2019

MarksButton

Josiah Bull | Blue Barn

10/3/2019

Product Outcome

MarksButton was a big success, people at the office use it at least 2-3 times a week and overall it more than accomplished all goals set out for it. The 3D printed model fit all components well, with only a few minor glitches. Which were easily solved with the use of some hot glue. The namesake of the project, Mark, finds it amusing and likes the idea, so I intend to keep it around.

Time Management Outcome

This project took significantly longer than initially anticipated. I had originally aimed for this project to be a 1 week affair, but difficulties with software caused a lot of issues. I started with Node.JS and the Johnny-Five, RobotJS, and node-notifier (for Windows toast notifications). I got a working prototype, but it turns out RobotJS and node-notifier do not package well into a .exe from node.js (using pkg anyway). This led to me running this prototype using the forever library and a .bat to start it. I ended up trying to use C# but ran into issues trying to utilize a .net library I found for serial communication to Arduinos using standard firmata. Eventually I had to write my own Arduino firmware in C++, then write my own communication software in C# to interface with it. Getting toast notifications to spawn with C# was a new challenge. UWP apps support it easily, but they cannot run in the background like I require. Console C# apps seem to have almost no support for toast notifications either. I ended up using a Windows Form app which seemed to have all of the features I needed. It didn't make toast notifications as easily as on the UWP, but with the help of the Microsoft notifications library I got it done.

Unfortunately, all of this development time added 2-3 weeks to the development process, putting me well over budget on my time management.

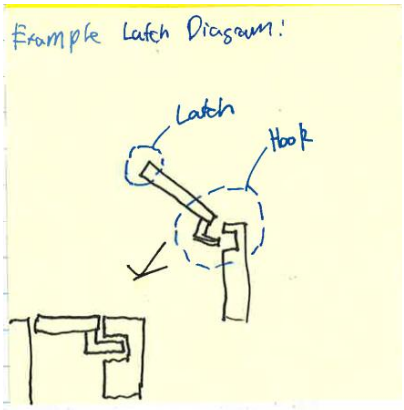
Learning/Takeaways

I learnt a lot in this project, in several different fields.

Inventor

Honestly every time I pick up Inventor, I learn something new. This was my first project designing for a 3D printer, and it went splendidly, with only a few minor hiccups:

- Walls on a 3D printed object should be at least 0.8mm thick to have any strength with them.
- 0.2mm of clearance is required between parts which are designed to slide into each other (like a male pin/female socket).
- 1mm for text indentation is surprisingly good for making the text stand out.
- I used 3mm walls for my 3D print on this project, but one could probably get away with 2mm, or even down to 1mm in places without too much loss of strength.
- 1mm thick PLA has a surprising amount of give.
- If I have a physical model available in the hand, I should always use that to measure from rather than relying on models downloaded from an online library.
- When designing a holder for a pin bolt, designing a mechanism to hold a nut usually isn't required. The plastic grips the bolt quite well and a nut is kinda overkill.
- I only locked down the button on one side of the mechanism. In the future I will put a flange on the opposing side of the lock bolt to act as a lever for the button allowing it lock down much better (see right).



Overall though, I feel the modelling side of this project went exceptionally well and I am very pleased with the result.

Programming

This was my first project dabbling in several different languages, such as C++ and C#. Overall I feel it went very well, I learned how to use a bunch of different features of different languages.

- The C# WiX packaging for .exes.
- Visual Studio and some of its features.
- The limitations of Windows Forms/UWP/C# Console Apps.
- How classes work, and instantiation etc.
- Toast notifications on windows apps.
- Simulated keyboard presses for windows apps.
- Custom audio for windows apps.
- How to have something start in the tray and stay there.

I would feel confident making a basic non-GUI C# application in the future (with a lot of help and slowly), though I don't know much about how to create a GUI yet. The initial learning for C# has been completed at this point.

I still have a lot to learn about C++, particularly as I only used it in a very limited capacity with Arduino.

Overall, I found this project very interesting to develop new learning in programming styles and languages. I still have a long way to go before I am fluent in any of these languages, but I enjoyed this project and I am looking forward to writing more C# apps in the future. 😊

Summary of Project

MarksButton had some significant challenges on the software side of things, which required a lot of working through and rewriting again and again. If I was to do it again, I would jump straight to WinForms C# rather than wasting time with Node.JS, Console C#, or UWP C#.

The CAD/hardware side of this project went splendidly, and on the back of it I am going to buy a 3D printer for future projects.

I feel this was an excellent introduction project into the hardware-software side of programming. I look forward to more advanced projects in the future.

Relevant Links

Photos and Renders

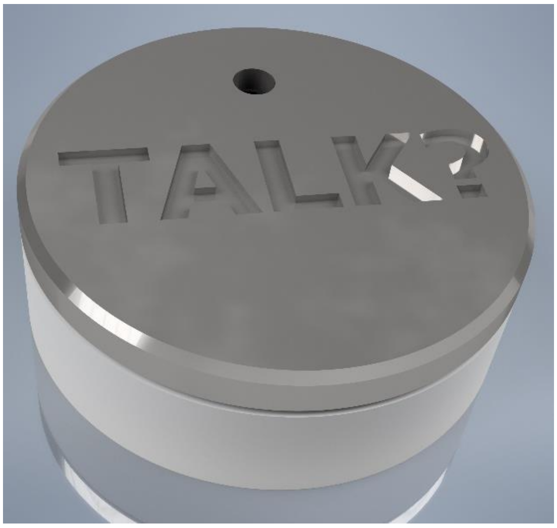
https://photos.app.goo.gl/KdRxudGr9K38g9dz9

GitHub Repo

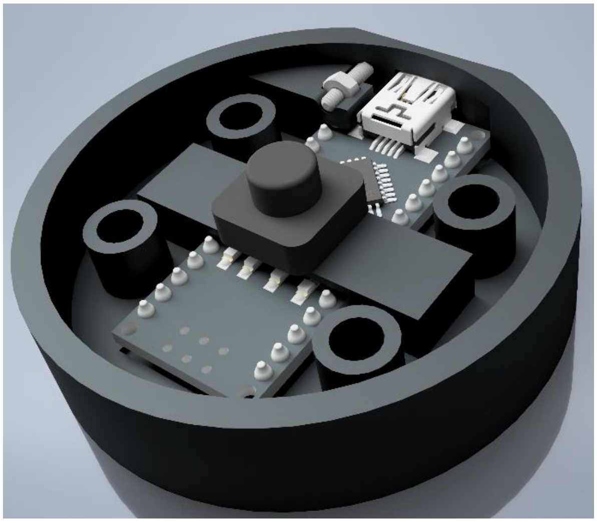
https://github.com/badtoro2/MarksButton

Photo Gallery

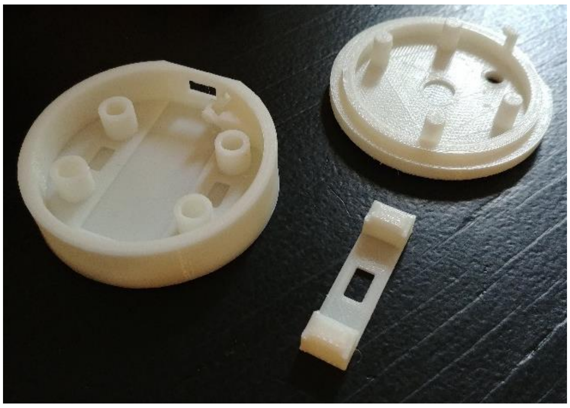
External Render (full model)



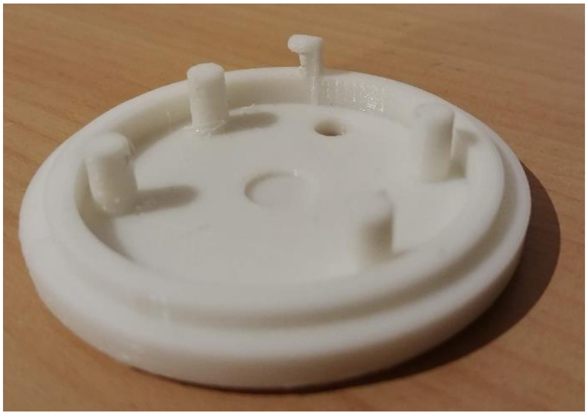
External Render (No Top Button)



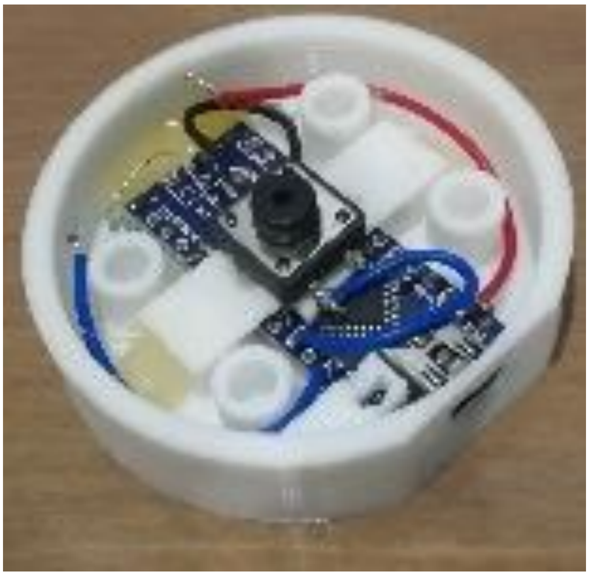
3D Printed Parts



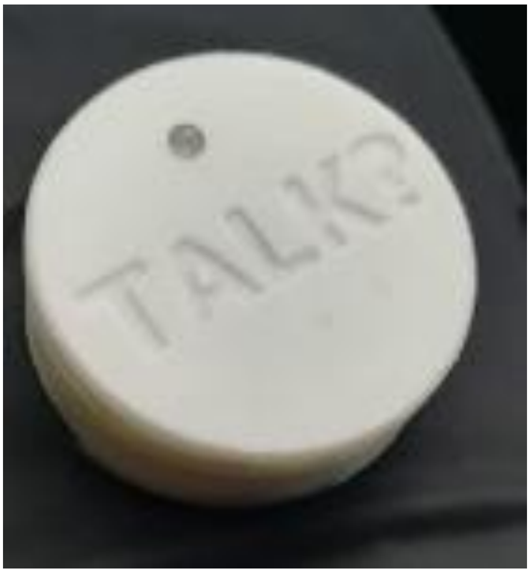
Bottom of Top Plate



Internals



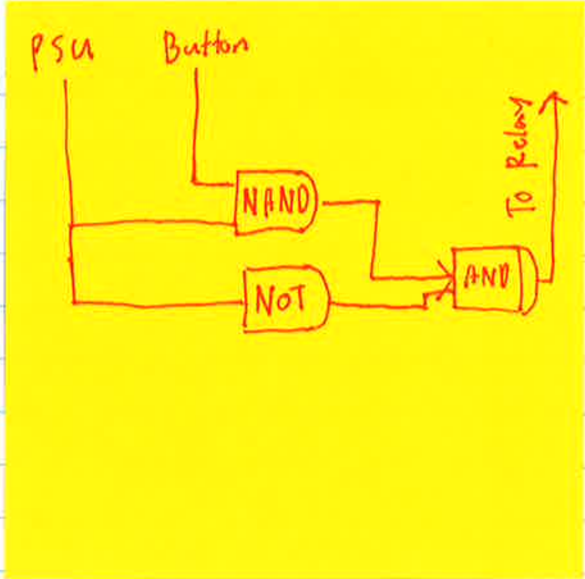
Finished Model



Original Project Drafting:

When I first conceptualised this project, it was far too complicated. This is that original conceptualisation. It involved batteries, a RPI, wireless communication, and a bunch of other stuff.

Project for Mark:



Function:

- ~~w~~ Twist top to select user
- Flip latch to choose come to me, etc
- Press down to send
- ~~LED~~

LED codes:

- Blue: Nominal
- Green: Sent!
- Flashing Green: Sending...

Chassis Design:

- Cube, $8cm^3 - 10cm^3$
- 3D chamfered corners
- Edges rotate around center,
- simple flip latch in the middle to allow switching between come here / I'm coming.
- Press down to ~~sem~~ send

- Led ring for feedback?

Power:

- small single-cell rechargeable Battery Pack
- Time how long it'll power the project
- turn on an LED when it is time to plug it back in
- turn off LED and do not turn back on after it has been plugged in for an hour.

or get an ADC module and check the voltage manually using RPI.

Revised & Simplified Project Drafting

The more appropriate project I ended up going with.

MvR - Button

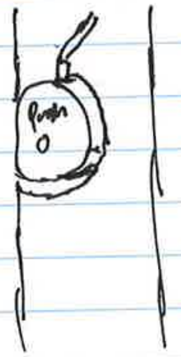
Chassis:
Two - Part [illegible], Top mounted on Switch, Bottom Base.
Leave Slot for power.

Arduino Nano interfaced with nodejs applet for notifications.

Cylindrical Design, Led indicator in middle, ~~Smo~~ Min dim possible.

internal Spring ~~for~~ ~~[illegible]~~ to push top upwards

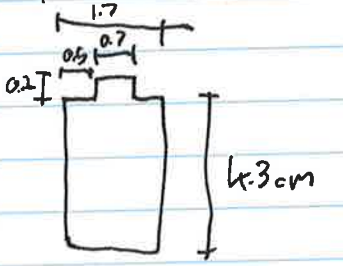
M2 screw on bottom for acess,



Top:

Todo:

CAD Model of Design
- Spring
- nuts
- Joinery


Arduino Dims:

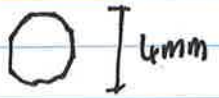
Contact George for 3D printing Quote

Source Parts
- M2 Machine screw and nut
- Spring
- Mini USB cable of appropriate Length (riser cord?)
- Button Switch
- Bright RGB LED

Further Notes & Features

Project feature checklists, small diagrams, important notes.

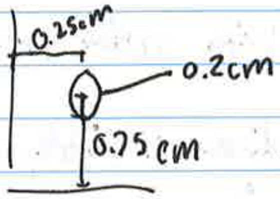
M2 Bolt:



0.23 cm Deep.

1.6mm thick

3.398



New Feature Additions:

- Turning off LED Should not send notification
- custom sound plays not as part of the toast.
- icon on .exe
- include extra files for easy installation.
- PWM Arduino LED to decrease brightness.

Further Features:

- Hover in tray Should bring up a tool tip.
- Handle disconnection of serial port.
- Wix open on Startup

Button: D7

LED: D13 or D9

Example Latch Diagram: