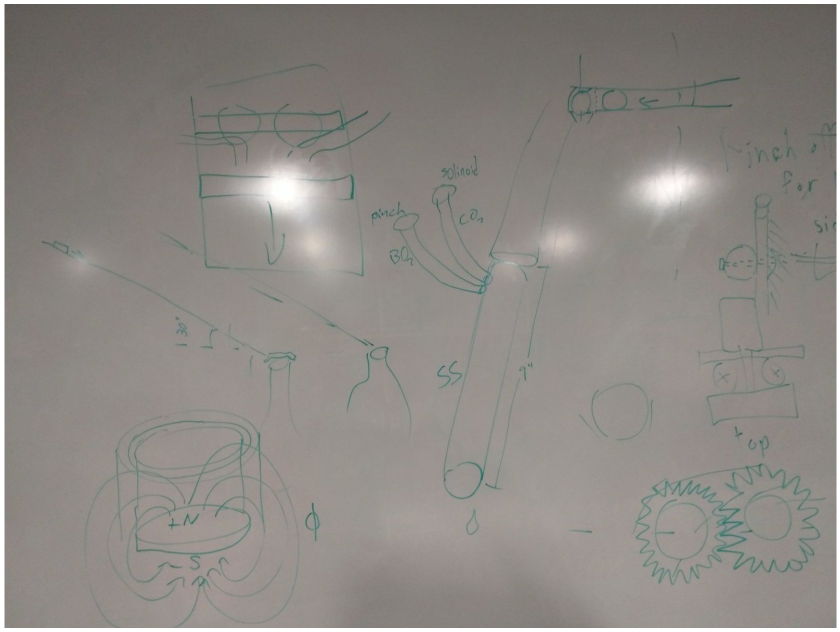

Figure 2: Early Design Ideas

In this sketch, in the center, is the first design for the filling station. The piston would be hooked to a metal tube which would be lowered into the bottle, with inlets for both CO2 and the beverage. This was chosen over leaving the input to be inserted from the top of the bottle out of concern for the bottling of carbonated beverages. It was feared that allowing the beverage to fall freely would result in a loss in carbonation and the potential for a foam over and spilling. (3)

To the right and in the center is depiction of the flow control. The two circle represent the cross section of beverage lines, surrounded by a piston with something to pinch the line with. When actuated the piston would contract and stop the flow. The idea was to avoid contacting the beverage if possible with any piece of machinery. We had also had the idea of getting proper solenoid valves for flow control which would be actuated electronically. (4)

A design for cap dispenser was made and later scrapped as a result of problems in manufacturing. This design was a set of symmetrical sheet aluminium chutes which were dimensioned to fit a bottle cap. They would have been 1 ft in length and had a small set of pins on either side of the rails and angled at 45 degrees. The chutes would be positioned to allow a bottle moving past to catch a bottle cap at an angle and pull it out of the chute on top of it. This design was an adaptation of the design for the Railroad City Automation design researched, but made far more difficult to implement on account of the angles required to have it fit the frame. (5)

B. Selection of Design Concept

1. For the stock of bottles loaded into the machine a simple ramp was chosen in our design, with a sliding index controlled by a pneumatic cylinder for control. This was done directly as a result of the Meheen bottler. The earlier ideas were complex and offered essentially no benefit over the ramp. Out of concerns for cost, machining difficulty, and time we chose the ramp.
2. For the station design early ideas emphasized exacting position for the bottles with the rotaries. It was ultimately dropped after research and the finding of Meheens pneumatically driven slide design. The indexer was far simpilar and the proof that it would function emboldened us to pick it over our initial more difficult and expensive to produce design. The stations were eventually also changed from being 3 stations (Filling, cap index, and cap press) too 2 stations. (Filling and cap press) Cap distribution was to be added in line between these two stations and abuse the movement of the bottles. This idea was taken from the Railroad City Automation design.
3. The fill station did not see major innovation from the initial design outlined above. The idea was simple, easy to construct and we saw no issues with it. We never even came up with solid alternatives to be honest.
4. The flow control did not see major innovation from the initial pinching design outline above. We wanted to keep the beverage from being contaminated, and this was the only idea we had that didn't put the beverage into something that was not trusted for cleanliness.
5. This design was entirely dropped from the final product.

C. Hardware Design

The hardware of the Dry Tap Bottler can be broken down into multiple subsystems as follows:

1. Basic Form
2. Reserve/Drive/Index
3. Fill Station
4. Capping Bell

Design elements of each will be covered in their respective section

1. Basic Form

The basic form of the system can be found below in Figure 3. The frame of the machine was made from Aluminium 1010 extrusions formed a two tiered box, stemming from a long rail on the bottom. In the back is a pivoted ramp. The bottom rail constituted the main line for bottoms to be driven and processed through. It was enclosed on two sides and below using plastic sheet and divided in the center as to create 2 lanes in which bottles were processed in parallel. The two station areas, denoted by the raised platform areas, are for the filling of bottles and the capping of bottles respectively.

The basis of this shape was conceived as a bastardization of the Meheen design found in research. (Figure A-1) Their design was extremely simple and allowed for continuous parallel rails for bottles, making farther models to increase throughput a potential. We took this idea tried to further simplify it. The framing used in the Meheen model was steel and welded together to shape. Using aluminium extrusions allowed parts to be assembled piecemeal, and deconstructed and reconstructed if need be. Meheen's bottler would lift both the fill rods and capping bell in tandem on a single frame. While this method requires less controls to operate, the hardware must be more complex. Considering the fabrication difficulties, this was cut and two section run and cut independently was the final design. The last major deviation from the Meheen bottler design was the cutting of a set of wheeled legs, creating a cart for the device to be transported on. This was cut because it was assumed that the user would be fine making do using the machine on the ground or finding a table for it, so ultimately cost.

2. Reserve/Drive/Index system

This section refers to a single component which performs the functions listed in the title and will be called the ramp and slide. The ramp was a sheet of plastic with rails on 2 sides attached by a pivot to the main frame, which allowed for a variable incline. The slide was a piece of plastic of dimensions ½ x 2.4 x 4.8 in, which was attached to a pneumatic piston underneath the ramp, and could be actuated from under the ramp entirely to extended out the full 2.4 in dimension.

Figure 4: Ramp & Slide

The function of the ramp and slide worked as follows. Two lines of bottles would be loaded onto both the lower center line area at the bottom of the ramp and onto the ramp. The weight of the lower bottles would keep this position in place, while the gravity of the bottles on the ramp would act to push the line down and forward though the rest of the process. When the drive piston was actuated and the slide was pushed out, the lower line of bottles would be pushed forward with it eventually to 2.4 in when fully extended. In the meantime the bottles on the ramp would have the space in front of them opened and they would be free to drop onto the now open space on the slide. Once the new bottle was in place the slide was retracted quickly to allow the bottle on top of it to drop to the initial position of the bottle in front of it.

This process also acted as our indexer. The bottles themselves were 2.4 in in diameter at their fattest point, and so with proper operation would move the line forward one full unit bottle length. The dimensions of the line were molded to be stationed in units of bottle lengths.

In order to have the bottle line off of the ramp consistently fall in the correct orientation a bar across the line was added. This bar would clip the tops of the falling bottles and allow the force acting on them to create a rotational moment which would reorient the bottom face into a position to fall.

3. Fill Station

The fill station was fairly simple. When bottles are pushed under it, tubes are lowered into the bottles. Once fully extended, the tubes begin to deposit fluid into the bottles until they are full. Once full, the flow stops and the tubes retract.

The filling tubes are 7 inch long segments of PVC. They are extended and retracted by tying them directly to 7in. stroke pistons. The tube is connected via a brass right angle pressure fitting, giving us the ability to drill and tap them for interfacing with the piston.

Flow is controlled through another piston that pinches and releases a pressurized vinyl tube. This can be seen in Figure 5.

Figure 5: The pinch mechanism

4. Capping Bell

This portion of the system does the actual capping. Bottle caps are designed to be press-crimped through a linear action. Our capping bells accomplish this through the use of our pistons.

Our pistons are 2in in diameter which, at a pressure of 100 psi, yields a final pressure of 314 lbs. As it turns out, this is exactly what is required to be able to cap our bottles.Pistons are connected to capping bells directly by drilling and tapping the bell. Due to the volume of the piston being relatively large and the flow rate of the air being relatively low, it takes a few seconds for the piston to get up to pressure. Because of this, capping takes a few seconds.

D. Software Design

It has been a goal since the beginning of the project to keep all of our project accessible. We also decided early on that we wanted to open source the software, and so all of the development took place in a GitHub repository. The repository can be found at https://github.com/fattredd/drytap. Using git to version track our software came with many benefits. First, git made it easy to keep track of my changes over time. This made writing up weekly project reports a breeze. Second, git allowed me to make different branches of the project to try different methods of getting a working finished product. The biggest benefit however, was that using git with GitHub meant that we never had to worry about accidentally deleting things, as it was all backed up. GitHub also allowed me to be able to develop from home without the Raspberry Pi at all.

Software was written in 4 languages, most of which ran on the Raspberry Pi through an Apache web server. The breakdown of the languages can be found in Figure 6, and the diagram in Figure 7 outlines how the software functions.

Figure 6: Software language breakdown

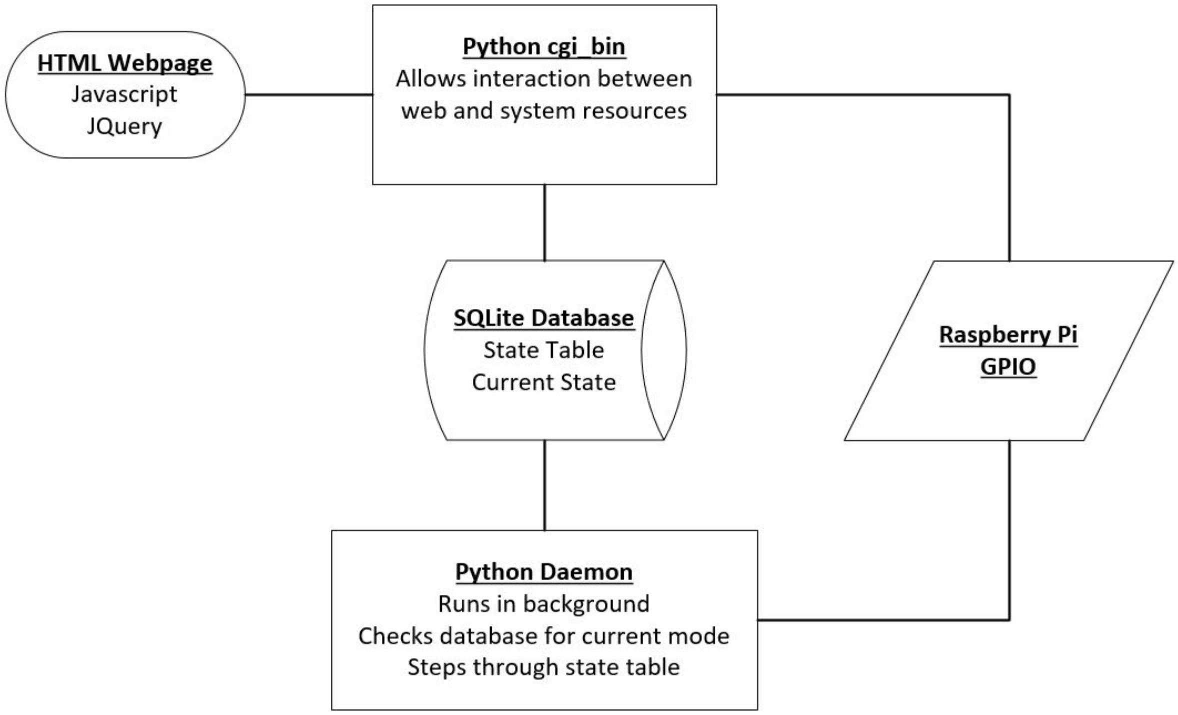

Figure 7: A flow chart of the software

Due to the nature of a web server, it is important to realize why a web-oriented programming language like Javascript must not be allowed to access system resources. Since the Javascript runs client side, anybody can modify it and would be able to do anything they wanted. As such, Javascript is completely unable to touch server-side resources. This includes the GPIO and any local databases that we might have. To solve this issue, we can utilize a server-side language like Python. By telling our Apache web server that we want to be able to run Python scripts through web calls, we allow clients the ability to access server-side resources without giving them server access. In this way we can have Javascript make an external request to our Python script, thereby allowing Javascript limited access to system resources through our Python scripts.

The software itself breaks down to two systems. The first is the manual system, which runs off of an Apache web server. An HTML page serves as an interface that allows users to interact with Javascript embedded within the webpage. The Javascript does a number of things using a library call JQuery, which provides a simple way of manipulating the HTML page, as well as call external pages.

Once the webpage is loaded, Javascript calls a Python script which polls the current state of the GPIO pins. The JQuery takes the current states and uses the information to update the color HTML buttons. This way users can keep track of the current state without looking at any code. This function gets called again every 500ms to ensure the buttons stay up to date.

When a button is pushed, Javascript calls another Python function that changes the state of a given GPIO pin. The Python script simply toggles the GPIO pin and calls the button's state update function.

The second system can be controlled by the manual system, but otherwise is completely an independent system. On bootup of the Raspberry Pi, a process is started in the background. This process is known as a 'daemon'. The daemon is written in python, and interfaces with two things: the GPIO and an SQLite database.

The SQLite database contains two tables. Only one of which gets modified during runtime. This table is autoOn.db, and it holds the current state information. This information is really just the whether or not auto-mode is enabled, and whether or not the system is currently paused. The other table is states.db, which contains the state table data. This is information such as which pins are high in each state, the delay for each state, and the current state.

The aforementioned daemon is constantly checking the auto-mode, and when it is set to active, it begins its work. The daemon steps through the state table, setting outputs and then waiting for the desired delay. After each step it checks auto-mode variable again to see if it should continue.

Meanwhile the web server side is able to control the auto-mode and pause variable through the use of a button. The same process described above happens to the auto-mode and pause buttons, but instead of updating the GPIO, they store the output data to their respective databases.

E. Prototype Development and Testing

In actuality it is our belief that this project in entirety is in effect a prototype. The design as stands is a solid proof of concept and displays the fair possibility of a product like our existing for a similar price point, but nobody would buy our final product of its intended use.

The greatest example of prototyping and development of design was in our capping bell, the fitting on the front of the capping cylinder which would be making actual contact with the bottle cap.

At the onset of the project it was thought that the design of the capping bell would be something that was readily available online and would take just manufacturing time on the lathe to make. This turned out not the be the case as the dimensions ripped from online sources turned out to be bad. We set about solving the issue ourselves as a result.

Our initial test for the online design was a 3-D printed ABS piece which was pressed using the arbor press in the machine shop. It did cap the bottle, but was heavily damaged in the process. This was expected considering the materials used, so not much was thought of this and the test was considered a success.

Figure 9: Plastic capping bell

The next piece was made from aluminium and was actually our only part made on the CNC mill. This was also used on the arbor press and produced similar results. Bottle was capped, damage was done, and all seemed well so form was taken to the lathe with a piece of 2 in 1020 steel billot. **unfortunately we don't have any pictures of this piece*

Our finished steel mock-up was grossly oversized because of the fact that it was cut from the 2 in diameter stock we had. When tested it again on the arbor press and it capped a bottle so tight that we initially couldn't get it off of the bottle. This was mildly concerning so we did a few tests. The crown of the caps ended up being sheared off entirely in about half of all cases. This was of course unacceptable, so we decided to go about righting the design.

Figure 10: Bad Steel

Our first Idea was to open the cut the pocket wider to allow it more relief, so we did that. It worked fine for making the capping less destructive. Next we set up a rough testing rig for our capping cylinder to see it work in a mirrored situation to how we expected the final design to work.

Figure 11: Test Rig

Of course it didn't work. Our cylinder was 1.5 in in diameter and we were operating at an expected working pressure of 90 psi. When it didn't cap immediately we started incrementing up the pressure and trying again. It never worked up to 120 psi, the highest available pressure the shop had in supply. From here we came to the conclusion that the fault most likely lay in the design of the bell. The hard 90 degree angle of the pocket was not effectively distributing the force of the piston. We decided to add a chamfer to the entire pocket. This started at 5 degrees and then became as much as we could manage on the lathe. We measured this to around 8 degrees. It still did not function, though it was noticeable easier to cap on the arbor press.

Figure 12: Final Bell

At this point we conceded that the amount of force we expected to need to cap a bottle was fundamentally low, and so moved to getting a larger diameter pneumatic cylinder. We found two identical 2 in diameter cylinders in the project graveyard and set them up on the rig. Eventually got to a pressure of 100 psi and capped a bottle on the rig. It takes about 3 seconds for the cylinders to reach the desired pressure on the campuses system and cap. This is on account of the large volume of the cylinders, on account of their 7 in. stroke, 4 in. larger than initially desired. It worked though, and rather than spend the money on new cylinders, we took the hit and kept them.

V Cost Data

Our budget is pretty cut and dry. We spent all of our money on materials for building our project. The only materials we didn't use are some of the microcontrollers, which got scrapped when we got the HMI donated.

Date	Vendor	Item	Cost
8/2/2017	McMaster Carr	Plastic sheet & Rubber Gasket	$59.57
8/2/2017	80/20	Aluminum Extrusion, hinges, screws	$272.03
8/2/2017	Bimba	Pnuematic Cylinders	$224.47
8/31/2017	Digikey	Power Supply, Voltage regulator, transistor array	$44.87
8/31/2017	McMaster Carr	Aluminum angle 1x1x1/8 2'	$11.78
9/13/2017	Digikey	Microcontrollers	$23.39
9/29/2017	McMaster Carr	1x1x1/8 8'	$30.41
9/29/2017	80/20	Screws & nuts	$67.86
9/29/2017	Automation Direct	Rate limiters	$31.00
10/18/2017	Amazon	Raspberry Pi 3	$78.94
10/24/2017	Digikey	Terminal Block Plug	$11.23
12/5/2017	Home Depot	PVC tubing	$21.22
		Total	$876.77

Figure 13: Cost breakdown

VI Results

Far and away the largest obstacle we faced in this project was manufacturing time. We did not chose to have any of it produced off site and so all workload fell to us. Our knowledge of machine was relatively poor and a fair amount of time was required to learning. Combined with a loss of work time over the summer between 403 and 440 and and a few choices which exacerbated the issue, we were never ahead of schedule for building.

The earliest issue we faced was in poor material selection in the screws used for assembly. They were too long to fit through our purchased bracket without digging into the aluminium framing, essentially making them useless. More screws of a proper size were order immediately after this was noticed, but in the week before they arrived assembly of framing required that screws be ground to to size before they could be used, taking about a third of assembly time.

The framing itself was also simply not as easy to assemble as it was assumed it would be. Assembly was initially assumed to only require at most 2-3 hours for the full unit once all parts were manufactured. This was not the case and the total time just in assembly was almost 4 times more than this. Reasons for this were either the need in almost every step to refine cut pieces before assembly or the basic difficulty of fitting screws to brackets.

Another poor decision made in the build was the choice to save on our cost and produce our own brackets. These were simple 1in x 1in aluminium 90 angled single hole brackets. From 80/20 our initial distributor we were paying about $4.00 each for these brackets. We used well over 100 of these in the final build and it was determined that that cost would be ridiculous and it was decided not to take it. A 6 ft extrusion of aluminium 1010 1in x 1in angle stock was ordered with the idea of drilling and cutting our own. This ended up taking the pair of us a full week of effort with far more inconsistent results than the previously purchased pieces. Holes were initially not reliably indexed and cut by eye. This resulted in a large number of brackets with poor dimensions. At the time our design made no considerations to the inconsistency in these brackets and key dimensions were chosen based on this length being a constant. Brackets had to be carefully selected and in quite a few places completely removed after assembling.

All of the plastic components had some level of warp in them after cutting. In most casing this was not pronounced because of rigid support. On the ramp in particular thought this was an issue. It was explained to us by one of the other groups that it was common for there to be some internal stresses from manufacturing that was released upon machining, which resulted in these prominent curves. Some attempts at using heat treatment to fix this were unsuccessful and no further action were taken on account of time.

One of nice aspects of our design which was not initial considered to be nearly as valuable as it ended up being was the ability to adjust fine placement of almost all pieces of the design very quickly. The brackets could be loosened and slid down the line they sat in without much time or effort invested. This was mostly as valuable as it was in the final build as a result of poor machining tolerances. With the number of parts cut, these errors would often compound and result in dimensions vastly off of the intended. In most cases the parts were remachined or remade, but as time was running out the standard became to adjust the frame in this manner to make pieces fit. In the end there were no major noted consequences of this and the final product would not have functioned in any capacity without it.

The final product is shown below. It was not very pretty and had lots of obvious machining errors and things were not refined as we would have liked them to be.

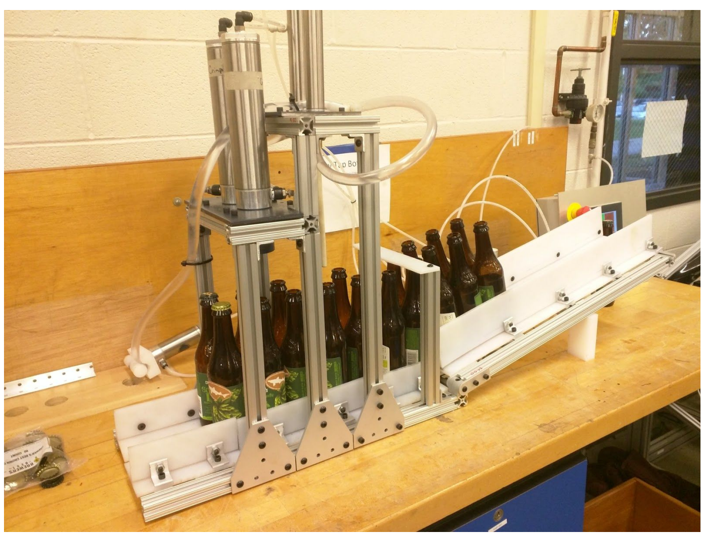

Figure 14: Final Product

The indexing subsystem worked almost exactly as intended, but was not consistent. Bottles coming off of the ramp were lucky to have a higher than 50% of successful transition onto the slide. This value was vastly improved with move forward pressure from bottles behind. The system wouldn't work though with a ramp at an angle steeper than about 15 degrees. This meant that for best use the system had to always be full, which makes the automation system need constant supervision, making it borderline useless. This could have been fixed with a combination of a lesser friction slowing the bottles and the plastic rails on the ramp warping less. The indexing in the stations was very consistent in its placement. This placement was also ever so slightly short. If we had literally gotten a piece of duct tape and put it over the front of the slide, it would have been perfect. Really the answer would have been to machine the slide more exactingly.

Figure 15: Slide

The filling station was one of the last sections made. Drilling and taping of the elbow was simple and had no issue. The PVC tubing suffered from the same issue as the other plastic parts. One in particular would list by almost a half inch at the bottom on average from vertical. This created a potential for the rods to completely miss the bottles and be press out and break the rod. The tube could be pressed back into vertical by hand, which would hold for a few seconds, allowing operation again without proper automation.

Figure 16: Fill Rods

The capping press was the most tested substation and saw a fair number of iterations to work. As a result it worked great at the time of demonstrations, only to fail catastrophically. (blow up a bottle completely) We still have no idea how this happened and are convinced that it was a result of a defect in one of the new bottles brought in for these demos.

Figure 17: Capping Press

The flow control or pinch was the last component made and was honestly not even remotely close to how we had hoped it would be. Initial design called for 2 lines to independently be run from the beverage reserve (keg) and to both be pinched. Our final was a single line up from the keg which could be pinched, and was later on split to allow for both fill tubes to be fed. This was done simply to be able to show an operating product, even if it was bad, and it was. The fill was the most inconsistent component. One bottle would normally overflow and the other be under the intended fill line, whether it be the left or right line was random. Nothing was right about it really other than the fact that it could be turned on or off at will.

Figure 18: Pinch

VII Conclusions

The biggest thing that the team got out of this project was an appreciation for the complexity of automation tasks. Early designs and ideas were complex and grandiose, which in hindsight would never have been feasible. All of the final components implemented were the most simple ideas we had found and even then some were not finished and others had to be completely cut.

Other largest finding from this project was that hand machining is a terrible thing and the reason people make careers in doing it is because it takes real knowledge and skill to do right. If we were to start again in fabricating the bottler we would have attempted to contract it out and never touch it, or made heavy use of the CNC mill and lathe. Maybe the laser cutter too for the plastic pieces. This would have required using different plastic, but it would be worth it for the saved machining work.

Having done this project I expect to in the future take greater stock in the value of machining constraints.

VIII Future Work

At the end of the day, what we've produced is a prototype of the system. A lot of work needs to be done to have a final product. For starters, we need to work on getting the indexing more consistent. This can be accomplished by revising the sliding mechanism to be smoother. Getting this right first is the first step to getting a consistently packaged bottle.

Our biggest issue however is the capping. We need a larger chamfer on the capping bell so that the bottle is consistently fed into the bell properly. We would also like to buy new pistons with a shorter stroke. This is so that when a bottle does fail, it doesn't fail so spectacularly. Our only broken bottle of the project started by merely cracking, then because the piston continued to exert force, it started to break. After breaking for about ½", the piston began to throw pieces of it, causing a lot more danger than necessary.

When that is accomplished, we can move on to the issue of the fill rods. Simply put, they are not consistent. They only successfully enter the bottle about 95% of the time. While this is pretty decent, it simply isn't good enough. I am under the impression that having a more consistent indexing slide will likely solve this problem, but it's hard to be sure until the work had been done.

Lastly some documentation need to be done on the mechanical front. One of our early goals was to have the entire system (electrical AND mechanical) to be completely open source under the MIT license. While the software and electrical systems are fairly well documented, the mechanical is basically just a CAD drawing. It would be helpful to document the final counts of bolts, tubing lengths, etc. This way we could easily create a marketable kit with everything a customer needs to be able to build one themselves.

Once the documentation is created, I would very much like it to all be packaged together so that anybody who wants to can use our plans in nearly any way the want.

IX Appendix

5. Advanced professional detailed drawings
6. Electrical schematics
7. Software manual – procedures for using your software
8. Listing of any computer code
14. Progress reports
15. Poster
16. Final Presentation printed six slides per page

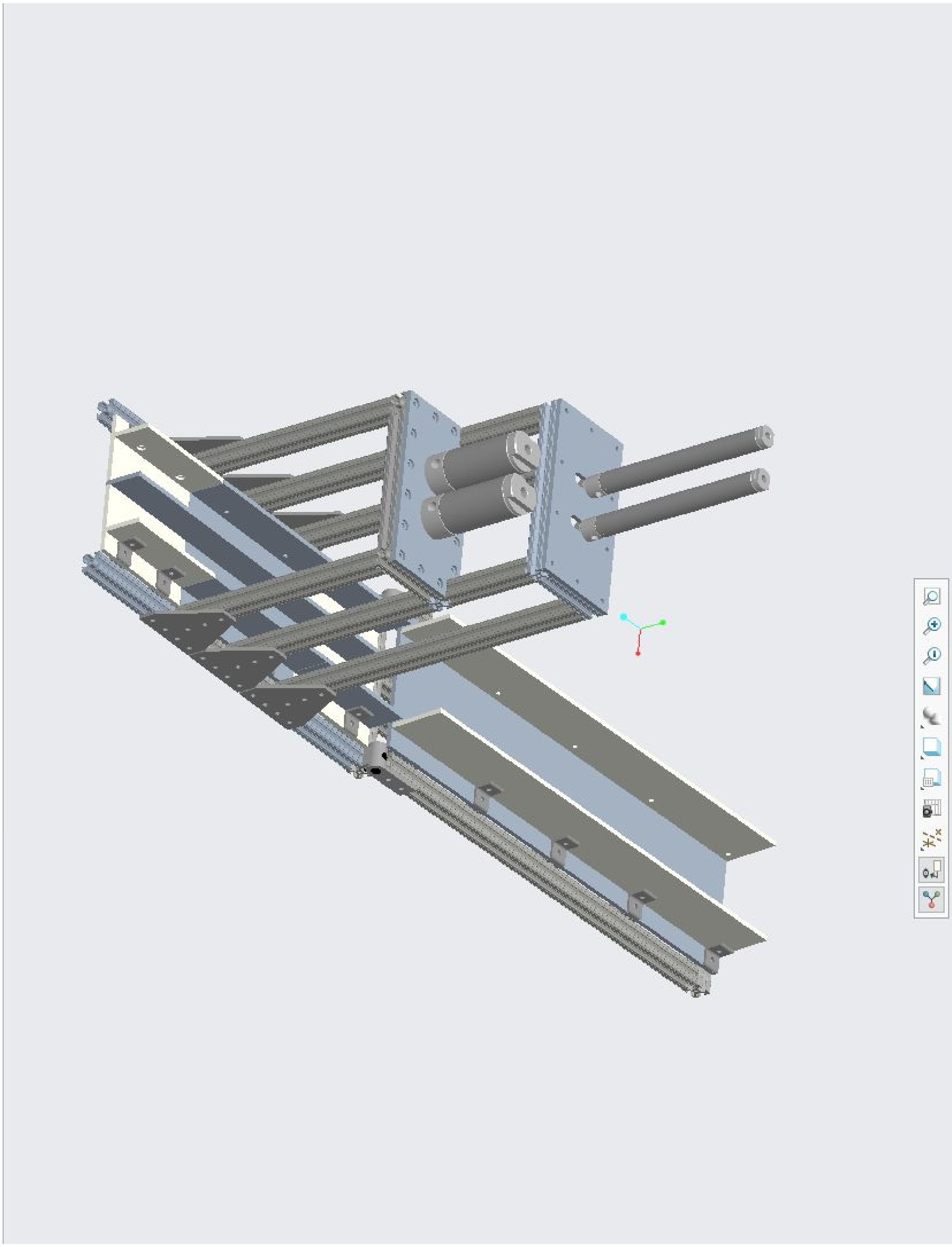

Figure A-1: Full Assembly

Figure A-2: Fill Cylinder

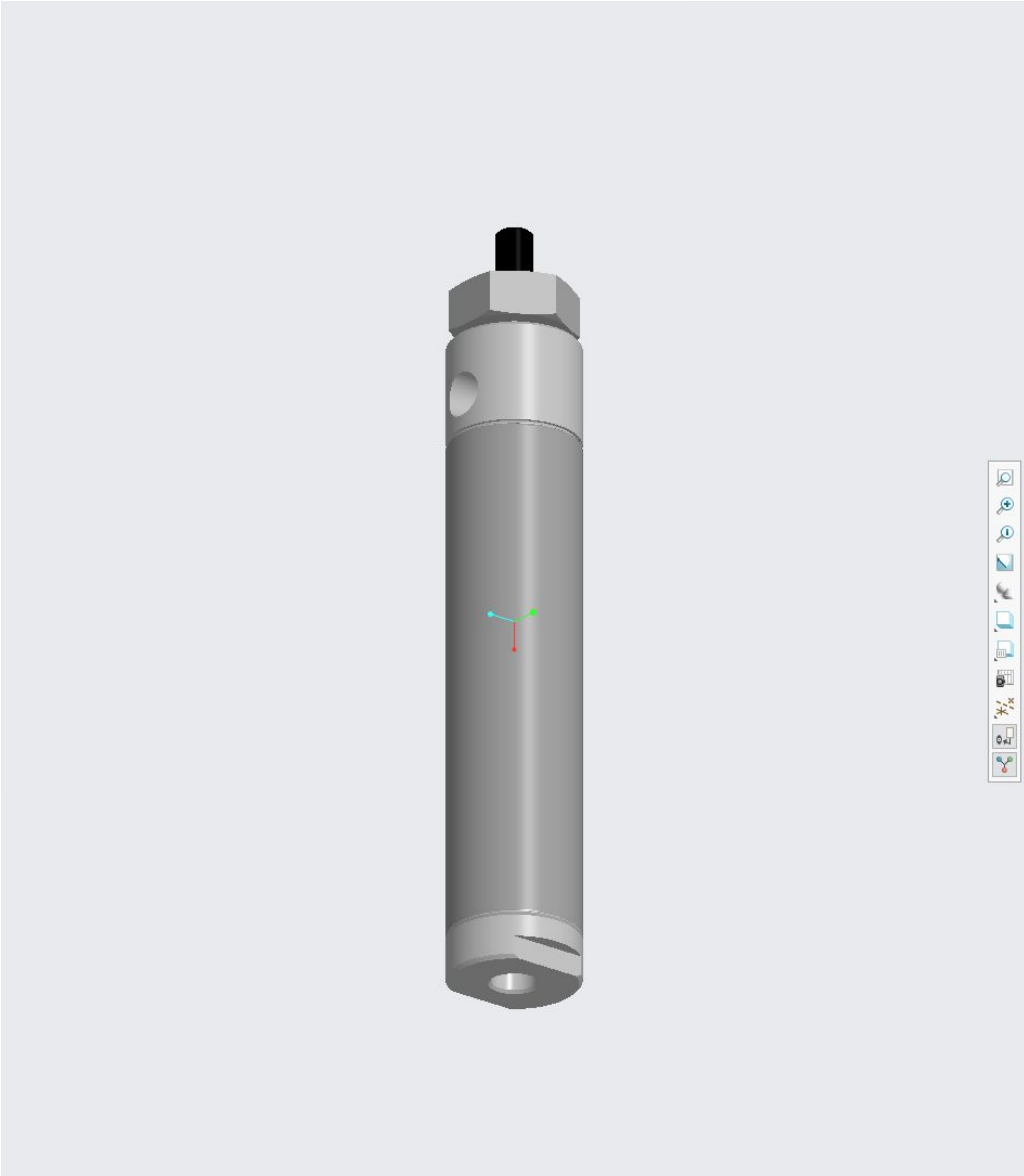

Figure A-3: Slide Cylinder

Figure A-4: Press Cylinder

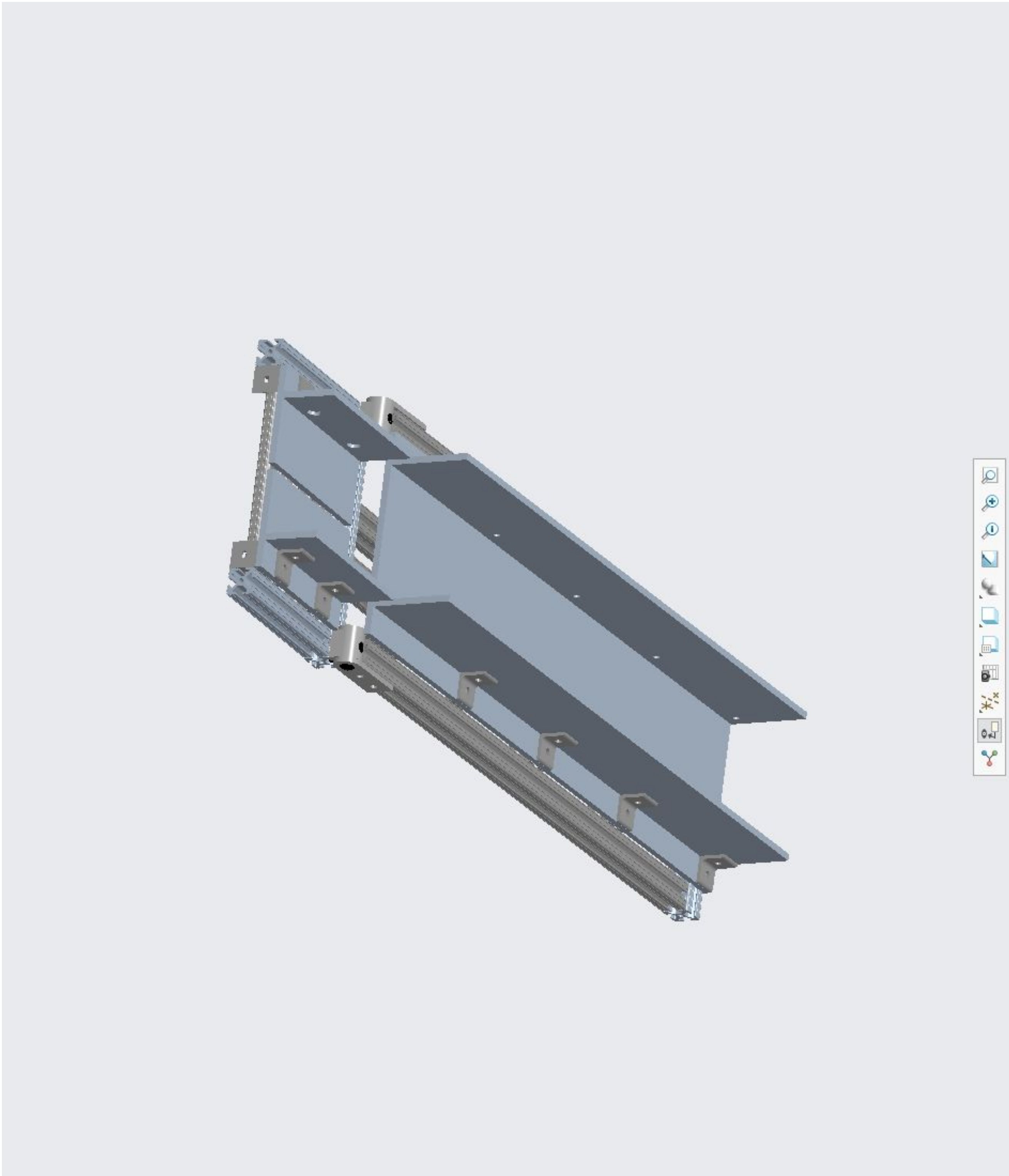

Figure A-5: Ramp Subassembly

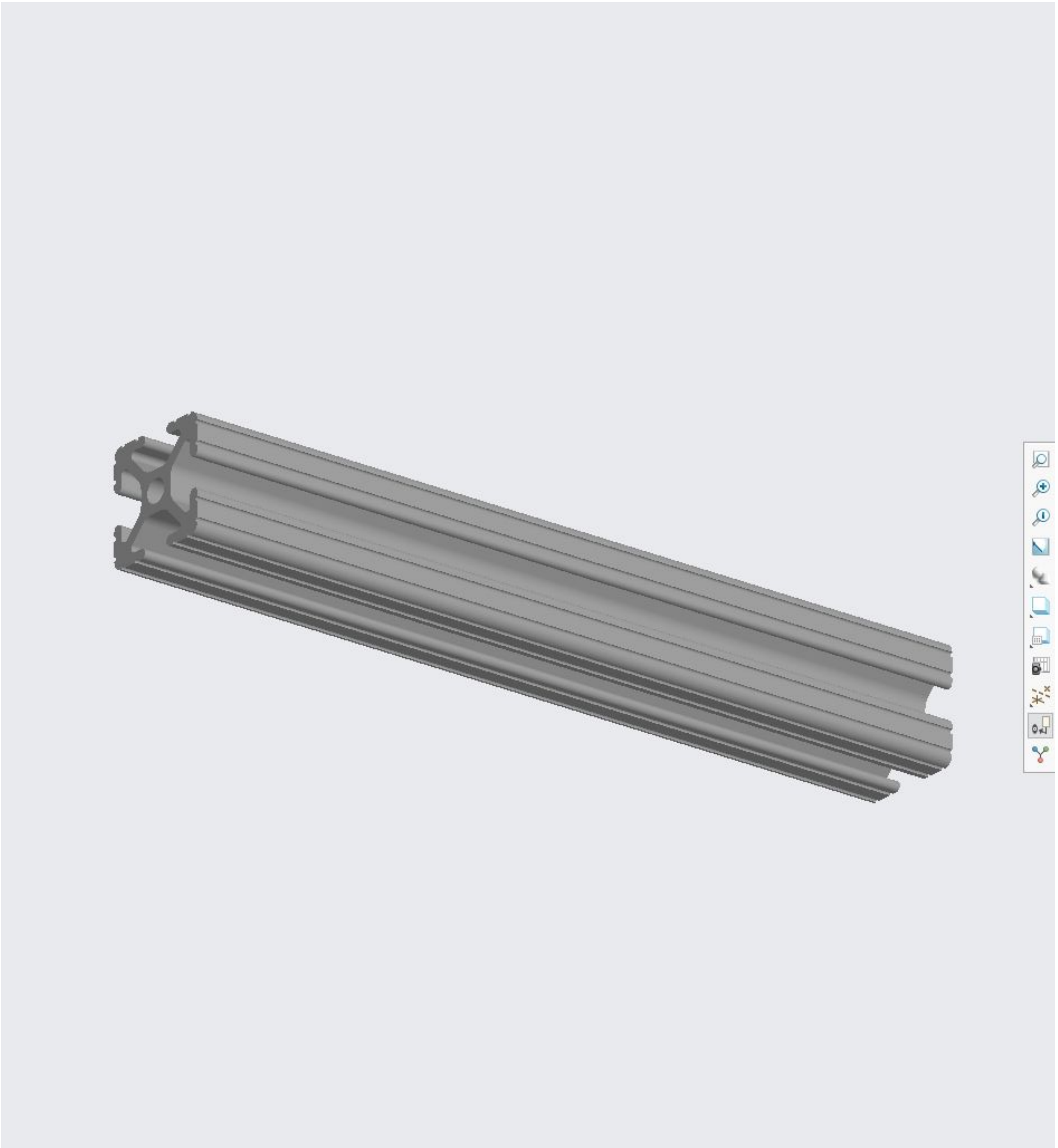

Figure A-6: Aluminium Extrusion

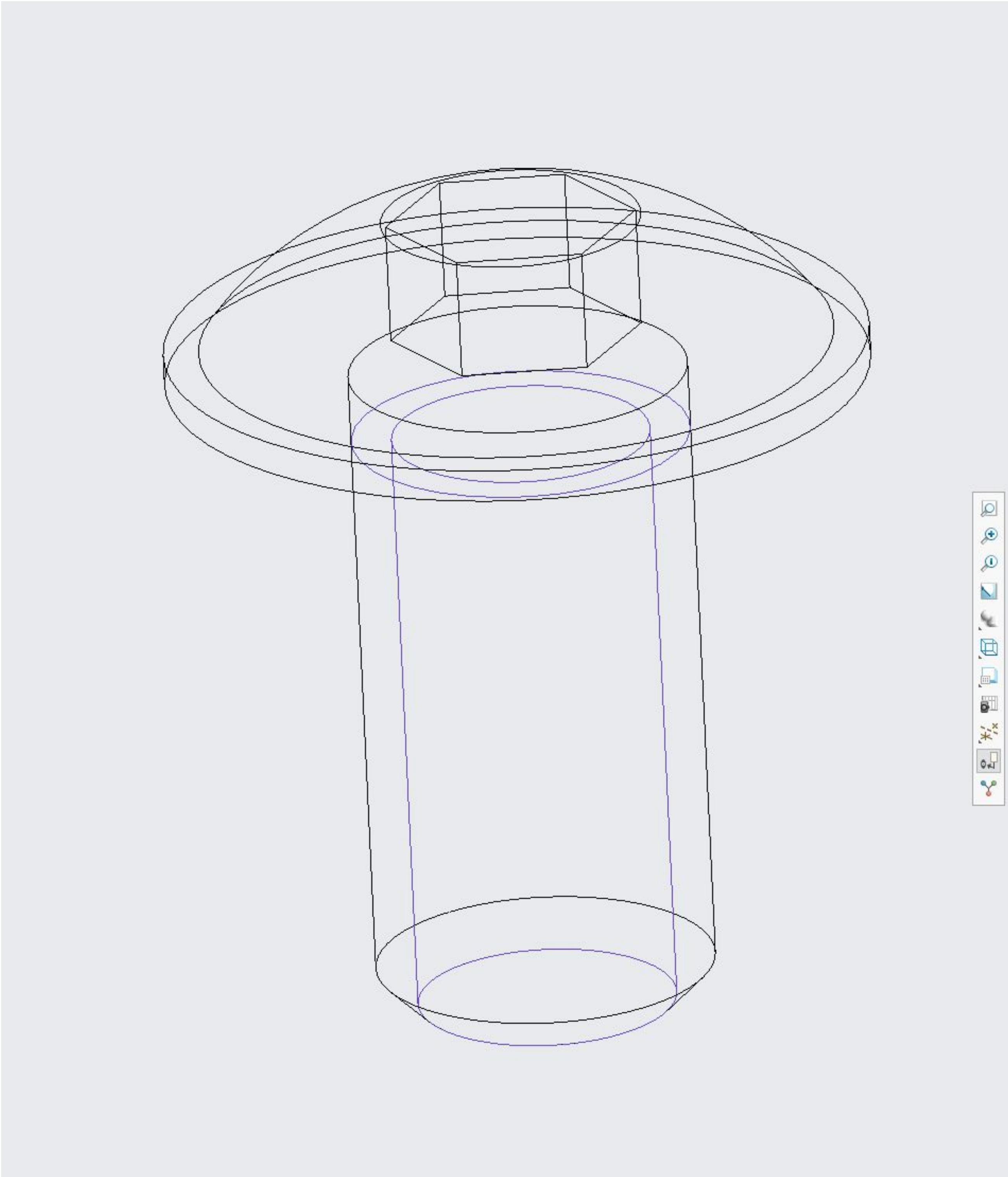

Figure A-7: Screw

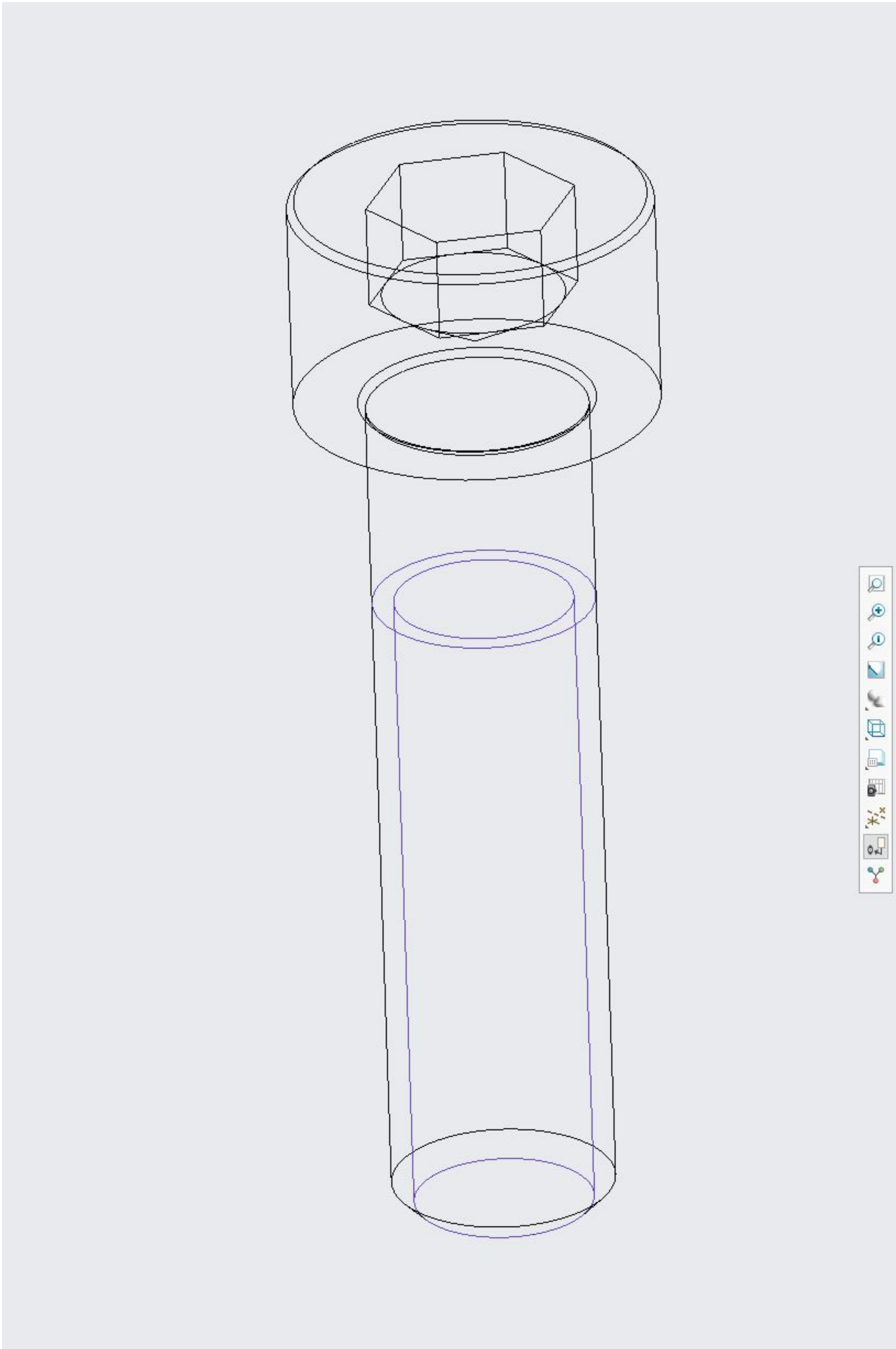

Figure A-8: Screw

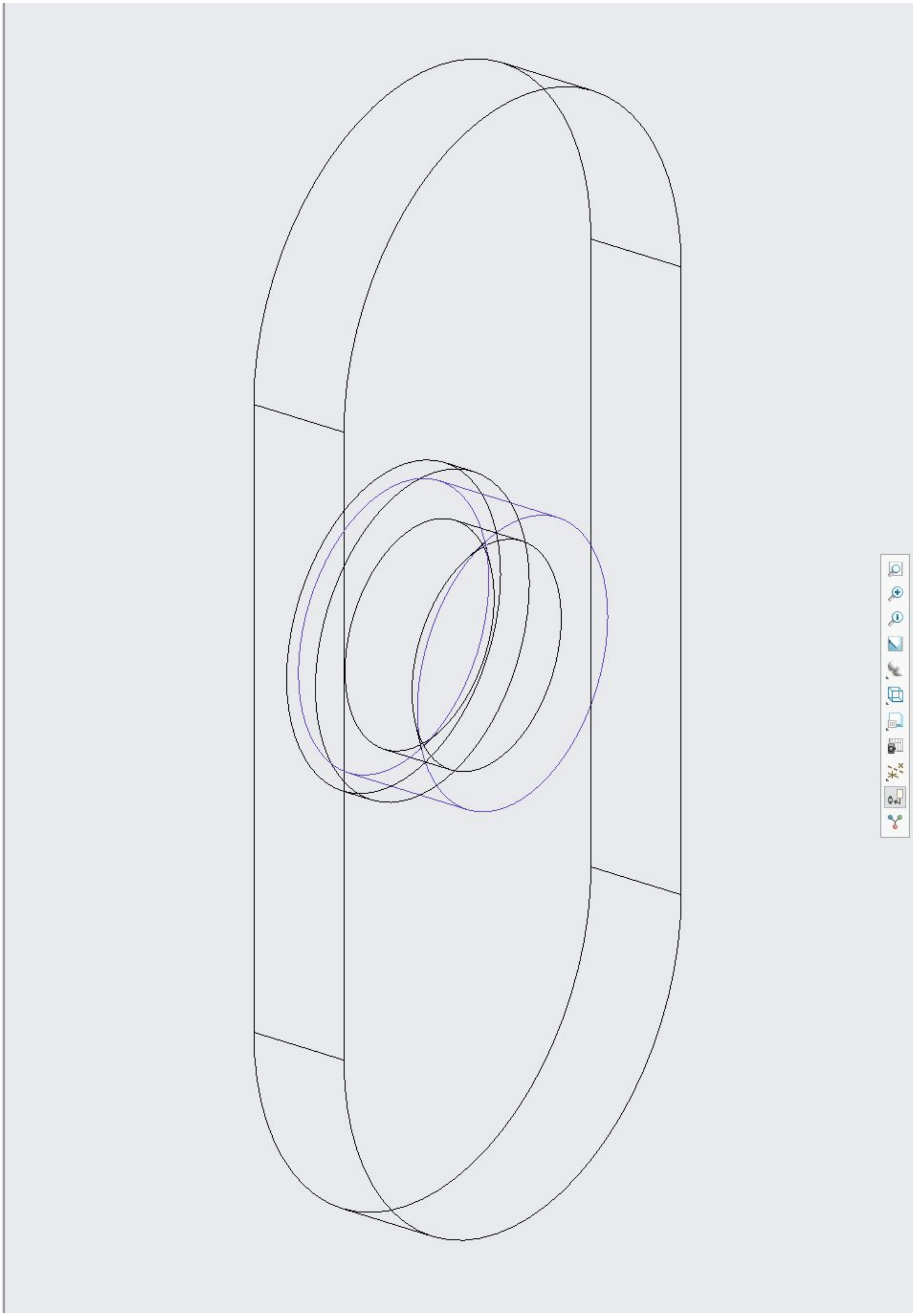

Figure A-9: Slide Nut

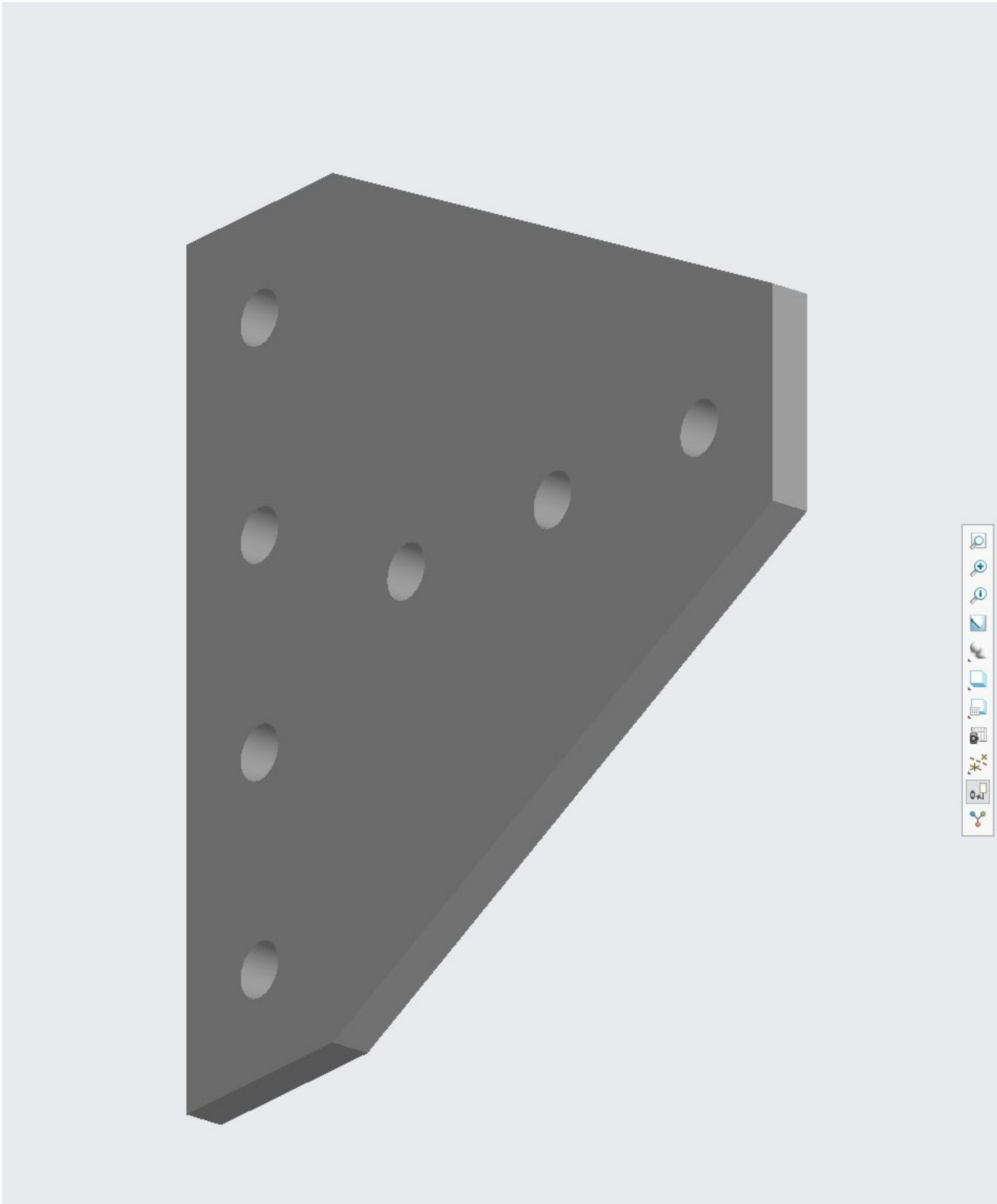

Figure A-10: Plate

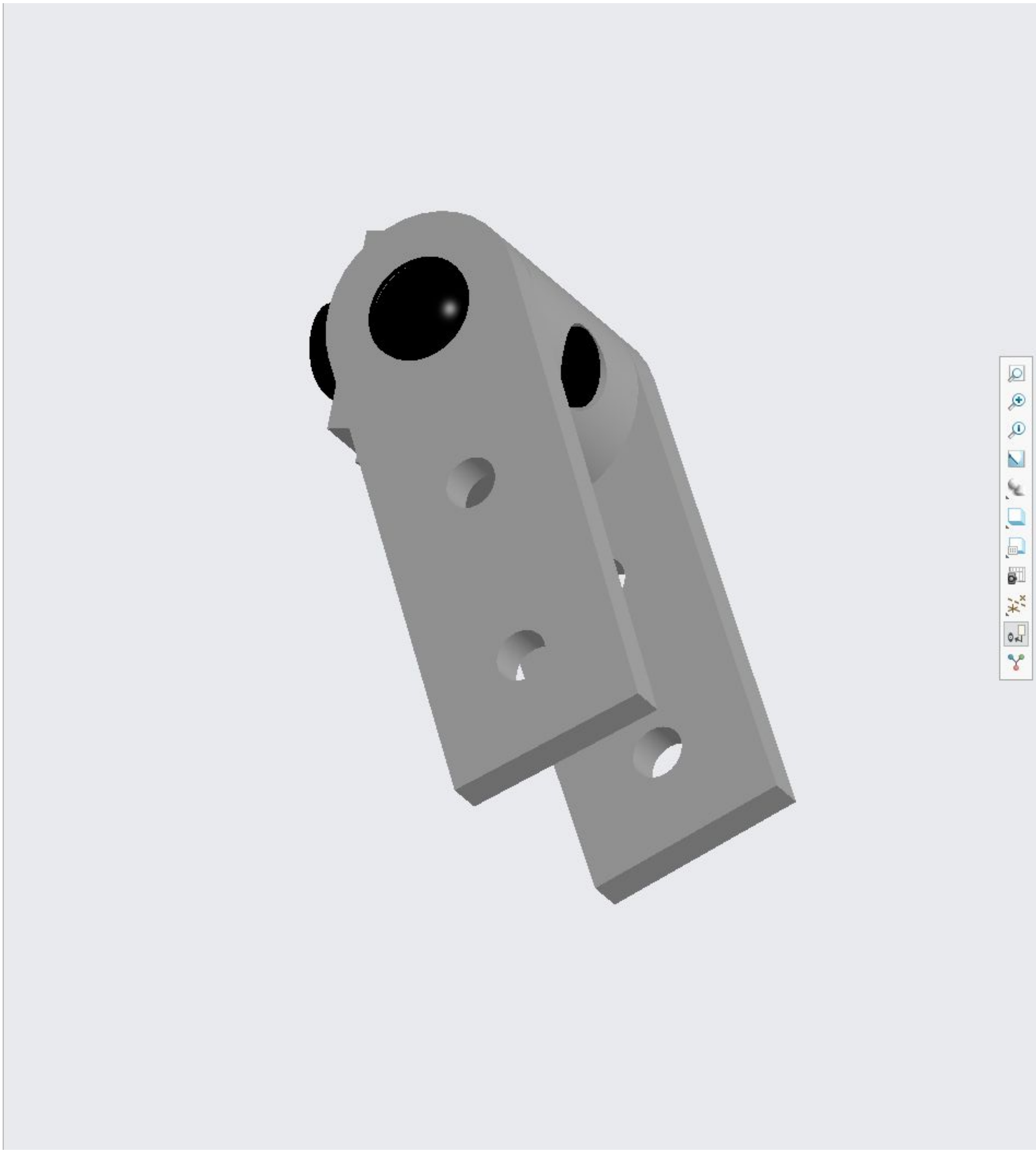

Figure A-11: Ramp Pivot

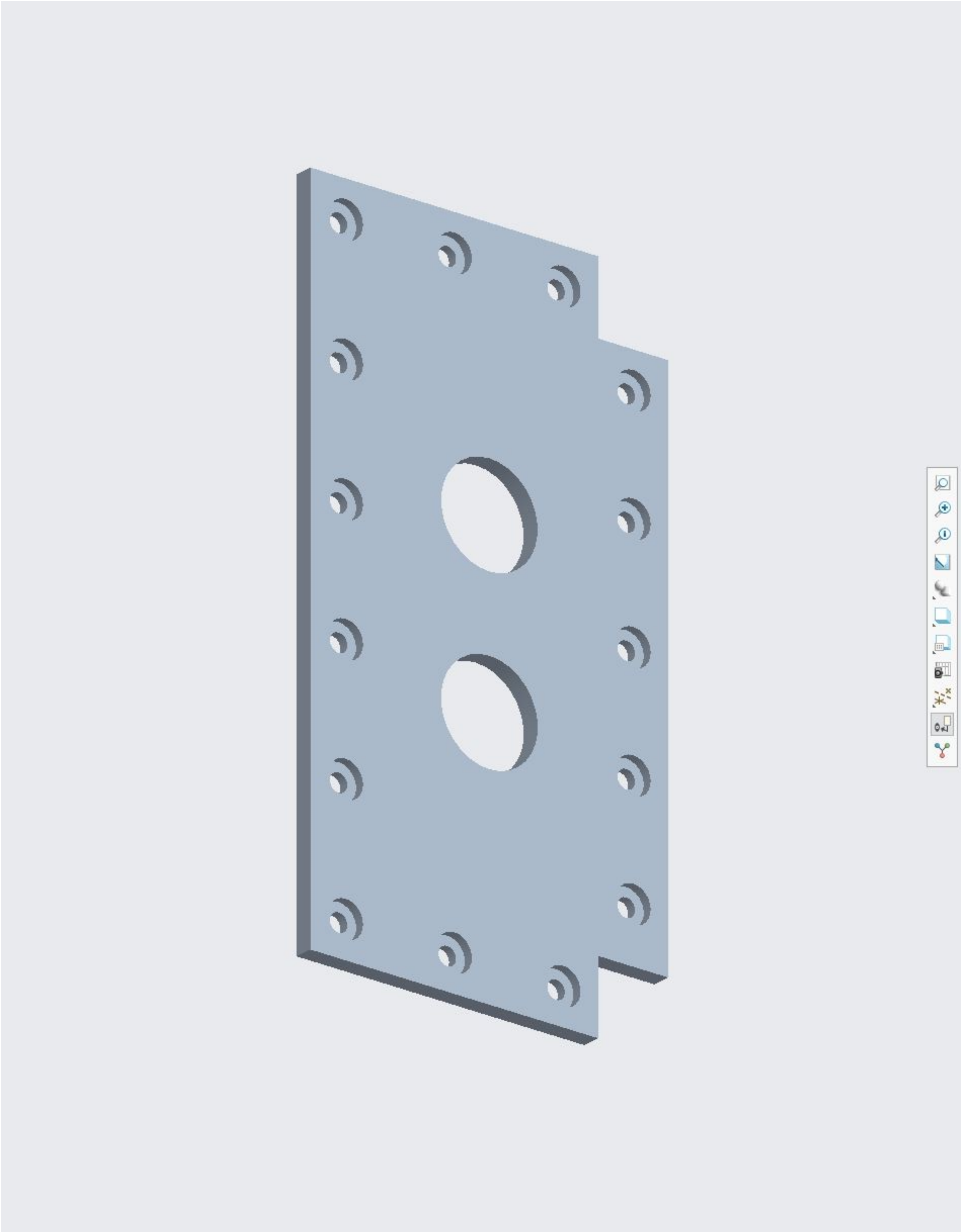

Figure A-12: Press Cylinder Mount Plate

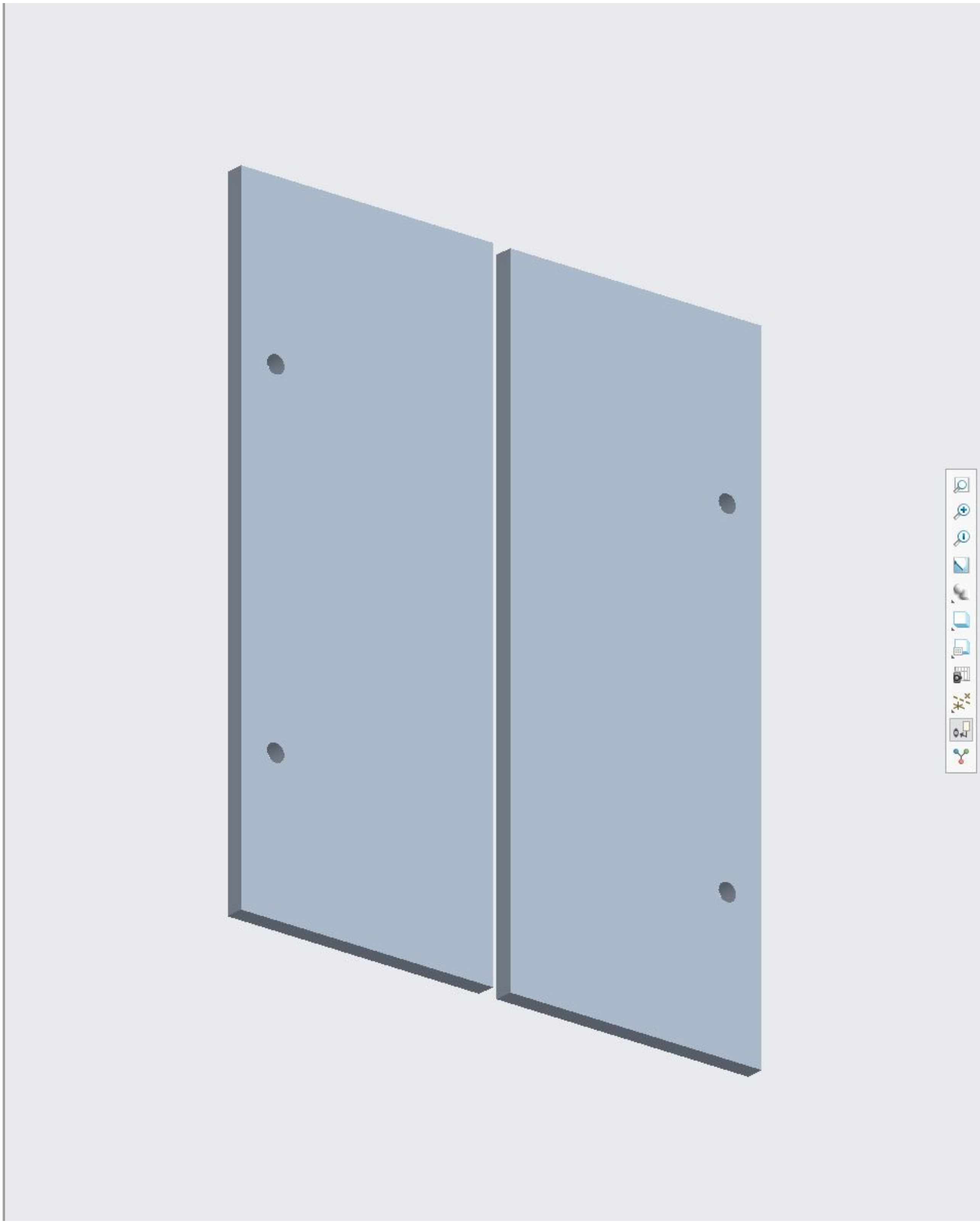

Figure A-13: Bottom Plastic

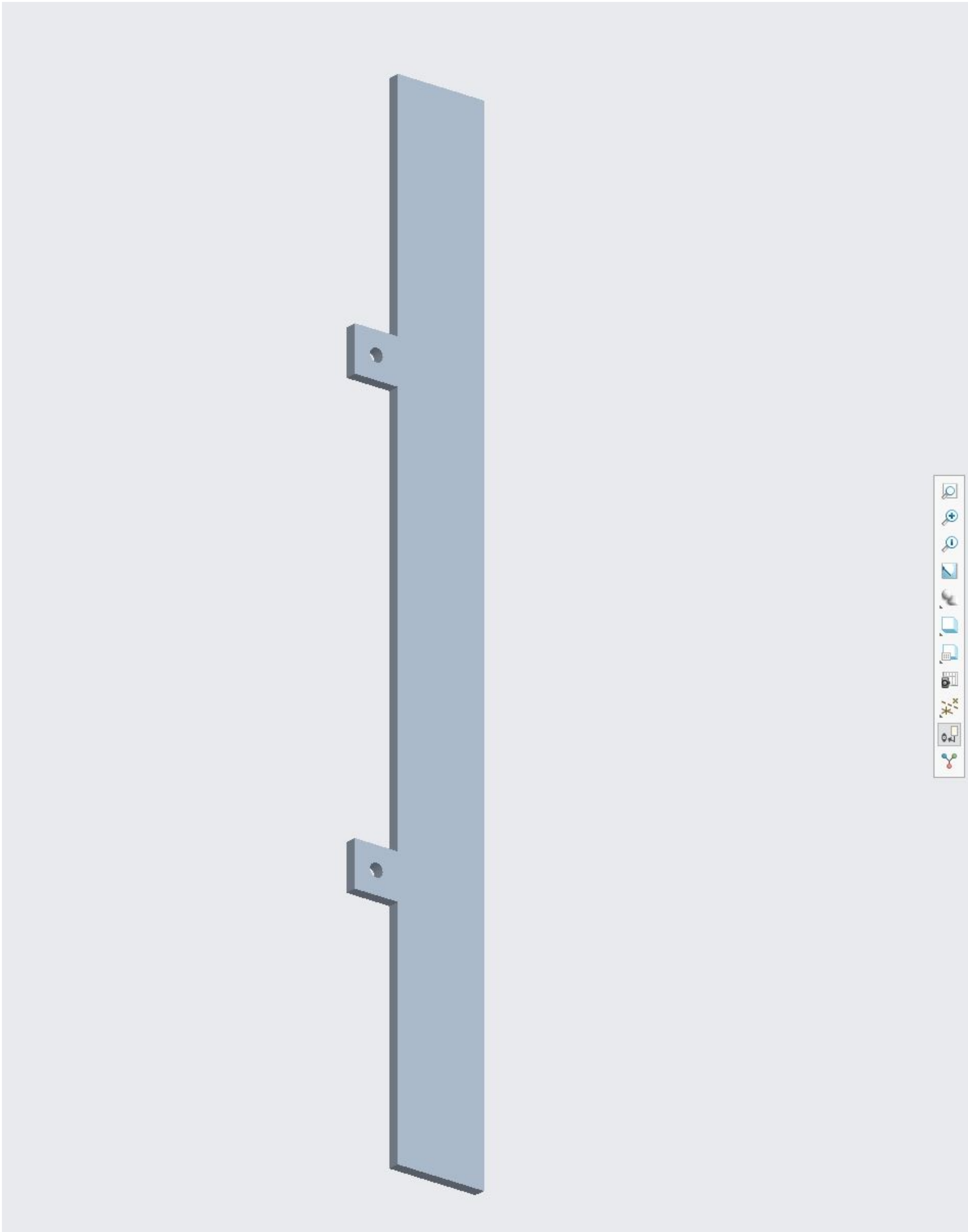

Figure A-14: Dividing Rail

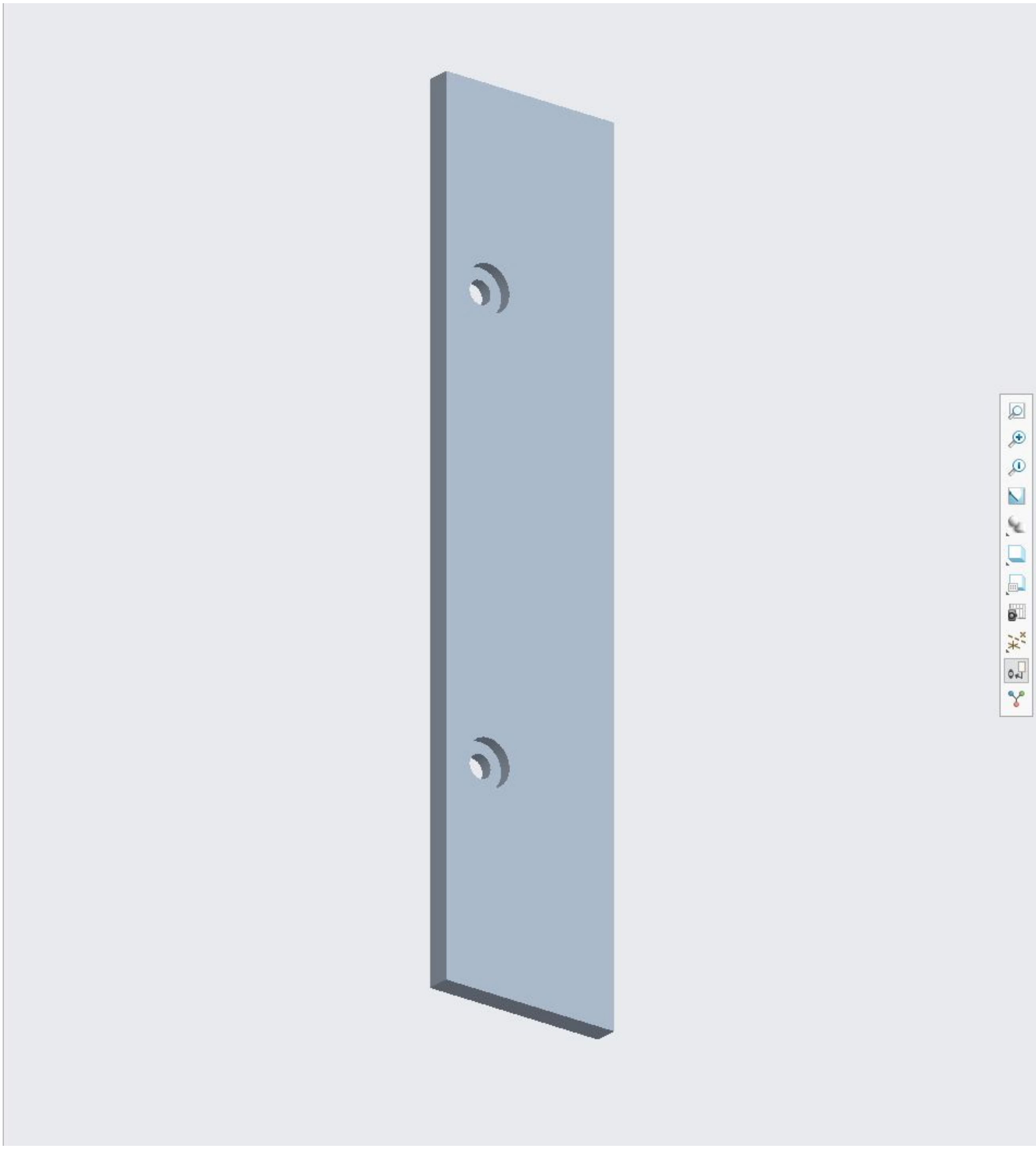

Figure A-15: Rail Plastic

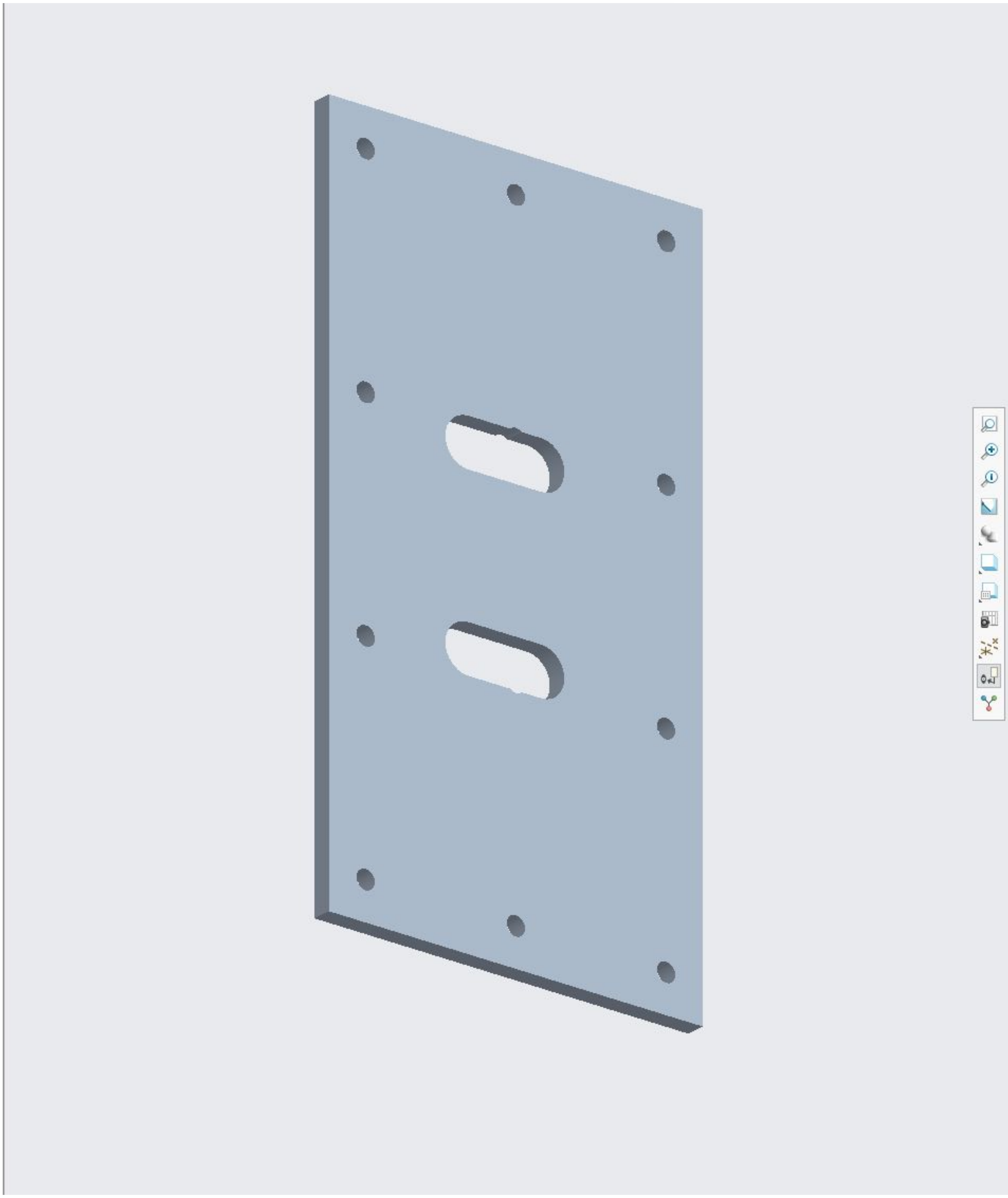

Figure A-16: Fill Cylinder Mounting Plate

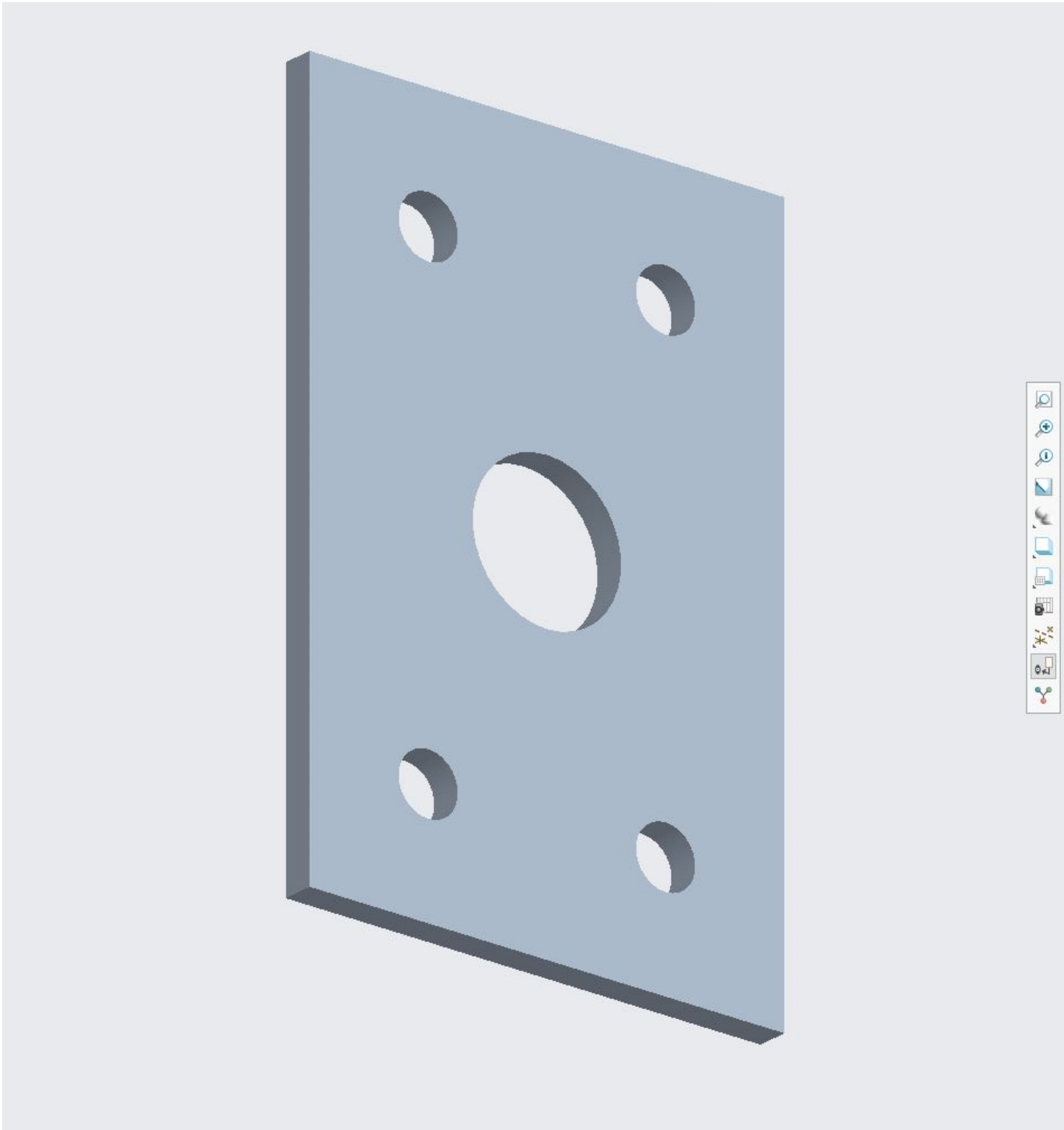

Figure A-17: Index Cylinder Mounting Plate

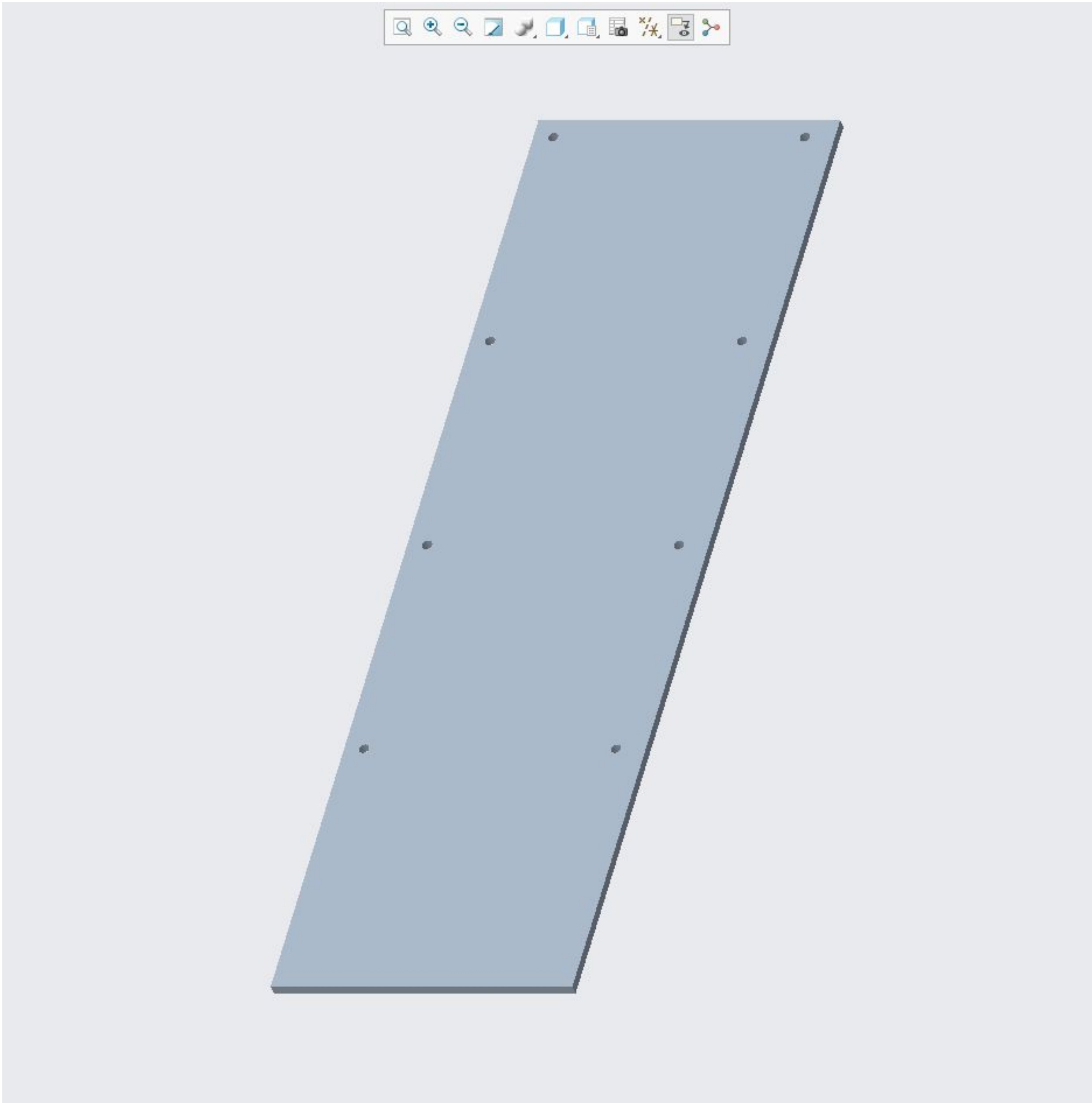

Figure A-18: Bottom Ramp Plastic

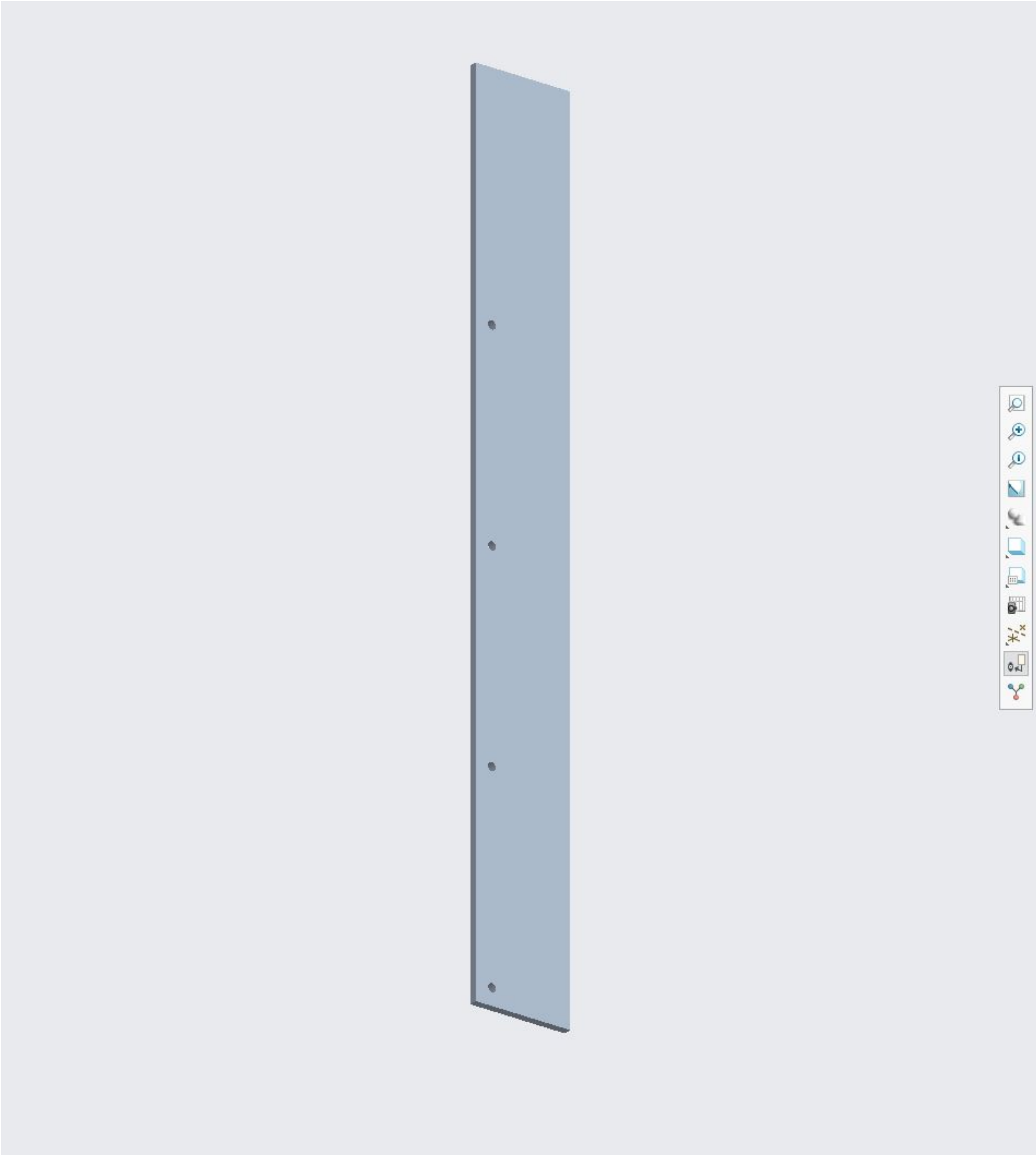

Figure A-18: Ramp Rail Plastic

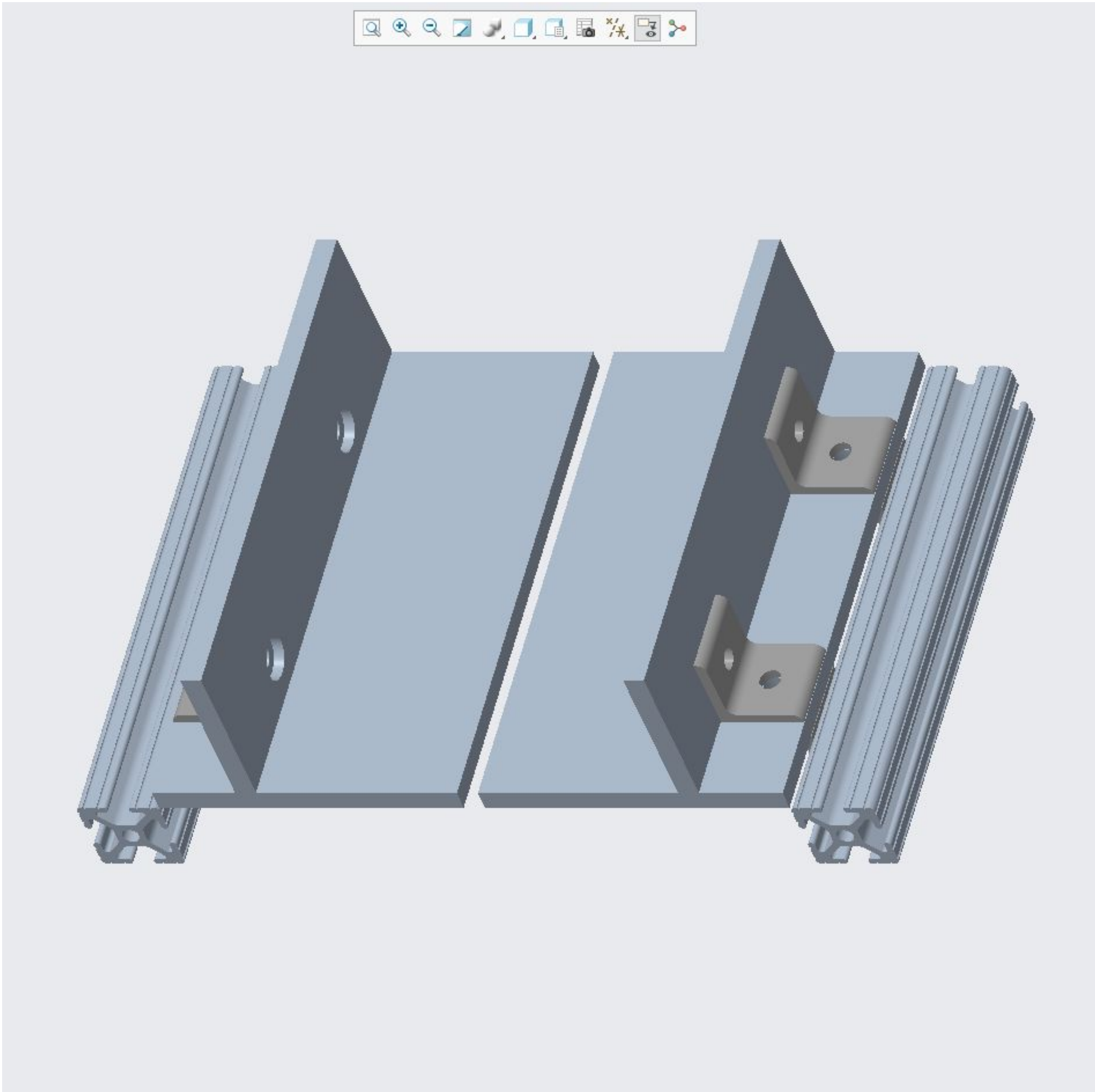

Figure A-19: Rough Constructed Framing

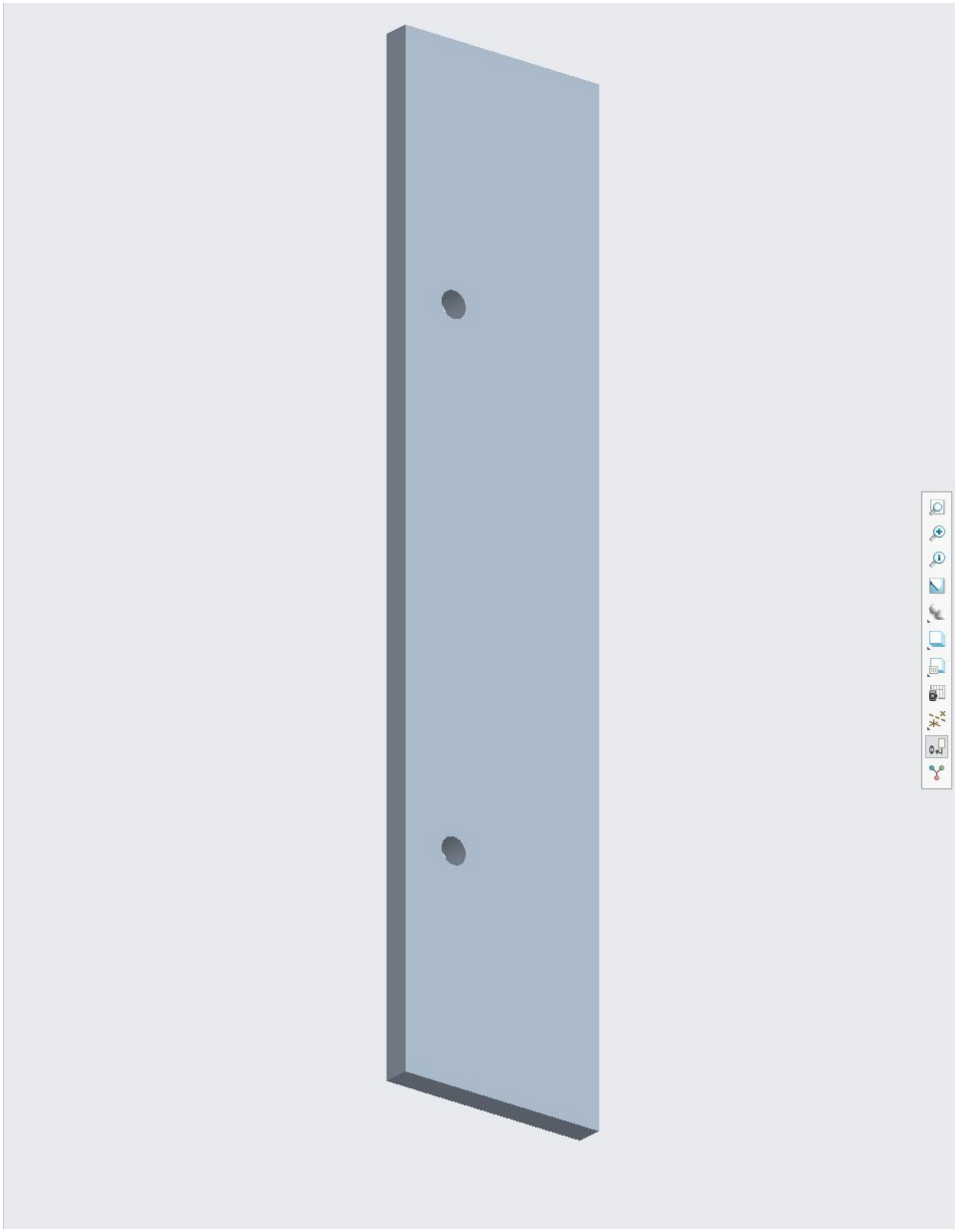

Figure A-20: Plastic Rail

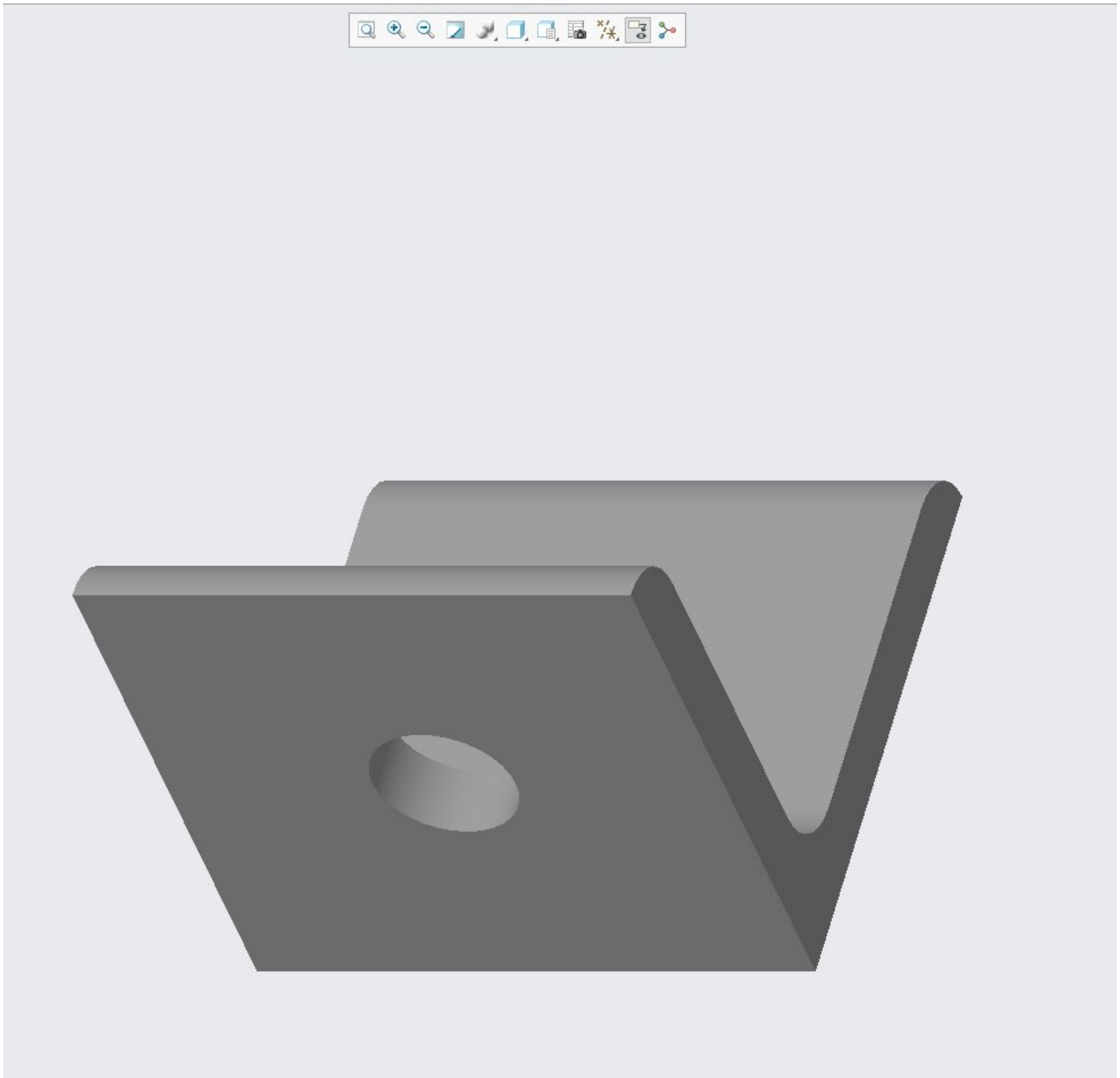

Figure A-21: Bracket

Merlin Portable Bottling Line

Source: Meheen Manufacturing Inc.

The Merlin is a fully automated filling and crowning machine with bottle pre-evacuation and long filling tubes for low air pick-up.

Introducing the Merlin portable bottling line. The Merlin is a fully automated filling and crowning machine with bottle pre-evacuation and long filling tubes for low air pick-up.

Merlin features new 3-tube filling technology controlled by computer which greatly increase speed. This machine can produce up to 1,600 bottles per hour with only 4 filling heads. Merlin is also available in a 6-head version producing up to 2,400 bottles per hour. The big news with Merlin is the price and options available. All Merlin bottling machines can be outfitted with an after crowning bottle rinsing option and coming soon a touch screen operator interface. Contact Meheen Manufacturing for more information or visit our website. Meheen Manufacturing Inc., 325 N. Oregon Ave., Pasco, WA 99301. Tel: 509-547-7029; Fax: 509-547-0939.

MEHEEN

CRAFTED IN THE USA

2016 US & Canada Direct Meheen Equipment Price List

Product	Description	Approximate Output (range)	List Pricing (US Dollar)
Standard Equipment			
M6L	6 Head Filler & Labeler system	30-40bpm	$85,250
M6	6 Head Filler-only	30-40bpm	$68,750
M2L	2 Head Filler & Labeler system	10-12bpm	$57,000
M2	2 Head Filler-only	10-12bpm	$39,000
ML	Standalone Labeler	40-50bpm	$20,500
Tank Manager	Carbonation and Temperature control manager	2 brite tanks	$8,850
Special Order Equipment*			
M4L	4 Head Filler & Labeler system	15-20bpm	$85,250
M4	4 Head Filler-only	15-20bpm	$68,750
Optional Add-ons			
Inkjet printer (Labeler)	Anser U2 pro		$3,250
Adjustable Leveling Feet	(Replaces standard caster)		$200
Crown Big Hopper	Large capacity crown holder		$700
Crowner Shut-off	Code-base override		$750

*[illegible]

For Meheen Parts & Services visit - www.meheen.com/parts

www.meheen.com PH: 509-547-7029