Arcade

Machine

Project

Overall

Project

Documentation

Arcade Machine Project

Part 1

Finding The Best Materials

**The Brains**

So probably one of the hardest questions I had to answer when it came to deciding on materials was deciding what the “brains” of the machine would be. When I first started this project, I knew that I either wanted to use a laptop/desktop or a Raspberry Pi, but I wasn’t 100% sure which one to go with at the time. I also knew I wanted the arcade machine to be capable of running games made on Unity so I could make my own games for it. So, to start I made a list of requirements for the “brains” to help decide what would work best.

1. Must run unity games
2. Windows 10 or Linux
3. “decent” GPU
4. “decent” processor
5. 4+ USB ports
6. Compact and light
7. Not to expensive
8. 8gb of ram

The two big things to note here are a “decent” GPU and processor. After doing some initial research it became extremely obvious that a Raspberry Pi would not be a good fit for this project. The major reason behind deciding to cut the Pi (pun intended) was due to the lack of a decent GPU stacked on top of the fact that it is currently not capable of using an external GPU. So, with the Pi out of the picture that left me with trying to find a mini pc.

As far a finding the mini pc goes, I got kind of lucky in that aspect. After doing some general searching on Amazon for a cheap mini gaming pc it became apparent these pcs were not that cheap. The lowest costing one that was even remotely worth it was around 500$. I knew this was going to be the most expensive part, but I was not looking to spend 500$. After having no luck on Amazon, I chose to start looking around on the Facebook Marketplace. On there I got extremely lucky finding a mini Alienware gaming pc that came with 8gb of ram and an i3 processor for 200$. So, I messaged the seller and set up a deal.

**The “Brains”**

A black rectangular object on a wooden surface

Description automatically generated with low confidence

**Monitor/Speakers/Jump drive**

As far as finding a monitor, speakers, and jump drive went I was actually able to get extremely lucky again on the Marketplace finding a monitor with a built-in speaker for 25$. Although it wasn’t the fanciest or nicest monitor it was a pretty good fit for what I was looking for. As far as finding the jump drive went I assumed that I could find one at Walmart in Norfolk when I went to get the other supplies.

**Monitor + Speakers** **Jump Drive**

A picture containing text, electronics, display, monitor

Description automatically generated A picture containing floor, indoor, wooden, wood

Description automatically generated

**Coin Acceptor**

When I originally started looking into finding a coin acceptor, I was pretty surprised to find out that setting up a coin acceptor with regular desktop is not as simple as a few plugs. After doing a fair amount of looking around I found a coin acceptor called BLEE 6 Type Coins CPU Multi Coins Acceptor Selector… try saying that six times fast. I chose this particular coin acceptor partially because of it’s highish rating on Amazon as well as the fact there was a semi decent tutorial available showing how to get this device working… somewhat. However, one of the major downsides to this was it would require additional work like soldering and creating custom PCB parts to make this work. But given my options this seemed like the most viable solution.

**A picture containing indoor

Description automatically generated A picture containing text, floor, indoor

Description automatically generated**

**Soldering Tools**

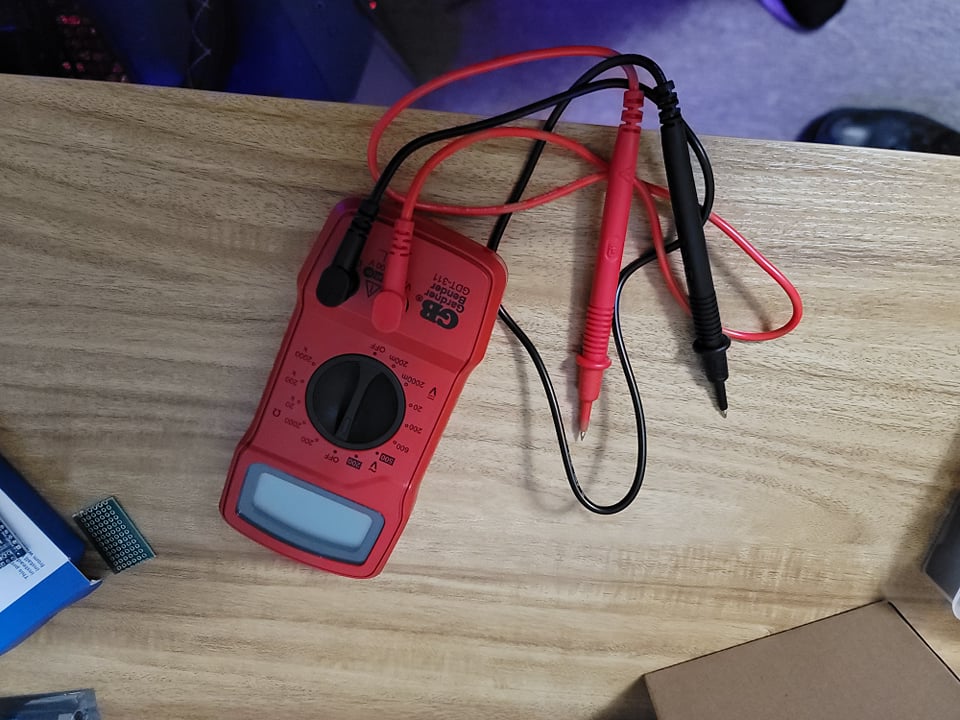
As mentioned above in the previous paragraphs I knew based on my research that I would likely have to do some soldering at some point during this project. Given I’ve never even so much as picked up a soldering tool I ended up having to order this tool set from Amazon. As far as the actual set is considered it came with a variety of different tools including the actual soldering iron , desoldering pump, 5 soldering tips, tin wire tube, soldering iron stand, tweezers, wire stripper cutter, and 2 electronic wire.

A picture containing text, indoor, floor, tool

Description automatically generated

**Multimeter**

So this particular tool I actually ended up purchasing from Bomgaars due to some trouble shooting issues I was running into in the middle of the project. As mentioned above I have very limited experience with soldering and there for when soldering I wasn’t 100% if I was even close to doing any of it right. In this case the multimeter helped me quite a bit because when I began running into bugs and issues with getting all of the components working



Crimping Tools

In the end I didn’t end up actually needing this particular tool but when first looking at the coin acceptor I originally had a plan to use the crimping tool to crimp the wires for the coin acceptor. After further inspection it became apparent I could just replace the wires given to me with the coin acceptor with regular dupont wires without the need for crimping which made things a tad bit easier overall. The toolset though was still an interesting buy overall given it came with quite a few different parts for crimping both female and male ends as well as extra wires.

A picture containing text, floor, indoor, items

Description automatically generated

**Glu gun**

This is another tool I ended up buying thinking I would need it but again found out otherwise to an extent. Originally, I had the intentions of gluing the custom PCB parts to the box so they wouldn’t move when I used them. How ever I decided not to in the end to avoid possible damage to the custom PCB parts. This mostly due to the extreme pain it was to get soldering part done properly and I have no desire to attempt more soldering until after the project is over. The glue gun itself is a basic tool with a stick of glue that when heated up allows you to apply glue to any desired surface.

**Resistors**

Part of this project ended up requiring 10k resistors for the coin acceptor / display part. In this case though, there is fair chance that the PCB part that has the resistor won’t end up making the final cut given the issues with display not working. As far as the actual purpose of the resistor it’s pretty much in the name. The resistor is designed to resist electricity or control the flow of the current to avoid allowing too much energy getting to a specific component and causing damage.



**Dupont Wires**

Dupont wires are wires with M-M, F-M, and F-F connectors that allow you the ability to hookup things like micro controllers and other devices to work together as well as pass current to said devices. For this project I ended up buying all three variations and ended up using them for wiring the arcade machine components

Text, letter, whiteboard

Description automatically generated

**Mini Arduino**

So, I originally bought this with the intentions of using but things did not go as planned at all. The Mini Arduino is essentially a smaller more compact version of the Mega 2560 Elegoo microcontroller. The major difference is that the Arduino has male DuPont connectors that require soldering of each pin. I did attempt to complete this process but unfortunately it did not go as planned. I ended up completing the soldering process but when I went to connect a wire the board, I managed to separate the connector at the point of soldering. Given that I was frustrated at this point and already worked quite a few hours on all of this I made the decision to grab the Mega 2560 from my other kit and see if that worked which in the end it did so I ended up deciding to buy one of those as a replacement.

Text

Description automatically generated

**Elegoo Mega 2560**

As explained above the Mega 2560 is simply a micro controller that allows me to upload and run code and get outputs based on the connected components. I ended up picking the Mega 2560 because of the failure of the Mini Arduino as well as the fact that I was already semi familiar with this device and it’s ability to work for what I needed.



**Display For Coin**

The original plan for this project was also to include a display allowing the user of the machine to see how many coins they had inserted. Unfortunately, it ended up not working out as planned and for what ever reason failed to display the amounts as desired. At the end of the day, I was still able to get the coin acceptor functioning with out the display and decided that instead I would simply display the amount on the screen instead. The display I picked was the HiLetgo 4-Digit Tube LED Segment Display. There really isn’t much to say about this display. It’s has 4 connections and the ability to display 4 separate numbers. Unfortunately, it didn’t work for my need and will be put on hold / scrapped for the time being.

Text, letter

Description automatically generated

**Arcade Buttons & Joysticks**

So, for the buttons and joystick I ended up picking up two EG Starts 1 Player Joystick and Arcade Button kits. I chose this device partially because I just liked the old retro style joystick look that it had. The actual setup for this device was also fairly simple given the amount of buttons and wires. The one thing I was not so fond of though was the fact that the joystick was a button-based joystick instead of analog. Given that the games I have planned are not to overly complicated this shouldn’t be a major issue, but I was not extremely happy to find this out after receiving it in the mail. Other than that, though it was overall a good kit.

A group of plastic bags on a table

Description automatically generated with low confidence

**PCB board kit**

So, throughout part 1 I mentioned custom PCB components a few times but never really explained the PCB board itself. The PCB board material I used to create the custom components is essentially a laminated sandwich structure board of conductive and insulating layers. The boards actually come in fairly large boards that required me to use tin snips to separate the board the make it smaller and better for desired use.

A picture containing text, plastic

Description automatically generated

**USB Extender**

I figured this was at least worth mentioning. The mini-pc I picked up turned out to have two failing USB ports in the front of the device. So to fix this issue I ended up getting a USB extender that allowed me to access the capability of all 4 ports with out the need to rip open the computer itself.

**A picture containing table, floor, indoor, wooden

Description automatically generated**

Part 2

Gathering The Best Materials

Although this particular topic isn’t super important I figured I’d share where I got each component in a what will hopefully be a fairly short section. As far as the mini-pc / speaker, monitor, and jump drive were concerned these were probably the most time consuming parts to retrieve. Both the mini-pc/speakers and monitor as well as the jump drive all required me to make a trip to Norfolk to retrieve them. The jump drive I ended up purchasing at Walmart while the mini-pc and monitor were bought second hand from private sellers on the Facebook marketplace.

Things like the wires, PCB board, crimping tools, soldering tools, micro controllers, and other obvious parts that I couldn’t pick up at a local hardware store were bought through Amazon. Things like the multi meter, tin snips, and glue gun were bought from Bomgaars after the realization they were either needed or would come in extremely handy. Apart from that there isn’t to much to include in this section as there isn’t a-lot of explain as far as buying these went.

Part 3

Picking The Best OS

One of the bigger things I had to do a fair amount of research into was which OS would I want to run this arcade machine on. Obviously, I could have chosen the easy option which is Windows 10 given I had the most experience with it and generally know the in and outs of said OS. But one big issue I faced was the limitation of the mini-pc I had picked up.

The mini-pc I have has an I-3 Processor and 8gb of ram and generally isn’t the worst computer I’ve ever seen. In fact for the price, it was a pretty decent steal. Unfortunately, Windows 10 although not that resource intensive was still quite a drag even on this system. Before launching anything and general playing around with machine it was blatantly obvious that Windows was stressing the system pretty bad at times. So for that reason I decided it would likely be a good idea to do further research into a different option.

That leads me to the most obvious choice possible for this project, Linux. Linux is by far the best choice in this particular case given the need for an OS that is not extremely resource intensive as well as needing the capability of running a Unity game. In this case Linux takes care of both of those issues. The only real question at this point is which version of Linux to go with. This was also a fairly simple choice as the only two that really fit my need were either Debian or Mint. In the end I chose Mint only because of how new I was to Linux as well as it’s stability. Debian although known for its stability was not the overall most user friendly.

**Part 4**

**Building The Arcade Machine**

**Creating The Arcade Box**

In this case I’m using the word box literally. For the arcade box I essentially took a regular old box and cut a design in the box for the buttons, joystick, and coin acceptor. The joystick it self was a bit tricky getting it into place given I had to remove a metal plate off the joystick that allowed put the mechanical parts in from one side and recrew the plate on the other side of the opening.

The buttons weren’t to difficult to install either. My only real concern with the buttons is that the actual holes cut isn’t much smaller than the width of the button and screw that’s holding it in so I worry that if pressed to hard it might just go straight through the box. With that being said it does feel fairly sturdy for what it is.

The coin acceptor was arguably the worst part of this creation. The coin acceptor is oddly shaped was not the easiest par to cut a hole for or fit properly to the box. For what it is, it does work but I’m overall not a huge fan of it.

After the getting all parts positioned properly the next part was wiring. For the buttons and joystick, the wiring wasn’t to bad given all it took was running wires from each button and the joystick to a decoder that then went straight to the PC with a USB cord. The coin acceptor was a bit of a different story. It required being hooked up to a micro controller, 4 digit display, and incorporated two custom PCB parts along with a 12v socket to power it all up.

**Installing The OS**

As far as installing the OS goes it was a pretty easy process that required more sitting and waiting than anything in this case. I essentially just had to use the jump drive I had to make a bootable image of the Linux Mint and then have the mini-pc boot said image via the bios options to start the install process. Most of the work was just finding out how to get into my specific bios and making sure I wasn’t making any major mistakes in the process. Otherwise, the rest was a straight forward OS install just like any other.

**Soldering & Wiring**

One of the absolute biggest hurdles I faced when wiring up this arcade machine was my total utter lack of knowledge when it comes to soldering, wiring, and really anything that involves physical hardware in general. To top that off there are almost 0 tutorials on how to wire up a coin acceptor work with a micro controller like I did for this project. What made this even better was the only tutorial I found did not exactly do an impressive job of displaying each part in detail and at times would even arguably mislead at certain points. For ex. when hooking up the coin acceptor to the 12V power supply and getting it power on it showed originally hooking the ground the in to one pin and then in a separate part it was actually in a different area of the board. In that instance I was just lucky enough to guess where it would have made sense to put it given the micro controller ground was also in a similar spot as the coin acceptors ground needed to be as well.

As far as the soldering goes let’s just say I had no idea what I was doing, and I still barely have a clue what I’m doing. With that being said after multiple attempts and quite a bit of practice I was able to bridge a few header pins and get things set up the way they needed to be in the long run. I know when I first started working on the soldering part all I could think was what did I get my self into but after quite a few failed attempts and less than appropriate words muttered under my breath I was finally able to complete both custom PCB parts needed for this project. I also did attempt to solder a nano Arduino board with header pins as well but when wiring the board up I ended up break one of those solder points and decided to replace it with the Mega 2560.

As far as the rest of the wiring and stuff goes it wasn’t extremely difficult to figure out with the tutorial, but it also was extremely easy either. The angle and which he shot the video made it extremely hard to figure the exact parts he was plugging each wire into and the wiring diagram he used was one that involved a bread board and not the custom PCB parts he had created. But thankfully in the end I was able to get most of the project working hardware wise minus the display. After 10 hours of working on this part though I decided I could do with out the display for the time being and perhaps worry about it at a later time as it wasn’t 100% necessary.

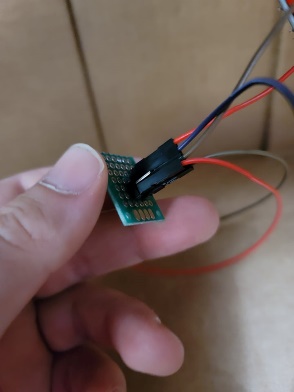
After getting everything wired up and working, I also decided to hook up the machine to my PC to see if I was able to get a communication through micro controller to the Arduino IDE and unity. This probably took a good two hours in it self of just testing and playing around with different code but I’ll save the in depth explanation for a slightly later date as well given this has more to do with the software side then the actual building side.

A picture containing electronics, adapter

Description automatically generatedA picture containing person

Description automatically generatedA picture containing electronics

Description automatically generated

A picture containing person, hand

Description automatically generatedA picture containing floor, indoor

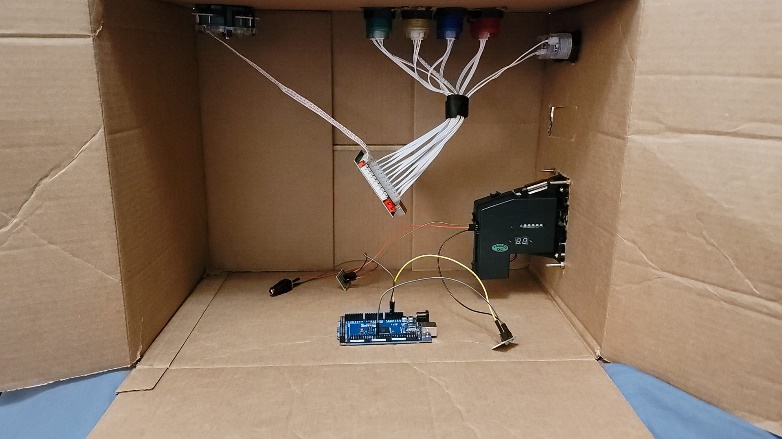
Description automatically generatedA picture containing person, indoor, hand, holding

Description automatically generated

**Rewiring**

This wasn’t really planned, but I ended up getting annoyed by the wiring and how messy it was, so I chose to rewire the project just a bit to clean up all the extra unneeded wires for the display that doesn’t work as well as some daisy chained wires that I created after breaking the nano Arduino. So now it only has the wires it absolutely needs and looks a-lot nicer overall.

A picture containing wall, floor, indoor

Description automatically generated

**Part 5**

**Creating The Arcade Game**

**What The Game Was About**

So as far as the actual game goes, I chose to create what I believed to be a fairly basic arcade style game that required very minimal code and effort overall. The general idea of the game was to mimic a game called flappy birds which at one point was an extremely popular mobile game. In this case though I did make some slight alterations on of which included giving the player the ability to both go up and down as well as slow the block down and speed the block up. Note however that the player can not completely stop the block from moving. The block itself is in a state of always moving forward into the oncoming obstacles. A few other interesting notes are that the levels are randomly generated from premade obstacles, the levels get progressively harder, you can get points for both staying alive and completing the levels, and like flappy birds losing is when you hit the block.

**Making The Player Move**

One of the more interesting parts of this project was figuring out how I was going to get player to move using the arcade buttons and joysticks. Part of the process ended up involving getting a library specifically designed to get inputs from things like game controllers and so on but also ended up working with the arcade box I created as well. Below is the code required to get my arcade box working as well as the controller mapping that works along side this code. The code itself is fairly self explantory. Vector3 is in reference to the current location on the XYZ grid of the game and the variables above control how fast the player moves based on the buttons pressed. The control layout if compared to lines 19-22 you will see a-lot similarties in the code vs the layout it self. Overall this part wasn’t to complicated and only took a little research.

**Text

Description automatically generated**

Graphical user interface, text, application, email

Description automatically generated

**Making The Coin Acceptor Work**

This was probably my favorite part of the project because when I was finally able to get Unity to recognize the quarter being inserted into the coin acceptor, I felt like I had accomplished the main goal of this project.

The first image below shows the code that is uploaded to the micro controller to interact with the coin acceptor itself. The coin acceptor is setup to recognize when a quarter is inserted and send a certain number of impulses based on that. In this case the impulses expected is 1. When the micro controller receives that impulse, it then prints “1” to the serial monitor. When testing you would normally use the serial monitor provided by the Arduino IDE but in the current circumstance the serial monitor in this case is the game itself.

The way this works by the code utilized in the second picture as well as the third picture demonstrating how it can be used. The second pictures code is part of the Unity IDE / game. When the game is running it opens a serial monitor on COM 5 which is where the micro controller is communicating to and reads anything the micro controller outputs. Line 30 is specifically looking for the output “1” and when it receives that out put adds .25 to the balance and updates the text on screen. This is more a proof of concept proving that it is more then capable of becoming a working arcade machine and not so much that it is currently a working arcade machine.

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generatedGraphical user interface, text

Description automatically generated

**The Rest Of The Code**

So, all though this game is a pretty simple game there is still quite a decent bit of code I ended up writing. Now given my current time constraints as well as the fact that this Practicum is more based on the hardware side of things than the coding side of things, I decided to include all of the non-hardware code in this section and only give a brief explanation to each part.

So, to start the Camera Code essentially gives the game a 2d side scrolling feel by making the camera follow the player on the X axis as the cube/player moves forward. The offset, offsets the camera so it’s not directly on top of the player but instead a few grid numbers back on the Z axis.

The second code example is of the Level Manager code which essentially does what it sounds like it does. It’s kind of the “Main” of the level and helps keep everything intertwined like the music manager, level generator, and tracking of player details like scores and levels. It in itself has very little logic but is meant to allow easier communication and combined effort through the rest of the code.

The next code is the Music Manager code which essentially randomly picks one of 10 possible songs to play and then waits for that song to end before playing another song. Apart from that it has no other purpose.

Player Details manages the players details like score, current level, and anything directly tied to the player. The other thing it does is give bonus points when each level is completed.

The level generator which also monitors when the player completes each level is arguably the most convoluted code. The code essentially generates the level based off the player’s current level. It uses pre created obstacles called prefabs and places them in the world at the beginning of each level. Each level is randomly generated and becomes progressively more difficult by including new obstacles and decreasing the space between each obstacle. There is also some animation work involved as spinning obstacles and obstacles that move up and down. Although those are spawned by the level generator the animations are not tied into the level generator itself.

**Camera Code**

**Text

Description automatically generated**

**Level Manager Code**Text

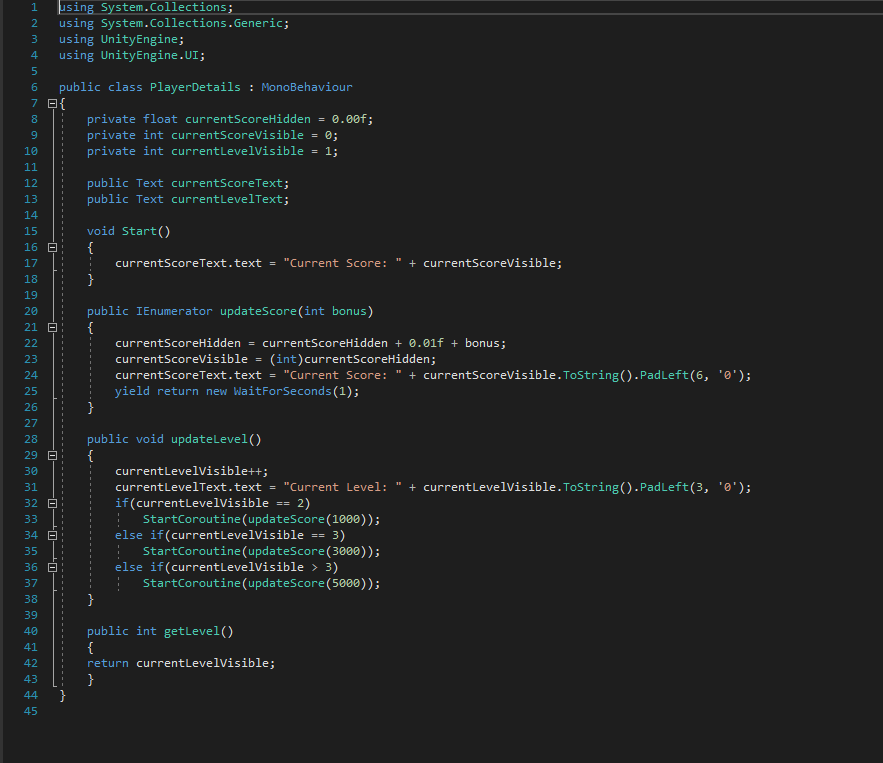
Description automatically generated

**Music Manager**

Text

Description automatically generated

**Player Details**



**Level Generator**

Text

Description automatically generated

**Randomly Generated Level**

**A picture containing graphical user interface

Description automatically generated**

**Part 6**

**Closing Statements**

As far as this overall project goes it was a surprisingly fun and challenging project for the most part. When I originally started this project, I was a bit afraid that it was going to be a-lot simpler and easier than I first imagined but after really getting into some of the more complicated parts like making custom PCB parts I quickly discovered things like soldering for instance were not as easy as I first imagined. At the end of it all I am mostly just happy I managed to get the parts I needed working in the first place.

As far as the game design part went quite a bit of that was sped up dramatically by my previous knowledge in Unity but not to the point where I didn’t run into issues there as well. Although I do have some experience with Unity this was my first time working with it in almost 2 years. So, although it did help, I was still stuck looking up quite a bit of “How do I” google searches for unity.

The only other thing I wanted to note was as far as things like the game background and music goes those were all acquired through the Unity Asset Store and not works of my own. Other than that I think this project although not a complete arcade machine is what I set out to create which was a “proof of concept” showing that what I wanted to create was possible.

Additional

Time Sheet Documentation

Date: 10/6/2021  
Time: 9:00pm – 12:00am (3hrs)  
Task: finding the best materials for the project

Full Description

If I’m being completely honest, I probably spent over 3 hours on this particular part of the project researching and finding parts on the Facebook Marketplace. Most of the time was spent looking into whether a Raspberry Pi would be a good fit and for the project. (It wouldn’t) and what would work as replacement. I also dealt with two people who were arguably the worst people to deal with when buying stuff because one couldn’t respond / tried changing our meet up time and the other wanted a deposit ahead of time. Overall a good reminder why I despise selling and buying things on the marketplace.

Date: 10/8/2021  
Time: 2:30pm – 5:30pm (3hrs)  
Task: Gathering materials (monitor, pc, jumpdrive)

Full Description

Most of this time was spent driving to Norfolk, meeting the people in Norfolk to gather said supplies, and going to Walmart to get jump drive as well as driving back to Wayne. I also spent close to an hour inspecting the mini Alienware pc I was buying before deciding to take it given I didn’t want to risk it not being what I actually needed for this project.

Date: 10/10/2021  
Time: 2:00pm – 5:00pm (3hrs)  
Task: Researching the best OS

Full Description

This time was spent looking into different possibilities for an OS for the arcade machine. The only real worry I had with Linux was making sure that Unity games would work on this OS and specifically on Mint. I also did take Windows 10 into consideration but given the system isn’t very powerful chose Linux for obvious reasons. I also researched Debian vs Mint and other Linux options.

Date: 10/18/2021  
Time: 8:00pm – 10:00pm (2hrs)  
Task: Researching coin acceptor / figuring out a way to get it working

Full Description

The big issue with deciding on a coin acceptor was finding one that I could find enough information to get it working as well as making sure I could get it into unity. I was actually surprised how it’s not as hard as you’d think to get the output but the wiring and soldering is a different story. I made my choice based on a tutorial I felt I could likely follow to hookup the coin operator.

Date: 10/19/2021  
Time: 11:00pm – 12:00am (1hrs)  
Task: updating documentaiton: Timesheet Explanations (old documentation)

Full Description

This was the documentation I started for the time sheet before deciding to go a different way. So this was just time I wasted on the old documentation that I will include for the purpose of evidence/proof.

Date: 10/24/2021  
Time: 12:00pm – 1:00pm (1hrs)  
Task: updating documentaiton: Timesheet Explanations

Full Description

This was the documentation I started for the time sheet before deciding to go a different way. So this was just time I wasted on the old documentation that I will include for the purpose of evidence/proof.

Date: 10/24/2021  
Time: 9:00pm – 10:00pm (1hrs)  
Task: ordered additional supplies

Full Description

I actually almost forgot to include this, but I spent a good hour ordering additional parts for the soldering and other things like that on Amazon on the 24th. I’m not sure of the exact time but I know it was the 24th and it took around an hour to find and order everything I needed. I only know it’s the 24th because of my actual order history.

Date: 10/24/2021  
Time: 10:00pm – 11:00pm (1hrs)  
Task: updating documentaiton: Timesheet Explanations

Full Description

This was the documentation I started for the time sheet before deciding to go a different way. So this was just time I wasted on the old documentation that I will include for the purpose of evidence/proof.

Date: 10/24/2021  
Time: 11:00pm – 12:00am (1hrs)  
Task: Installing Linux Mint

Full Description

This was just time spent getting a bootable Linux mint image on a jump drive as well as installing Mint. The process was smooth painless.

Date: 10/25/2021  
Time: 12:00am– 2:00am (2hrs)  
Task: set up and run basic unity program

Full Description

After installing Linux Mint I wanted to be 100% sure that a Unity game would run on this OS so I made a really basic game that was just a cube slowly moving away from the camera to see if it worked as well as how well it ran. Thankfully I had no issues for the most part minus figuring out how to launch it at first.

Date: 10/26/2021  
Time: 5:00pm – 6:00pm (1hrs)  
Task: purchase additional supplies from bomgaars

Full Description

I ended up going to Bomgaars to buy tin snips to cut the PCB boards, a multimeter to test my soldering and general testing of other things, and a glue gun which I ended up not needing.

Date: 10/26/2021  
Time: 10:00pm – 1:00am (3hrs)  
Task: Set up basic arcade "box" / test with unity

Full Description

Basically I just cut a box up using a box cutter put buttons and joystick as well as coin acceptor and other stuff in the box and wired up the buttons and joy stick / tested it with Unity to see if it worked.

Date: 10/28/2021  
Time: 6:00pm – 2:00am (8hrs)  
Task: soldering / hooking up coin machine / setting up coin machine

Full Description

I spent close to 10hrs over all soldering, trouble shooting, testing, and just generally trying to get the coin acceptor hooked up and working as needed. By this point and time I did have all the soldering and wiring done for the most part but still had some work to do the next day.

Date: 10/29/2021  
Time: 12:00pm – 2:00pm (2hrs)  
Task: finished getting coin acceptor hooked up and confirmed code in arduino worked / got an ouput into the unity console

Full Description

I ended up spending an additional 2 hours uploading the code to the micro controller and general testing to find the best way to get output from the coin acceptor to unity it self.

Date: 11/4/2021  
Time: 10:00pm – 12:00am (3hrs)  
Task: updating documentaiton: Part 1

Full Description

Spent a good 3 hours taking pictures as well as writing in depth documentation on the Arcade Machine project for part 1 as well as covering all the components used.

Date: 11/5/2021  
Time: 6:00pm – 9:00pm (3hrs)  
Task: updating documentation: Part 1-4

Full Description

Finished what I had left for Part 1 and completed all of 2-3 and most of part 4 as well.

Date: 11/7/2021  
Time: 12:00pm – 2:00pm (2hrs)  
Task: updating documentation of part 4 + Rewiring

Full Description

Completed part 4 and did some rewiring to clean up the project so it doesn’t look like a rats nest anymore.

Date: 11/7/2021  
Time: 2:00pm – 3:00pm (1hrs)  
Task: timesheet documentation

Full Description

Basically, redid the entire timesheet documentation. Everything above this point was done on 10/7/2021 from 2:00pm – 3:00pm.

Date: 11/7/2021  
Time: 3:00pm – 5:00pm (2hrs)  
Task: Making a basic unity UI for the coin acceptor / getting it to display the current balance for the coin / basic wiring issues

Full Description

I did run into a small wiring issue that I caused my self after removing the display and cleaning up the general wiring. But thankfully it was a quick fix that only took 15ish minutes to find and fix. Apart from that I got the UI text to update on my Unity game when a coin was inserted to the coin acceptor and general clean up of the code.

Date: 11/13/2021  
Time: 10:00am – 4:00pm (6hrs)  
Task: Working On "That One Block Game"

Full Description

I spent around 6 hour working on getting the start of a randomly generated level, creating obstacles, working on the music manager code, getting the block moving from keyboard inputs for testing and general coding.

Date: 11/14/2021  
Time: 1:00pm – 6:00pm (5hrs)  
Task: Finished developing arcade game and setting up with arcade buttons

Full Description

The last 5 hours of development was reorganizing code, getting the game to work with the arcade buttons/joystick itself, making a better level generation, and general code / quality improvements to the game.

Date: 11/21/2021  
Time: 3:00pm – 6:00pm (3hrs)  
Task: Finishing up documentation and closing statements / wrapping things up

Full Description

Spent the last 3 hours finishing up Part 5 & 6 of the documentation as well as updating the time sheet documentation.