Enhancing an Analog Pinball Experience using IoT, NodeJS and Visualizations

DGMD E-598 Digital Media Design Capstone
Frank Pizzuta - Fall 2019

Table of Contents

Abstract

In the first section of this text, the author describes the project to connect analog pinball machine events to external digital displays. The purpose of the project is to enhance and extend the pinball game experience outside of the pinball machine itself. Pinball machines have evolved over time. A survey of the approach to evolving pinball was undertaken and the shortcomings of these methods explained.

The project is broken into three phases. Phase 1 covers the detection of the firing of a pinball switch to a digital signal using an Arduino. Phase 2 constructs a Nodejs web service to listen for and react to the Arduino posts. Finally, phase 3 uses the Node server to communicate with browsers, Wi-Fi lightbulbs, and an augmented reality app. Five criteria measured project success: how often events fired, timing of the external event, placing an image on the screen, distraction to the player, and cost.

1. Project Scope

1.1 Project Title

Enhancing an Analog Pinball Experience using IoT, NodeJS and Visualizations.

Digital Media Capstone Track
CSCI E-31 Web App Development/Node.JS
CSCI E-3 Web Programming / JavaScript

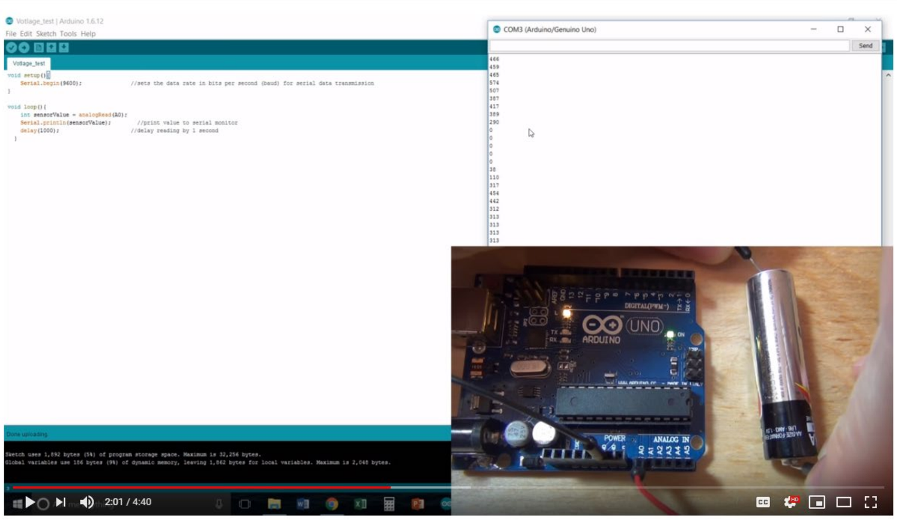
1.2. Project Goal

Older game systems are a large niche market where customers are willing to spend large amounts of money for restoration and customization. There are many older analog game systems, but for this project I will focus on pinball. Newer pinball machines, while containing some digital technology, still create essentially the same game experience as their electro-mechanical counter parts. I want to build off this established game experience. My initial efforts will be at enhancing this experience.

To achieve this, the goal of my capstone is to convert an analog signal, such as a voltage change, to digital and cause a synchronized digital event to occur via a web service. This digital event could be a video played on a screen, effect the environment around the machine such as turning lights on and off, or video rendered inside of an Augmented Reality setup.

1.3. Learning Goals

1. Use an Arduino, Raspberry Pi, or other IoT device to detect a voltage drop.


Votlage_test | Arduino 1.6.12
File Edit Sketch Tools Help
Votlage_test
COM3 (Arduino/Genuino Uno)
Send
Done uploading
2:01 / 4:40
DIGITAL(PWM~)
UNO
ARDUINO
ON
POWER
ANALOG IN

```
void setup(){
   Serial.begin(9600);              //sets the data rate in bits per second (baud) for serial data transmission
}

void loop(){
   int sensorValue = analogRead(A0);
   Serial.println(sensorValue);      //print value to serial monitor
   delay(1000);                     //delay reading by 1 second
 }
```

```
466
459
465
574
507
387
417
389
290
0
0
0
0
0
38
110
317
454
442
312
313
313
313
313
```

```
Sketch uses 1,892 bytes (5%) of program storage space. Maximum is 32,256 bytes.
Global variables use 186 bytes (9%) of dynamic memory, leaving 1,862 bytes for local variables. Maximum is 2,048 bytes.
```

Code and Testing - Arduino Voltage Sensor Tutorial

Figure 1 Arduino Tutorial

The basic concept for detecting changes in voltage came from this YouTube video. https://www.youtube.com/watch?v=MjqND1y2X_4

2. Have the IoT device send information to a web service.

The Arduino can send http post statements to a web service. The following linkedin article talks about the basics of doing this. https://www.linkedin.com/pulse/how-make-rest-api-http-post-call-using-arduino-uno-esp8266-taha-ali/

3. Use the web service to drive the graphical events in sync with the originating signal.

4. Use the web service to change the state of IoT devices such as led lights that are completely separate from the pinball machine.

1.4. Target Audience

1.4.1 Personas

Dennis

Life long pinball player

Demographics

Age:	51
Job:	Programmer
Family:	Married with 2 kids

Facts

- Owns multiple pinball machines
- Plays in a pinball league
- Attends pinball tournaments and conventions.
- Active on pinball forums
- High disposable income

Mark

Occasional pinball player

Demographics

Age:	37
Job:	Designer
Family:	Married with 2 kids

Facts

- Dreams of owning a pinball machine
- Plays in a pinball league one night a week
- Moderate disposable income

Rachel

Little to no pinball experience

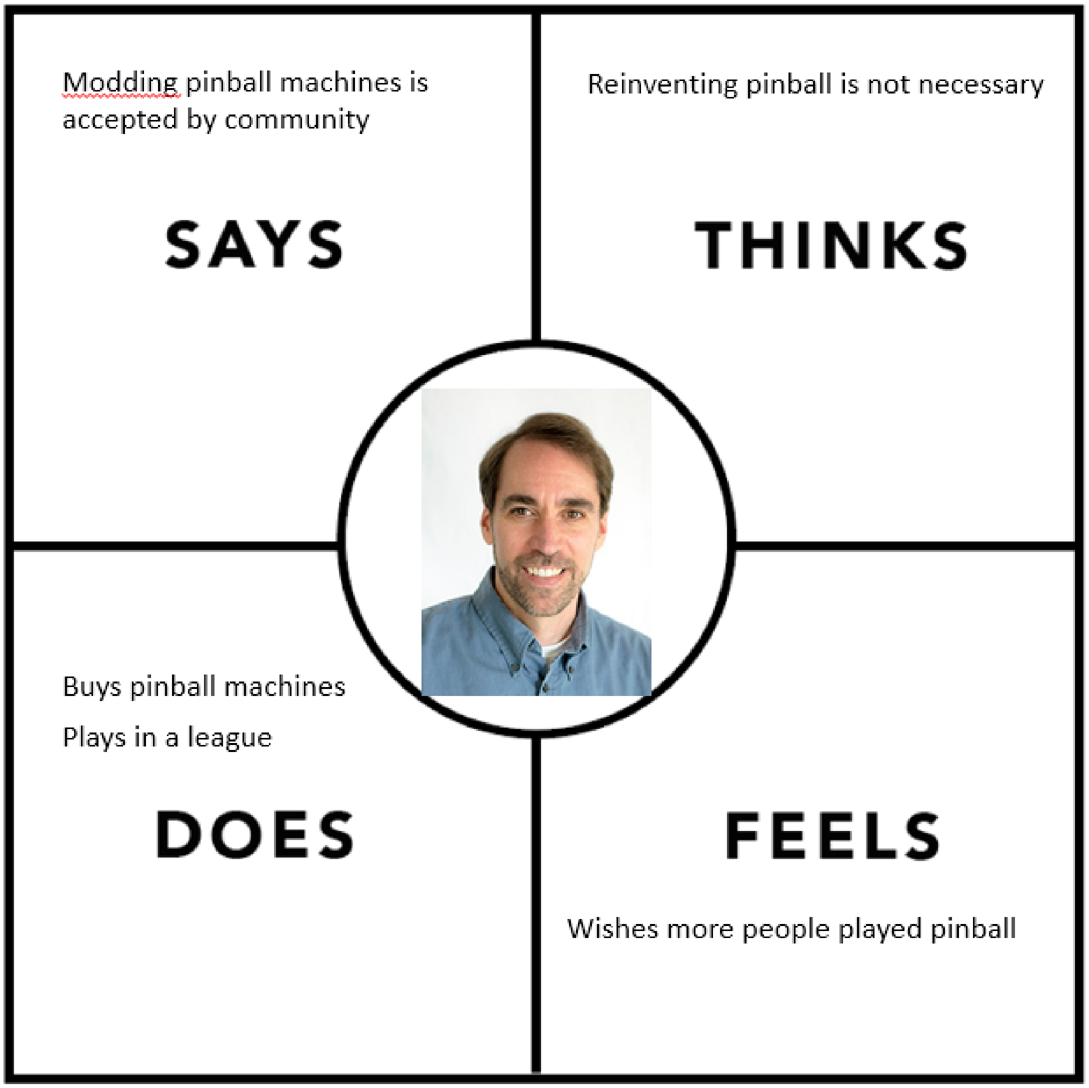
Demographics

Age:	27
Job:	Web Analytics
Family:	Single

Facts

- Has seen pinball machines at bars
- Thinks pinball looks fun
- Gaming is through phone
- Low disposable income

1.4.2 Empathy Maps



Modding pinball machines is accepted by community

SAYS

Reinventing pinball is not necessary

THINKS

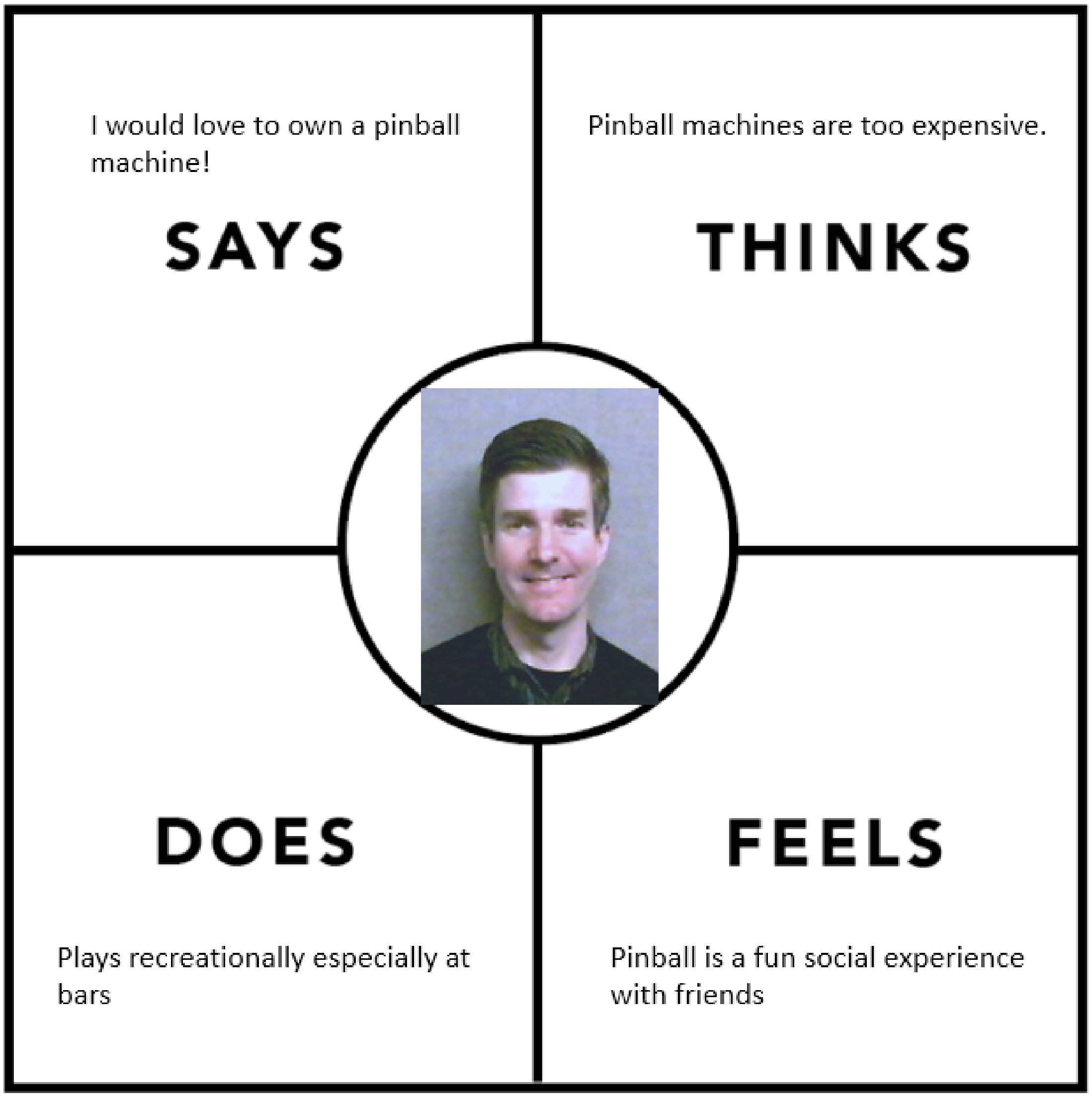
Buys pinball machines

Plays in a league

DOES

FEELS

Wishes more people played pinball



I would love to own a pinball machine!

SAYS

Pinball machines are too expensive.

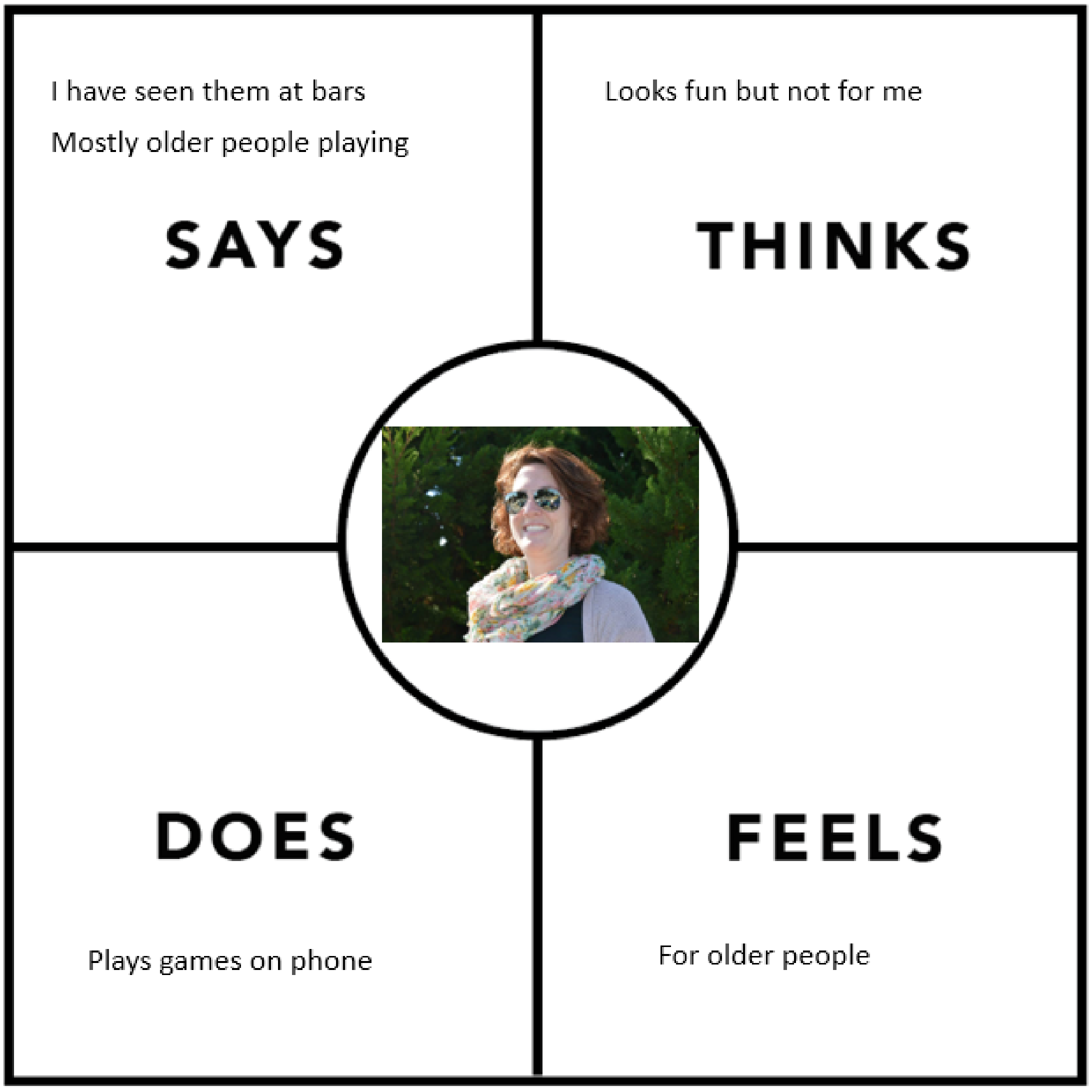
THINKS

DOES

Plays recreationally especially at bars

FEELS

Pinball is a fun social experience with friends



I have seen them at bars

Mostly older people playing

SAYS

Looks fun but not for me

THINKS

DOES

Plays games on phone

FEELS

For older people

1.5. Elevator Pitch

Older games, such as pinball, were successful because they physically engaged users in the experience. We can capitalize on this success not by completely virtualizing the experience but instead combining the physical and virtual worlds. My project will use this idea of physical computing to create a new gaming experience.

1.6. Metrics

	1	2	3	4	5
How often event fires	Distracts from gameplay				Enhances game experience
Timing of external event	Lag effects gameplay				No perceived lag
Play video on screen	Distracts from gameplay				Enhances game experience
Player experience	Does not like				Would buy
Solution Cost	> $500	> $400	> $300	> $200	<= $200

1 – non-satisfactory 5 – outstanding

1.7. Life of the project beyond capstone

Does this project exist beyond completing the capstone course?

Restoring my pinball machine is a passion project and I would like to extend its functionality and then share the results with the pinball community. Hopefully, other people will be interested and it can turn into a niche side business. However,
this project is a proof of concept around a broader idea. My ultimate goal is to integrate the viewing environment with the visual experience. I would like to create a standard by which third party providers could tie an in home movie or videogame to their smart phone, lights, or any connected technology that would enhance the experience.

Will it help you get a job, get a promotion, create a portfolio, or launch a business?

While it will not get me a job, I believe showcasing a wide range of talents and interests on your resume and LinkedIn profile can lead to opportunities. As mentioned previously, my project could lead to a small side business.

Will your capstone be used or experienced by your target audience after this class?

Use by my target audience depends on how hard it is to implement my idea and cost. I view this project as my proof of concept and feedback from the pinball community will determine if it gets adopted by the larger audience.

2. Competitor review

For my project, I am listing competitors in three categories. The first is new pinball machines. These machines have been manufactured in the last 10 years. For the most part they are still electro-mechanical but have a PC and Ethernet bus system instead of custom hardware to drive the game. With the PC they can integrate video and internet connections. The second category is virtual pinball machines. These solutions range from software based pinball fields to full virtual reality fields. Finally, there is the pinball modding community. All sorts of add-ons and additions are made for pinball machines. While none of these options are exactly what I am proposing there are pieces of each one in my project.

Competitor Matrix

	Cost	Setup Complexity	Interaction
New Pinball	$7k+	High	High
Virtual Pinball	$30 - $2500	Medium	Medium
Mods	$30 - $200s	Medium	Low
Project	<$200	High	High

2.1 New pinball machines

All newly manufactured pinball machines are computer based. There are a handful of manufacturers making them and only in limited releases. The top manufacturers are:

1. https://sternpinball.com/
2. https://www.jerseyjackpinball.com/

Both produce pinballs with LCD for back glasses (as seen in image below) and use a PC to drive the pinball game. The PC runs custom code and is only updateable by the manufacturer.

Figure 2 Modern pinball machines

2.2 Virtual Pinball Machines

Virtual Pinball Machines are fully rendered in software with a physics model that mimics real life play mechanics. The software allows the player to choose from a list of pinball machines. The playfield is then rendered on a 4k monitor.

2.2.1 LCD based and computer driven

Virtua Pin

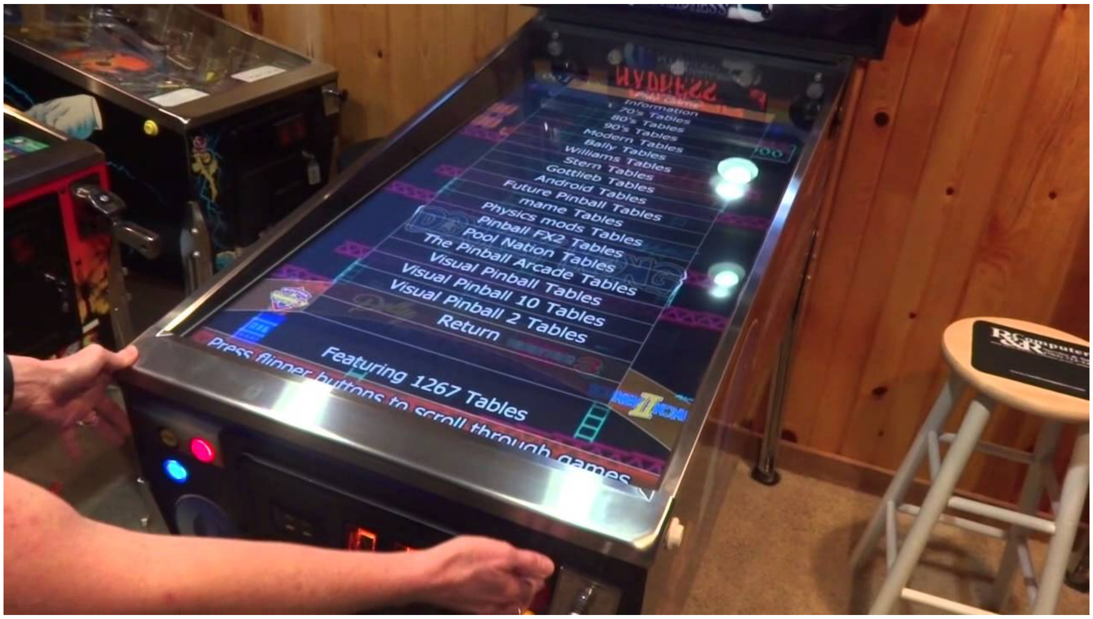


Figure 3 Virtual Pinball Machine

https://virtuapin.net/
Pros: Almost any pinball machine playfield is available.
Cons: No tactile feedback. It has the look of pinball but lacks the feel.

2.2.2 Virtual Reality Pinball

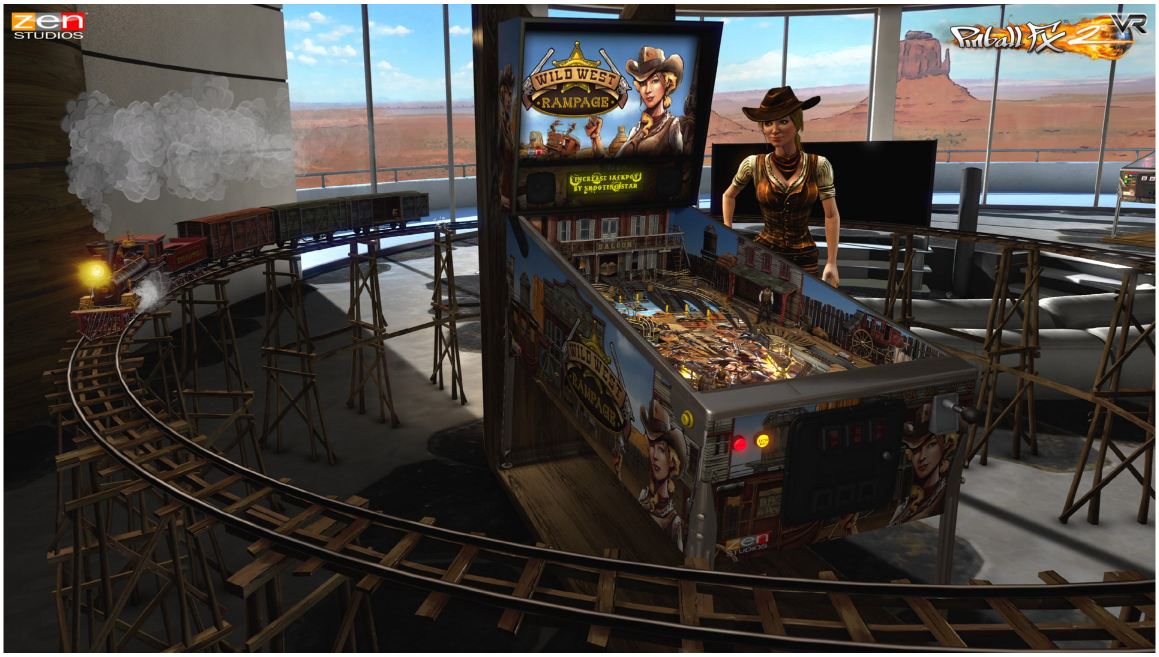


Figure 4 Virtual Reality Pinball

https://www.oculus.com/experiences/rift/1044201475659209/
https://www.youtube.com/watch?v=TLYIEm9kkvU
Pros: Little to no physical space needed. Large collection of playfields
Cons: No tactile feedback. Current hardware has trouble rendering pinball speed properly. Removes social aspect of playing.

2.3 Mods

Mezel Mods
https://mezelmods.com/collections/tv-video-display-mods

Pros: inexpensive, comes with pre-loaded video
Cons: limited to certain machines, has to be attached to playfield, no customization

Modfather Pinball Mods
https://modfatherpinball.com/collections/mods

Pros: Highly customized accessories
Cons: limited to certain machines, no change to gameplay, look only

Lighted Pinball Mods
https://lightedpinballmods.com/

Pros: Large assortment of lighted mods for most pinball machines, driven by Pinball machine power, turns on with Pinball machine
Cons: Look only, no effect on gameplay. Most mods light when machine is turned on and not to game event.

3. Technology requirements

Technology #1: Arduino or Raspberry Pi
Reason: I need a way of reading voltage changes when lights and sensors are tripped in the pinball machine. Both the Arduino and Pi have boards available for this type of interfacing.
I will be learning this technology as I go.

Technology #2: NodeJS
Reason: Node is used for back end infrastructure builds. It allows me to use JavaScript (which I am very comfortable with) to construct web APIs. Being open source, it has a large library of modules for me to choose from when adding functionality.
CSCI E-31 Web App Development/Node.JS, CSCI E-3 Web Programming / JavaScript

Technology #3: Home Automation (i.e. Insteon ISY)
Reason: Allows the project to control devices plugged into any power outlet or available via Wi-Fi.
I have installed Insteon digital switches but have not tried to control them programmatically.

Technology #4: Augmented Reality System
Reason: I am not sure what system, if any, I will be able to use. This part of the project may more of a survey of what could work.
I will be learning this technology as I go.

4. Visuals describing how your product will be used

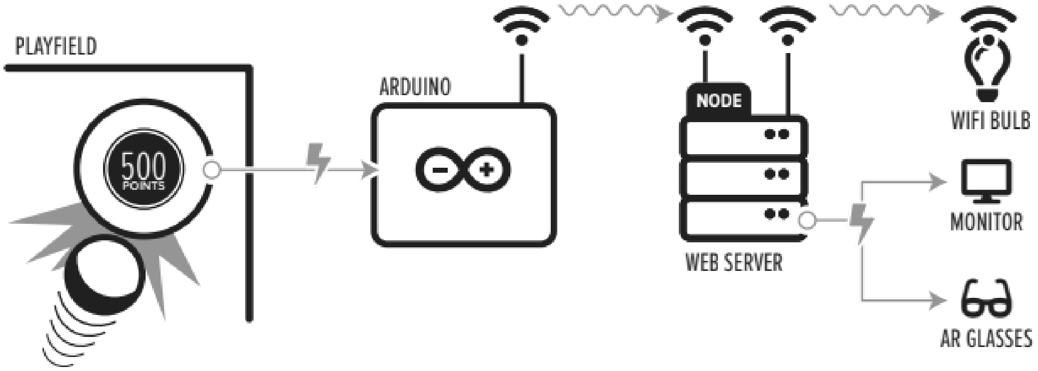


Figure 5 Communication paths

Installation

1. The user will connect the analog ports of the Arduino to the leads of the LEDs via alligator clips.

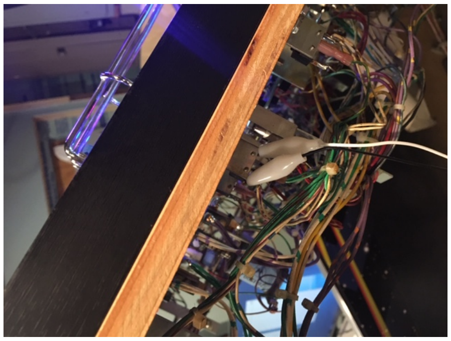
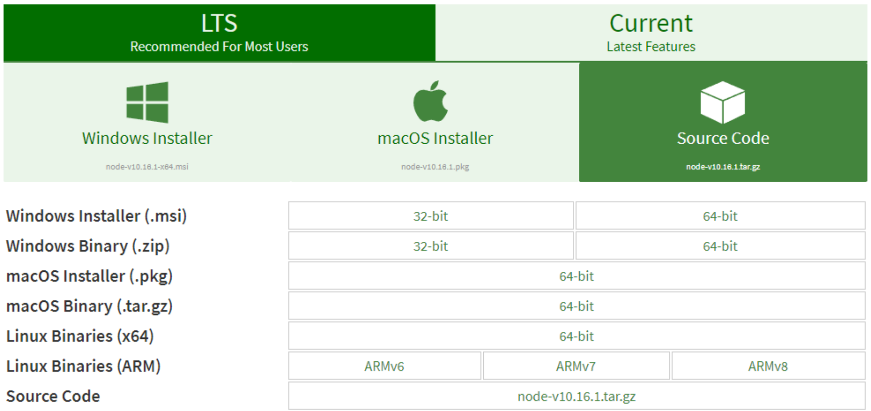


Figure 6 Arduino connection inside of pinball machine

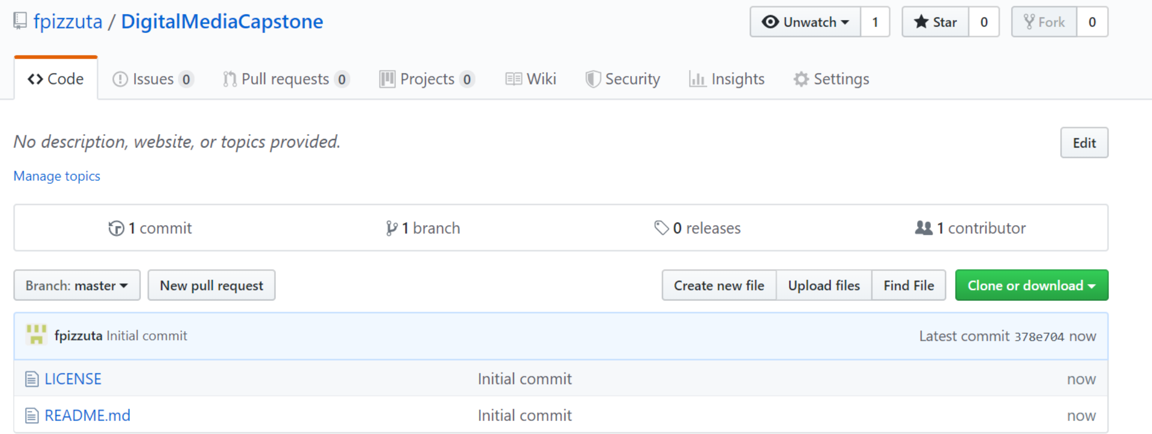
2. The Arduino will be plugged into a power port inside of the pinball machine and connected to wifi.
3. A PC with monitor will be setup near pinball machine.
4. Nodejs software will be installed and run on this PC.



Windows Installer (.msi)	32-bit	64-bit	
Windows Binary (.zip)	32-bit	64-bit	
macOS Installer (.pkg)	64-bit		
macOS Binary (.tar.gz)	64-bit		
Linux Binaries (x64)	64-bit		
Linux Binaries (ARM)	ARMv6	ARMv7	ARMv8
Source Code	node-v10.16.1.tar.gz		

Figure 7Install latest version of NodeJS

5. API software will be downloaded onto PC from Github



No description, website, or topics provided.

1 commit | 1 branch | 0 releases | 1 contributor

fpizzuta Initial commit — Latest commit 378e704 now

LICENSE	Initial commit	now
README.md	Initial commit	now

Figure 8 Project source code on Github

Monitor Version:

1. In the future, the user will also download the multimedia appropriate to their pinball machine. For purposes of this project, media will be included with the API install.
2. Game can now be played. When connected LEDs light the appropriate media should show on monitor.

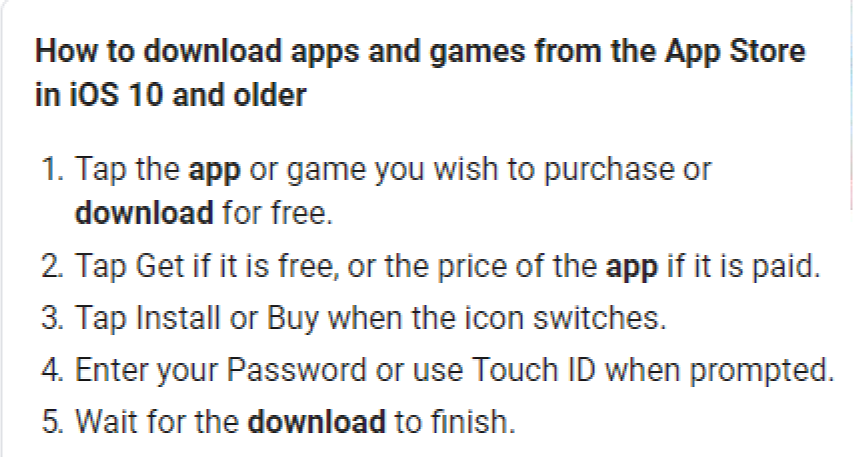
Insteon Version (monitor only needed for configuring PC)

1. Download Insteon connectivity software from GitHub.
2. Configure config file for Insteon API to allow access to Insteon ISY controller

3. Create scene in ISY for lights near pinball machine
4. Add scene name to API config file
5. Game can now be played. When connected LEDs light, home automation events should activate.

AR Version

1. For purposes of this project, steps for Monitor Version should be completed and working.
2. Download AR API software from GitHub.
3. Install AR app on phone.
4.



How to download apps and games from the App Store in iOS 10 and older

1. Tap the **app** or game you wish to purchase or **download** for free.
2. Tap Get if it is free, or the price of the **app** if it is paid.
3. Tap Install or Buy when the icon switches.
4. Enter your Password or use Touch ID when prompted.
5. Wait for the **download** to finish.

Figure 9 Install app on iphone or ipad

5. Game can now be played. When looking through phone or glasses while playing pinball QR code will show on monitor and be replaced via AR software with AR media.

5. Work plan and milestones

This project will have four phases.

Phase 1:
The first phase will be to successfully read a voltage change on the pinball machine and call a web service as a result. The voltage change can be the result of a led lighting or an electro-mechanical sensor tripping.

Milestone

1. Arduino detects voltage when led is lit.

Phase 2:
The second phase will involve playing a video, sound, etc. when the web service is called. This will likely be accomplished by sending html to a dynamic web page.

Milestone

1. Arduino detects voltage when led is lit.
2. Web service properly responds to test calls.
3. Web service updates web page when call from Arduino is received.

Phase 3:
The third phase would involve controlling devices such as led bulbs, strip lights, etc. that are not connected to the pinball machine.

Milestone

1. Home automation light turns on when web service receives message from Arduino.

Phase 4:
The fourth and final stage would be to investigate showing the video through an augmented reality setup so that the animation blends seamlessly with the gaming experience.

Milestones:

1. Augmented Reality media is displayed when QR code appears.

User testing will be performed by members of a local pinball league and a group of non-serious pinball players. I am curious to see if hard core players will accept a new concept applied to their hobby.

From a project management perspective, I will use a Lean methodology approach. I am looking at the capstone as my minimal viable product (MVP). I need to test the core concepts of my idea and then be ready to change my approach if it doesn't test well. Most importantly, I need to get something into the hands of my users as opposed to building the whole system without any feedback.

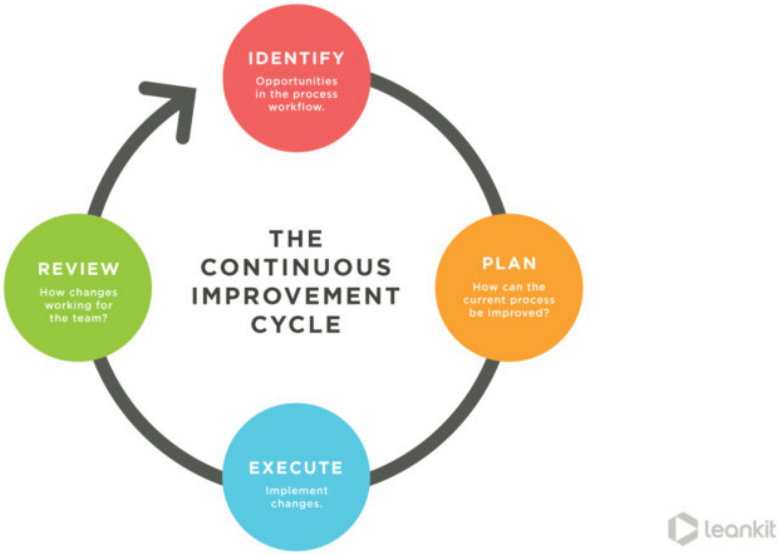
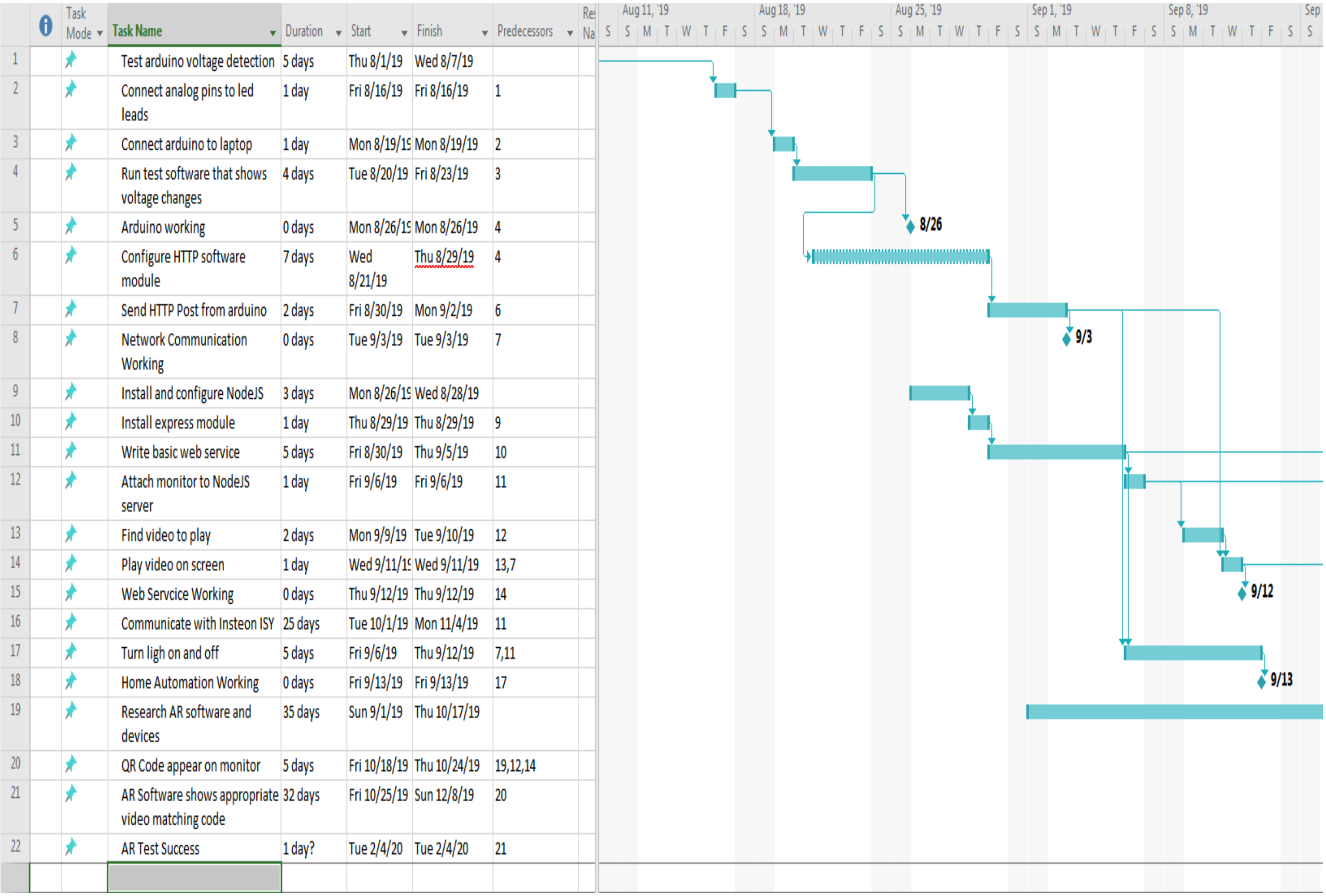

IDENTIFY
Opportunities in the process workflow.
PLAN
How can the current process be improved?
EXECUTE
Implement changes.
REVIEW
How changes working for the team?
THE CONTINUOUS IMPROVEMENT CYCLE
leankit

Figure 10 Lean Continuous Improvement (from leankit.com) (leankit.com, 2019)



		Task Mode	Task Name	Duration	Start	Finish	Predecessors	Re Na
1			Test arduino voltage detection	5 days	Thu 8/1/19	Wed 8/7/19		
2			Connect analog pins to led leads	1 day	Fri 8/16/19	Fri 8/16/19	1	
3			Connect arduino to laptop	1 day	Mon 8/19/1!	Mon 8/19/19	2	
4			Run test software that shows voltage changes	4 days	Tue 8/20/19	Fri 8/23/19	3	
5			Arduino working	0 days	Mon 8/26/1!	Mon 8/26/19	4	
6			Configure HTTP software module	7 days	Wed 8/21/19	Thu 8/29/19	4	
7			Send HTTP Post from arduino	2 days	Fri 8/30/19	Mon 9/2/19	6	
8			Network Communication Working	0 days	Tue 9/3/19	Tue 9/3/19	7	
9			Install and configure NodeJS	3 days	Mon 8/26/1!	Wed 8/28/19		
10			Install express module	1 day	Thu 8/29/19	Thu 8/29/19	9	
11			Write basic web service	5 days	Fri 8/30/19	Thu 9/5/19	10	
12			Attach monitor to NodeJS server	1 day	Fri 9/6/19	Fri 9/6/19	11	
13			Find video to play	2 days	Mon 9/9/19	Tue 9/10/19	12	
14			Play video on screen	1 day	Wed 9/11/1!	Wed 9/11/19	13,7	
15			Web Servcice Working	0 days	Thu 9/12/19	Thu 9/12/19	14	
16			Communicate with Insteon ISY	25 days	Tue 10/1/19	Mon 11/4/19	11	
17			Turn ligh on and off	5 days	Fri 9/6/19	Thu 9/12/19	7,11	
18			Home Automation Working	0 days	Fri 9/13/19	Fri 9/13/19	17	
19			Research AR software and devices	35 days	Sun 9/1/19	Thu 10/17/19		
20			QR Code appear on monitor	5 days	Fri 10/18/19	Thu 10/24/19	19,12,14	
21			AR Software shows appropriate video matching code	32 days	Fri 10/25/19	Sun 12/8/19	20	
22			AR Test Success	1 day?	Tue 2/4/20	Tue 2/4/20	21	

Aug 11, '19
Aug 18, '19
Aug 25, '19
Sep 1, '19
Sep 8, '19
Sep
S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S M T W T F S S
8/26
9/3
9/12
9/13

Figure 11 Project plan

6. Results

6.1 Metric Results

	1	2	3	4	5
How often event fires			x		
Timing of external event					x
Play video on screen					x
Player experience			x		
Solution Cost					X

Only two metrics got lower scores. Both were related to the fact that I only wired one switch and therefore players could only experience effects when that switch fired. To be a viable product, more switches need to be connected to the Arduino.

A key concern when I started this project was the lag between a reading the switch and responding with an external event. Luckily, my fears were unfounded. The events happen so quickly that there appears to be no lag at all.

6.2 Technology Used

Technology #1: Arduino

Technology #2: NodeJS

Technology #3: Lifx bulbs
Reason: I originally thought I would use a home automation standard such as Insteon to control the lights. After some research I found that Lifx produced inexpensive wifi bulbs (figure 13). By using an open source lifx/nodejs library I was able to achieve the effects I needed for a fraction of the cost.

Figure 12 Lifx bulb

Technology #4: Augmented Reality System - Wikitude
Reason: After playing around with a bunch of systems I settled on Wikitude. https://www.wikitude.com/ Their approach allowed me to write the app in html, JavaScript, CSS, and their js library. Then using Cordova (PhoneGap) I could compile it into a native IOS app using XCode.

Technology #4: Voltage Detector
Reason: The Arduino cannot survive more than 5 volts on its inputs. In order to ensure this limit was never surpassed I used a voltage detector (figure 14) that ensures higher voltages are dropped below 5 volts. This device had the added benefit of allowing me to drop low transient power spikes thus ensuring clean detection of a switch being thrown.

Figure 13 Voltage detector

Technology #5: Web Sockets
Reason: Socket.io https://socket.io/ is a NodeJS library for web sockets, providing bi-directional communication between a browser and the server. Both endpoints listen for specific strings passed through this communication channel in order to execute call specific functions.

Technology #6: Cordova
Reason: Cordova https://cordova.apache.org/ is an open source solution for converting HTML, CSS, and JavaScript into a native mobile application. An added benefit was the ability to use the web socket code on the iPhone which is not easily available in Swift coding.

6.3 Code Repositories

Server side: https://github.com/fpizzuta/DigitalMediaCapstone
Arduino code: https://github.com/fpizzuta/CapstoneArduino
App code: https://github.com/fpizzuta/CapstoneApp/tree/master/www/world/01_ImageTracking_1_ImageOnTarget

A note on the app code. Because I used the wikitude example app and was unsure what I could remove safely the project size was larger than GitHub allowed. I therefore uploaded only the portion of the sample project I made changes to. The full example app from the vendor can be found on their GitHub page, https://github.com/Wikitude/wikitude-cordova-plugin

6.4 Featured Code

As there is a large amount of code in github, this area will be used to call out the important code segments for each area of the project.

6.4.1 Server

The server side of this project uses nodejs and a standard express setup. Two libraries are loaded: node-lfx-lan and socket.io. Socket.io needs to listen in on any http coming to the server. To accomplish this I bound socket.io to the http object in the bin/www file:

```
server.listen(port);
server.on('listening', () => {
   console.log("Listening on %s", server.address().port);
});
var io = require('socket.io').listen(server);
app.io = io;
```

app.io = io is used to access socket easily from other modules.

6.4.2 Web Sockets

Web sockets provide bi-directional communication between a browser and the server. Both endpoints listen for specific strings passed through this communication channel in order to execute call specific functions.

```
 req.app.io.emit('master','You lose Doctor!');
```

The command emits a message named ‘master’ with a data payload of the string ‘You lose Doctor!’).

On the browser side, the socket.io library is loaded:

```
<script src="https://cdn.socket.io/socket.io-1.4.5.js"></script>
```

An instance is then declared and the named message listened for:

```
var socket = io();

socket.on('master', function(msg){
            $('#master').toggle();
          });
```

In this case, the code is simply toggling an image’s visibility.

6.4.3 Wifi Bulbs

The node-lifx-lan library is loaded in routes/pinball.js:

```
const Lifx  = require('node-lifx-lan');
```

This library allows us to talk to wifi enabled LED bulbs.

The following code discovers any wifi enabled bulbs on the network and places them in a list within the Lifx object.

```
Lifx.discover().then((device_list) => {
  device_list.forEach((device) => {
    console.log([
      device['ip'],
      device['mac'],
      device['deviceInfo']['label']
    ].join(' | '));
  });
}).catch((error) => {
  console.error(error);
});
```

Whenever a post of /drwho is received, the Lifx object is used to toggle the lights.

```
router.post('/', (req, res, next)=>{
 console.log('/drwho requested');
 req.app.io.emit('master','You lose Doctor!');
 // Turn on all LIFX bulbs in the local network
 if (req.app.locals.lightsOn == 0) {
    Lifx.turnOnBroadcast({
      color: {css: 'green'}
    }).then(() => {
      console.log('Light On!');
    }).catch((error) => {
      console.error(error);
    });
    req.app.locals.lightsOn = 1;
 } else {
    Lifx.turnOffBroadcast({
      duration: 3000
    }).then(() => {
      console.log('Light Off!');
    }).catch((error) => {
      console.error(error);
    });
    req.app.locals.lightsOn = 0;
 }
 res.end();

});
```

6.4.4 Augmented Reality

The app was created using Cordova and the Wikitude javascript sdk. Cordova allows for development of an app through html, css, and javascript. This code is then loaded into the appropriate native editor, in this case Xcode, and compiled into a native app. Because of the ability to use javascript, the app is able to leverage the socket.io library and work similar to the browser example.

```
var socket = io.connect('http://192.168.1.48:8080');

 socket.on('master', function (data) {
              if ( World.pageOne.enabled == true) {
                World.pageOne.enabled = false;
              } else {
                World.pageOne.enabled = true;
              }
  });
```

In this case, the app connects to the server. The connection establishes the web socket and allows the app to receive any emits from the server. After that, it is simply a matter of enabling the appropriate AR image to appear.

6.4.5 Arduino

The core of the Arduino code is to send a post to the node server when voltage is detected. The Arduino ethernet library provides an object for performing posts and gets.

```
EthernetClient client;

if (client.connect("192.168.1.48", 8080)) {
        Serial.println("connected");
        client.println("POST /drwho HTTP/1.1");
        client.println("Host: 192.168.1.48");
        client.println("Content-Type: application/x-www-form-
urlencoded");

        client.println("Connection: close");
        client.println("User-Agent: Arduino/1.0");
        client.println();

 }
```

It is important to declare the content-type correctly in the pack so that the server can respond appropriately.

In order to not read the same switch signal multiple times a 200 ms delay is inserted.

```
delay(200);
```

6.5 Installation Instructions

6.5.1 Connect Arduino to Pinball

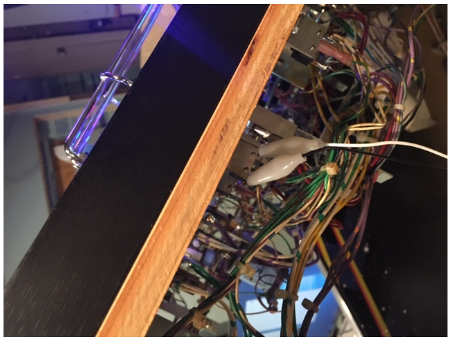


Figure 14 Alligator clips attached to switch

Load code onto Arduino. Connect the voltage regulator to analog port 0 and ground. Then connect the voltage regulator to the switch with alligator clips (figure 15). Because there are many different styles of switches, you may need to look at the manufacturer specifications to determine the correct points for tapping the switch.

6.5.2 Build Server

Install nodejs for your preferred platform https://nodejs.org/en/download/.
Load socket.io https://socket.io/
Lifx node modules https://www.npmjs.com/package/node-lifx-lan
Install server code from github. https://github.com/fpizzuta/DigitalMediaCapstone
Start web service with: *npm start*.

6.5.3 Test connection

Open a browser to the server. With pinball machine on, press switch. An image should appear on screen.

6.5.4 App

Download app code from github https://github.com/fpizzuta/CapstoneApp .
Open in Xcode and push code to attached phone. App is now ready to be used.

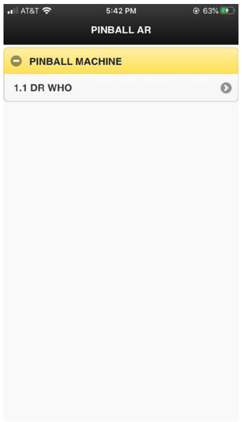


Figure 15 Pinball selection page of app.

Open app (figure 16) and select pinball machine. Look at the pinball machine through the camera. The pinball machine will be detected and scanning header will disappear (figure 17).

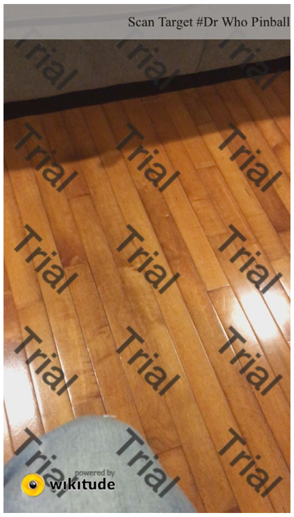


Figure 16 AR app scanning for pinball backglass.

Once again, trigger the switch. An image of the Master should appear, floating over the pinball machine (figure 18).

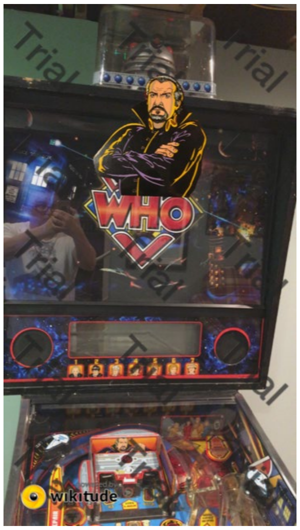


Figure 17 AR image floating over pinball machine.

6.5.5 Troubleshooting

You can use Postman https://www.getpostman.com/ to send posts to the server. This is a great way to test events without requiring the pinball machine to be on.

6.6 Final Thoughts

All major objects where achieved: converting an analog signal to digital, triggering a browser change on a screen, turning a light on and off, and affecting an augmented reality experience. Architecturally, the only change made was moving from Insteon for light control to direct wifi access. This was a net positive change as it makes the solution both cheaper and more flexible.

The biggest disappointment was augmented reality. Many of the AR firms have suffered due to lack of adoption and a true AR glasses solution. Another big issue, the AR libraries are handicapped by expensive licensing costs. The free versions have severe watermarking that makes testing very difficult. With Apple a few years from releasing their glasses solution, AR stays in the novelty realm.

There are many possible next steps for this project. Using sound recognition to trigger the events would remove the need for a user to tap switches, making for a much more accessible and expandable solution. The app framework can easily add more pinball machines, the only requirement being quality pictures of the back glass. Arduino and Node code can be used with buttons, touchscreens, etc. to enable physical computing results with both AR and home automation. I am hopeful that when true AR glasses go mainstream, I will be able to make a marketable solution.

7. References

Pinball
Bradford, Harry (June 26, 2013). "Jersey Jack Pinball CEO Explains How He's Making The Beloved Arcade Game Cool Again (VIDEO)". Huffingtonpost.com. Retrieved July 29,2019.

Virtual Pinball
"Visual Pinball", Sourceforge.net, Retrieved July 29,2019

Insteon
"Refresh!: Insteon Technology". *Electronic Design*. Penton Media, Inc. April 5, 2006. Retrieved July 29,2019

Arduino
David Kushner (2011-10-26). "The Making of Arduino". *IEEE Spectrum*.

NodeJS
"PHP 7 vs Node.js? They Can Be Partners, Not Competitors For a Developer!"
. Retrieved July 29, 2019

QR Codes
"What is a QR Code?", qrcode.com, retrieved Jul 29,2019

Augmented Reality
Metz, Rachael (2 August 2012). "Augmented Reality Is Finally Getting Real"
. technologyreview.com. Retrieved 18 June 2019.

lean-methodology
leankit.com. (2019, 04 8)
https://leankit.com/learn/lean/lean-methodology/

Apache Cordova
https://cordova.apache.org/, retrieved Sept 15, 2019.

Wikitude
https://www.wikitude.com/external/doc/documentation/latest/phonegap/, retrieved November 12, 2019.

LifX-Lan
https://www.npmjs.com/package/node-lifx-lan, retrieved October 2, 2019.

Arduino Ethernet Library
https://www.arduino.cc/en/reference/ethernet, retrieved August 10, 2019.

Socket.io Sample Chat App
https://socket.io/get-started/chat, retrieved Sept 28, 2019.

Wikitude Sample App for Cordova Plugin
https://github.com/Wikitude/wikitude-cordova-plugin retrieved November 12, 2019.