Automated Locker
Final Design and Build Guide

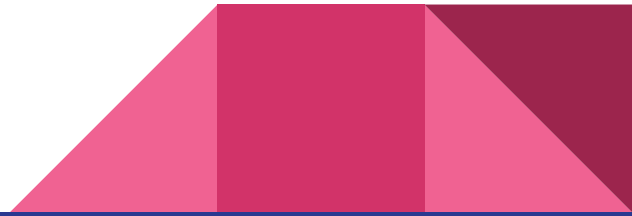
Dee Gao and Evan Brown

University at Buffalo | CSE453

Fall 2019 - Spring 2020

Disclaimer

- ➔ Part of this presentation will be addressed to the new group, which includes any mention of “you”
- ➔ We are also creating a comprehensive build guide that will be referenced at points, which will contain more detail for the incoming group
- ➔ If you have any questions, feel free to stop us at any point



Now First, A General Review of the Design Ideas...

Fundamental Design

- **RFID based unlocking system**
  - Simple to use for person with visual impairments
  - Press button to turn on system and scan RFID tag to unlock

- **Manual Override system for emergencies**
  - Manual key lock incase of emergency/Lost RFID Tag/Dead Battery
  - Staff/Teachers have keys

Requirements

- **Easily accessible for visually impaired student**
- Implementation
  - Bright colors & big distinguishable parts
  - Larger button & surface area to turn on RFID scanner
  - Sound and LED queues for unlocking/locking

Requirements 

- **Easy to troubleshoot**
- Implementation
  - Easily accessible in case there are hardware issues
  - Easy to understand instructions for troubleshooting potential future problems
  - Design should be simple

Requirements 

- **Power Efficient**
- Implementation
  - System turns on when user presses button
  - System turns off automatically in locked position after ‘x’ seconds

Requirements 

- **Exchangeable battery**
- Implementation
  - Use pre-built UL approved battery & docking station
  - Easy to switch out batteries for charging
  - Include an LED to indicate battery level

Requirements 🔑

- **Manual Override**
- Implementation
  - Provide a key to manually open the lock in case of emergency
  - Have mechanism be durable and easily maintainable
  - Must work at all times, has to be available for emergencies

Requirements 

- **(Un)Locking LED Queues**
- Implementation
  - LED to represent when system is lock/unlocked
  - **Locked**: RED
  - **Unlocked**: GREEN

Requirements

- ➔ **Unlocking SOUND Queues**
- ➔ Implementation
  - ◆ Mini speaker plays quick tone when ON button is pressed and when locker successfully unlocks
  - ◆ Helpful to have an auditory queue in addition to the visual LED

Parts List

Functional Parts & Tools

- Lock Style Solenoid 12VDC
- Door handle
- AdaFruit Push-Button Power Switch Breakout
- RFID Card Reader - Serial
- RFID 50mm Round Tag x5
- Key Lock and Key

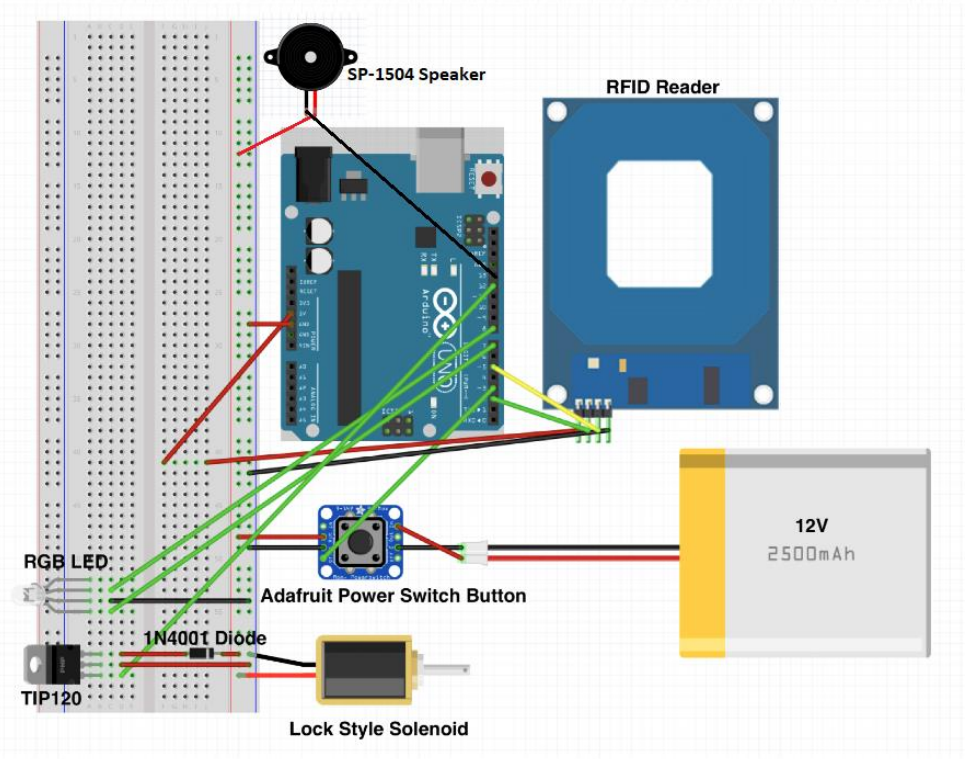
Circuit Parts

- Arduino UNO R3
- 220 ohm Resistors
- Diode Rectifier 1N4001
- TIP120 Transistor
- LED RGB Diffused
- Speaker: SP-1504

Battery Parts

- Dewalt 12V Lithium Ion Battery
- Dewalt 12V to 20V Lithium Ion Battery Charger

Circuit Design



Manual Override Re-Design

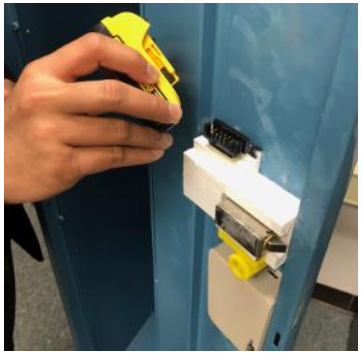
- ➔ Concerned with the longevity of the current manual override system
- ➔ Showed wear over time, with the bottom gear drooping making it harder to turn the key lock
- ➔ Potential Solution: Implement a mirrored design, so the gear would sit on top of the solenoid instead
- ➔ Freedom to experiment with other manual override designs

Things to Note

- ➔ Most things regarding the design will remain similar if not the same to the original Locker Design from 2019
- ➔ We wish we had more time to be creative with our parts, however once everything started to come together, stuff happened
- ➔ We believe we still have the capability to provide a seamless transition to the next group and give them a better and more comprehensive build guide that will fill all of the gaps and answer all of the questions that we had in the early design phase

Now for some Design Specifics

Battery



- ➔ For the Circuit to work properly, 12V must be applied to it when turned on
- ➔ This has to be done by disassembling the DeWalt Battery Charger into its bare components using a Torx t15 bit, with a long screwdriver neck to reach the screws
- ➔ Battery is then docked to the charger components, which are connected to the button
- ➔ Customer already has a charging station, so only one is needed for the build

Button

- ➔ We have both a Green Momentary Push Button, and the Adafruit Push Button
- ➔ The Adafruit Button can be connected to the Arduino, while the Momentary cannot
- ➔ We believe you may be able to combine the two, to keep the aesthetic and ease of use of the Big Green Button, while using the functionality of the Adafruit Button
- ➔ This has not been tested as we do not have access to the adafruit button at the time of making this, however we encourage exploring that option, as the other option is 3D printing one

Speaker

- ➔ The speaker is unique to our design, hence there is no previous design for it
- ➔ I was able to test the speaker at home and write code for it. I just can't test the code with the rest of the program right now, however it is very straightforward and can be edited among testing.
- ➔ Code includes an unlocking sound and a sound for when the circuit is activated upon the button press
- ➔ The speaker wires are quite small and flimsy, so we recommend striping the ends down and folding the wires over themselves to make a thicker wire and more reliable connection
- ➔ The circuit shown above has the wire colors reversed for the speaker. This is corrected in the build guide

3D Printing

- ➔ Made up of several different STL Files that hold the Locker Components
  - FrontPlate
  - KeyGear
  - LockGear
  - LockSolenoidHolder
  - LockerInterface
  - LockerInterfaceLid
  - button
  - buttonHolder
- ➔ These will all be held together with superglue and 3M Tape if needed
- ➔ It is essentially a puzzle, and should be simple to piece together
- ➔ Our customer’s favorite color is Green, so make sure to print the button and front plate with Green material!
- ➔ More details on each file will be in the build guide

3D Printing

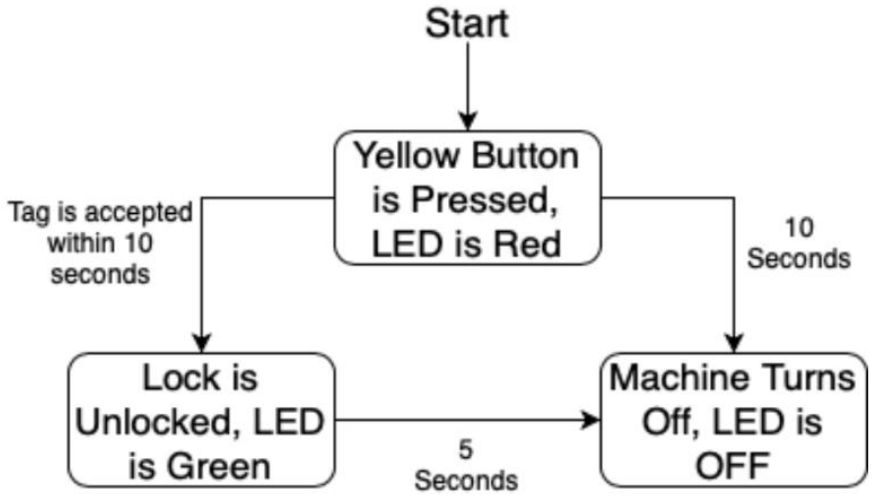
- ➔ This is one of the things we could not do in time, hence we have limited knowledge of the process
- ➔ However, we still have all of the required 3D models that just need to be printed
- ➔ Start printing immediately so that you can correct mistakes, since the circuit does not take long to build
- ➔ I will try to find relevant 3D printing tutorials to include in the build guide if you are unfamiliar, but I also recommend researching it independently before starting the build

Software

- ➔ We made little changes to the already existing code, therefore if the circuit is built properly, the code should work as intended
- ➔ There is relatively little code if you do need to change any, that consists of a setup function that initializes all of the components, a loop that checks for an RFID signal and unlocks/changes LED, and a function that reads the RFID Tag serial number
- ➔ There is also a file that is there to allow the programmer to print the RFID Tag numbers onto the serial monitor, allowing them to add the key to the main program

Software

Here is the flow of the program:



Manual Override Design

- ➔ Right now the manual override design uses the keyGear and lockGear 3D printed parts to essentially move the entire Lock Solenoid to the right to unlock it
- ➔ An actual metal key lock must be used as well that is connected to the key gear
- ➔ As mentioned before, this is a nice and simple design, it just poses a risk for wear and tear as the key gear can droop down over time, making it difficult to catch the gears and unlock it (requires user to manipulate key to get gear in right place)

Manual Override Design

- ➔ We have been boasting about improving the Manual Override Design, however our best solution so far is still trying to mirror the previous design as to try and avoid drooping of the parts over time
- ➔ Akin to the previous slide, this was going to be a hands on process with testing and brainstorming. This has made it difficult to provide solid details on its implementation
- ➔ We recommend trying both, and if the new group ends up going with the previous design, it should work fine. It may just need some adjustment in the future, and will provide a learning experience for the next build

Assembly of Parts

- ➔ The Circuit is quite simple and should work right out of the gate with the code provided
- ➔ When putting together with 3D printed parts, the LockerInterface part should hold most of the electrical components, while the battery should sit on top of the LockSolenoidHolder
- ➔ Soldering should be optional. All circuit components should be firmly connected to the breadboard and Arduino (Use a small breadboard so there is more room for components)
- ➔ Speaker has glued surfaces and is magnetic, so we recommend having it stuck directly to the inside of the locker door so the sound is not muffled too much (A little superglue wouldn’t hurt)

Assembly of Parts

- ➔ You will need to acquire the locker door to start assembling, so contact the client (Craig) as soon as you get started
- ➔ A cut will need to be made in the locker door to remove the original components and make room for the parts
- ➔ A Door Handle is also required to be screwed in, as there is no other way to open the locker given the old mechanism will be removed (We do not have this part, a quick trip to home depot should do it though)

Wrap Up

- ➔ We have learned a lot during this process, we only wish we had more of a chance to capitalize on it
- ➔ As mentioned before, we took a lot from the previous locker project, however we spent a lot of time filling in gaps that were left in previous design documents, and hope to provide the new group with something much more detailed and in depth that will make assembly a simple formality
- ➔ I am in Rochester, and will be driving back to Buffalo to deliver parts and meet with the new group come Fall. We will use this opportunity to formally transition the project in person

Please let us know if you have any questions or concerns

We wish the best to all of you and the new group!