ANIMATRONICS FACE: Dr. JoAnn Browning

https://youtu.be/iHOl3xcqUYQ

Victoria Wahlen
Dept. of Mechanical Engineering
San Antonio, TX, USA 78249
ify252@my.utsa.edu

Sabrina Hamdan Shepard
Dept. of Mechanical Engineering
San Antonio, TX, USA 78249
qtg349@my.utsa.edu

Jordan Traxler
Dept. of Mechanical Engineering
San Antonio, TX, USA 78249
pob174@my.utsa.edu

Submitted as part of project for ME4543 Mechatronics, Fall 2018

ABSTRACT

The purpose of this project was to bring to life an animatronics version of the one and only Dr. JoAnn Browning, Dean of the College of Engineering at The University of Texas at San Antonio. To do this, other animatronics projects were researched to gather an understanding of what is required to accomplish such a feat. The project required a minimum of two sensors, two actuating components, and a talking segment that is synchronized with the jaw motion. For the purposes of Dean Browning, she was equipped with an RFID sensor, requiring the audience to scan in to see if they would have the VIP access required to speak with the Dean, and two ultrasonic sensors that are connected to her eyes and used to track the audience. Her voice segment stated "Welcome to the College of Engineering! Birds Up! One more time, with feeling, BIRDS UP!", and the audio is triggered and played through a computer using a seperate program to listen to the COM port on the Arduino's serial monitor. These essential functions were accomplished through an Arduino Mega, the two sensors, three primary joints, an Arduino code to control the system, and a computer to power and trigger the audio.

Section 1: Literature Review

In order to begin the process of creating an animatronic face, several animatronic face projects were researched to gain an understanding of the process involved in the project. Each of the researched faces took different approaches towards creating their animatronic masterpieces, as evidenced by uniqueness of each design.

One of the first researched animatronics faces was called CoconutHead, which utilized an actual coconut to create the required circular head shape of the face. To create the head, a coconut was cut in half and sockets were cut out where the eyes would be placed. This animatronics project did not feature an animated mouth; hence, there was no need to carve out an area for the mouth. The project was done by a hobbyist, and the main feature the hobbyist chose to accentuate was the eye movement. To achieve not only eye movement but eyes that could track motion, three ping ultrasonic sensors were utilized to recognize and track the motion, and an Arduino Uno was utilized as the microcontroller of choice for this project. The written code to control the ping ultrasonic sensors was tested by connecting each sensor to the Arduino and a breadboard and using three separate LED circuits, one wired to each sensor. Then, the code was uploaded to the board, and the hobbyist waved his hand in front of each sensor. As his hand moved in front of a sensor, the associated LED would light up; hence, the system was verified to be in proper working order. These sensors were then programmed with an algorithm to sense direction, and the sensor output was converted into servo motion with a servo motor. A shield for the entire set up was then created, and the system was mounted inside of the coconut and attached to shiny, plastic eyeballs. The finishing touch made to Coconut head, aside from decorating, was to add a clear, plastic clown nose that lit up when motion was detected in certain ping ultrasonic sensors. This finishing element only required extra code for the sensors, an LED, and the nose itself. After decorations were added, CoconutHead's look and abilities were complete (Fig. 1) [1].

Figure 1. *The One and Only--CoconutHead [1]*

The second researched animatronic face was not given a name by his creator, so he will be referred to as "The Nightmare Bot" for the purposes of this report. The creation video of The Nightmare Bot provided instruction for the frame of the eyes, eyelids, and mouth, as well as the mechanisms that facilitated the motion of each component. The frame was structured out of aluminum strips and sheets that were cut and bent into the required shape to house the servo motors, create the forehead, holders for the eyeballs, and the eyelids. The eyeballs were created by cutting a rubber bouncy ball in half and painting terrifying irises on them. A small hole was pierced through the flat back of each eyeball so that they could be fastened to the screws protruding from the eyebrow base plate. Next, the aluminum eyelid bands were fastened around the eyeball holder and independently attached to a single servo motor using a metal wire. The upper and lower jaws were created in a similar fashion by bending, cutting, and painting aluminum sheets. Both jaws were affixed directly onto the second servo motor, and the upper jaw was fixed in place and attached to the lower jaw at a joint formed by a screw and nut. This joint allowed the lower jaw to move freely. The lower jaw was then attached to the servo motor at a point behind the joint using a metal wire to mimic the motion of a human mouth. The eye and jaw fixtures were connected to each other using a large strip of aluminum which wrapped around the back of the robot. The entire structure was then mounted on an aluminum stand. To generate motion, two servo motors were used with a 555 timer IC based circuit to create a pulse from a variable resistor. The final form of The Nightmare Bot is depicted in Figure 2 [2]. The Nightmare Bot can be a helpful tool for reference when designing the mechanisms for jaw and eyelid motion in the animatronic recreation of a UTSA professor.

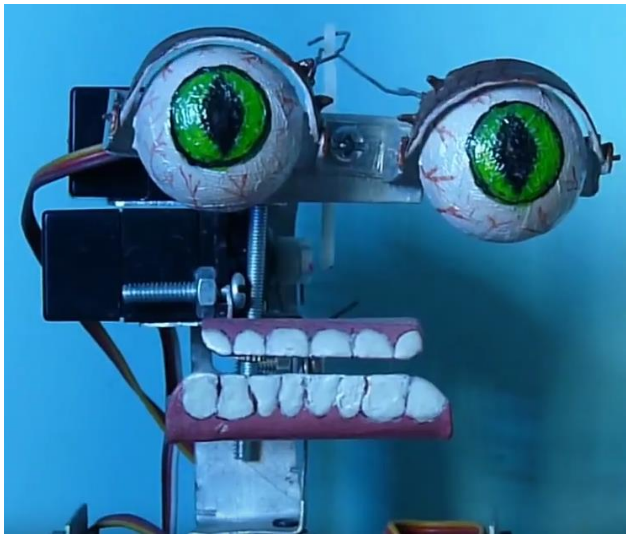

***Figure 2**. The Nightmare Bot [2]*

Robot Head 2 was yet another type of animatronics face that was researched. It mainly features a man’s head from the neck up, and it is an abstract representation of the face since the outline of the head can be observed from the side but not the front. This abstract aspect of the face is accentuated by having facial components like the eyes and mouth seemingly float without being limited to the actual facial frame (Fig. 3). This animatronics head is composed of a plywood support box and face frame, with a moving jaw and latex lips to simulate speaking, and a moving brass rod to simulate eyes that blink occasionally. The list of materials includes a servo controller (MiniSSCII), powered computer speakers, a single board computer (RAPU) with its compact flash card, a fan for heat dissipation, and one micro switch to control a spotlight for dramatic effect [3]. The RAPU stores the movement commands for the eyelids and the mouth, as well as the audio portion the robot speaks. Then, the servo controller takes serial signals from the RAPU and translates them into servo movements. According to the creator of the Robot Head 2, programming the head to not only speak and blink, but to also perform facial expressions that correspond to certain lines, such as tilting the head slightly sideways when asking a question, was the most challenging part of the project. The creator referenced it took about an hour of programming to achieve a single minute of motion. The code for these movements began as written dialog, and with the use of a text-to-speech utility, the written dialog was converted to a spoken dialog, which resulted in an mp3 file. The next step was programming the actual servo motors to accomplish the desired motion. Once the programing of the servos was complete, both the servo control file and the mp3 file were downloaded as a single control file onto a compact flash card, which went into the RAPU and completed the process of building this animatronics head. The purpose of building this head was for the entertainment of the user and of the audience, particularly children. The Robot Head 2 tells them funny stories about himself, gives information on technology, and encourages the kids to study hard in school. When he is done speaking, he says goodbye and closes his eyes to go to sleep. This head does not have any sensors since its main function is to inform the audience, which can be accomplished by running the code without physically interacting with the listeners. The creator of this robot had made a previous creation, which would interact with the Robot Head 2. This second creation would be used potentially for asking occasional questions and casually conversing about technology with Robot Head 2, thus communicating the information to the audience.

Figure 3*. Robot Head 2 [3]*

After researching these animatronic faces, various methods on the application of sensors and motors within animatronics was discovered. These examples will be useful in determining the best approach to take when creating the face and personality of one of the professors within the mechanical engineering department at The University of Texas at San Antonio.

Section 2: Brainstorming and Initial Planning

In order to begin the task of creating the animatronics face of Dr. JoAnn Browning, Dean of the College of Engineering at The University of Texas at San Antonio, the team needed to brainstorm different ways to accent her personality in the creation of the face, while completing each requirement of the project. The team will meet with the Dean, where the team will observe which characteristics will be represented in the animatronics version of Dr. Browning. These observations will allow the team to determine the main features and components of the animatronics face.

First, an important aspect to any person is the ability to communicate with others. This will be a main priority of the animatronics face so that Dean Browning will be able to interact with those passing by the display. As of now, this will be done using a servo motor to partially rotate the joint that controls the jaw. This servo motor will then be programmed and connected to an Arduino Mega 2560 so that the jaw will move according to when the Dean is talking. Since the mouth will need not just motion but a voice to go with it, a speaker will also be used and connected to the Arduino. This speaker will not be visible to the audience, as to not take away from the actual face. A motion sensor will also be connected to the Arduino and will be implemented as the main stimulus for the animatronics face. Similarly to the speaker, the motion sensor will be placed in such a way as to not distract the audience from their interaction with the animatronics face. However, unlike the speaker, the motion sensor must be placed somewhere in front of the main structure to allow the signal to reach the audience and bounce back to the sensor; hence, it must not be obstructed by any part of the structure. Whether the team utilizes an infrared sensor or an ultrasonic sensor will be decided after further research is conducted in order to determine which of these is more apt for the project's goals. The speaker will be programmed to start in unison with the jaw motions, and this command queue will start once motion is detected.

Additionally, the Dean's eyebrows will be animated to accentuate the enthusiasm with which she speaks. This will likely be done using a servo motor to control the up and down movement of both eyebrows. Since the eyebrows will need to move in unison, there is no need to have more than one servo motor to control both eyebrow movements. Both eyebrows will be connected with a wire or a free-floating hinge in the middle that will be attached to the servo motor. The servo motion will be vertical to either raise or lower the eyebrows. The outer edges of each eyebrow will be permanently fixed with a hinge to allow only rotation. The movement of the eyebrows will be programmed in Arduino and timed to occur with certain parts of the Dean's speech.

If time permits, the team has a few goals to add flair to the project. If possible, the team would like to place emphasis on Dean Browning's signature golden blonde hair. To accomplish this, the hair will be simulated either with a wig or curling ribbon and attached to a plate located at the top of the head. The plate will be supported by a single beam in the center that has a pivot joint to allow the plate to move vertically, similar to the motion of a seesaw. To facilitate this motion, a single side of the plate will be fixed to a linearly actuating servo motor which will be activated by the selected motion sensor. Additionally, the team will attempt to simulate the sparkle in Dean Browning's eyes by placing an LED behind each eye that will be activated by the motion sensor. If this goal is to be accomplished, the eyes would have to be made of a semi-transparent material, such as a ping pong ball, to allow the light to shine through. The LED can be used to simulate a wink or blink, depending on the selection made by the team.

Once the team meets with Dr. JoAnn Browning, an audio recording of her voice and a high-quality picture will be taken. Currently, the team is thinking of having the Dean say a welcoming message along the lines of "Welcome to the College of Engineering at UTSA!" There is also the possibility the Dean says something funny like an engineering joke to get a laugh out of the audience. The picture obtained from the meeting will be used to place on the project, which will be done by simply printing the Dean's picture on paper and attaching it to the structure of the face. The main structure for the face will likely be made of a light and malleable material that is easy to work with and allows the servo motors to function to their full capacity without having issues of weight. Currently, cardboard and plastic masks are being considered to serve as structural materials. As for the chassis that will physically support the face, sturdier materials will be required such as plywood or some malleable metal to withstand the weight of the facial components and servo motors, given that the animatronics face will continuously be in motion. An important factor that will contribute to the selection of the materials is their availability and whether they are easy to handle and work with. A box of the same type of material will be added as a base to support the entire animatronics face.

Lastly, a simple battery holder of at most 9 V will likely be used to power the Arduino, the sensor, and the servo motors. The team is currently continuing research on what type and size of battery to use, as it will depend on the amount of power that will be drawn from the electrical components of the animatronics face.

For the duration of the project, the actual functions of the face will be the main priorities, while the aesthetics and beauty of the project are a secondary priority. By completing the plan outlined, the team will be able to create a fun, semi-realistic version of Dr. JoAnn Browning.

Section 3: Supporting Structure

To create the supporting structure, the team decided to use pieces of wood to manufacture a square frame to serve as both protection for the animatronics face and as a structurally sound foundation for the placement of the electrical components such as servo motors and sensors. The team brainstormed various ideas for the actual design of the supporting structure, and the most suitable design was the selected square frame due to the rigidity and stability that its base would provide to support the animatronics face and its mechanical components. The drawing for this design can be seen in Figure 4.

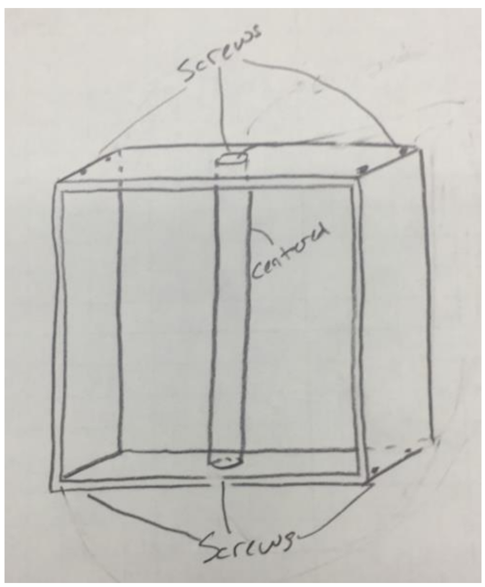

***Figure 4**. Drawing of Support Structure for the Animatronic Face*

After the shadow box type structure was complete, a supporting beam was cut out of a piece of door jam and screwed into the center of the box. This will allow the actual face structure to be mounted to a sturdy beam. Additionally, with a vertical bar, different components of the face can be mounted at different heights, such as the eyes closer to the top of the beam and the mouth near the bottom (Fig. 5).

***Figure 5**. Support Structure for the Animatronic Face*

The face for the Dean was constructed out of cardboard taken from a spare box. The original idea was to just print out the picture of the Dean's face and mount it onto a 3-D mask to give the face more depth and realness. This idea was put into action, but unfortunately, the way the face had to be cut to be placed onto the mask let many spots on the mask show through, which took away from the face representing the Dean. Not to mention, this version of her face was actually quite terrifying--not something you would want to show a room full of people and the Dean. Therefore, the team switched to mounting the picture of her face onto a cardboard cutout that is exactly the shape of her head. An exacto knife was used to precision cut the cardboard head and eye-shaped holes out of the face. The printed picture was then hot glued to this cut-out.

An L-shaped bracket was then mounted to the top, inside of the support structure to hold the face of the animatronics project. Additionally, another L-shaped bracket was mounted slightly below center on the support post to provide a mounting mechanism for other components of the face. This L bracket was then used to hold a popsicle stick, which was hot glued to the bracket, to allow for the servo motors controlling the eyes to be positioned (Fig. 6).

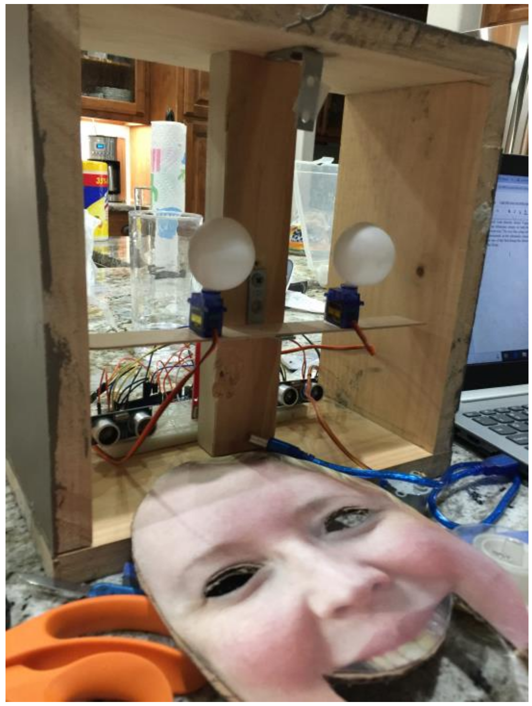

Figure 6. *The Supporting Structure with the Dean's Face*

During the interview with the Dean, the team inquired about what the Dean's favorite colors and patterns were. With Dean Browning's response being the classic orange, blue, and polka dots, the team thought it would be a nice touch to add that to the supporting structure for aesthetic appeal and to make the Dean even more beautiful (Fig. 7).

Figure 7. *The Paint Job for the Supporting Structure*

The hair of the Dean originally was supposed to have a 3-D voluminous effect from the curls made with curling ribbon. After over two hours of trying to implement this, the idea was scrapped as the hair made the Dean look like a completely different person and relatively creepy--none of the effects the team desired (Fig. 8). Therefore, a cut-out version of Dean Browning's hair from the photo she took was used and glued to the cardboard face. The final version of these mentioned support and aesthetic changes is seen in Figure 9.

Figure 8. *Curling Ribbon for Hair--Not the Best Idea*

Figure 9. *The Real Hair Look*

Section 4: Joints and Motors

In order to replicate the expressive nature of the Dean, the eyes of the animatronic face will move side to side to follow the motion detected by the ultrasonic sensors. To accomplish this, two servo motors were used--one connected to each eye. The eyes, hand-painted and modeled using a ping pong ball cut in half, were hot glued to the rotating arm of the servo motor. This mechanism allows the eyes to translate horizontally back and forth to mimic the look of eyes that track motion.

The eye servo motors were then hot glued to a cross bar that is perpendicular to the main support bar.

The mouth was actuated using a high torque servo motor. The motor was mounted to the base of the support structure with a slight lift in height using cardboard pieces and tape. A binder clip was hot glued to the back of the mouth to act as a connection point between the motor and the arm that facilitates the motion. To act as this arm, a paperclip was molded to fit through the hole of the binder clip and bent around the rotating component of the high torque servo motor. The combination of the paper clip and binder clip provided the transition from rotational motion to the final translational mouth movements.

The choice of servo motors dealt mainly with functionality, ease of use, and availability. The servo motors for the eyes were small and light to be easily maneuvered behind the face. The high torque motor was used because the motor would have to provide more force to raise and lower the cardboard mouth piece. Additionally, the high torque motor was used in place of a stepper motor, as the stepper motor was hard to program, finicky, and died shortly before the deadline of the project.

Section 5: Sensors

In order to bring to life the robotic impersonation of the Dean, the two types of sensors that were selected include two ultrasonic sensors and a radio frequency identification (RFID) sensor (Fig. 10). Each ultrasonic sensor is in charge of controlling the eyes' direction; for example, the right sensor when triggered causes both eyes to look to the right. By having these sensors coded as such, the Dean will be able to actively engage with her audience by following them with her eyes.

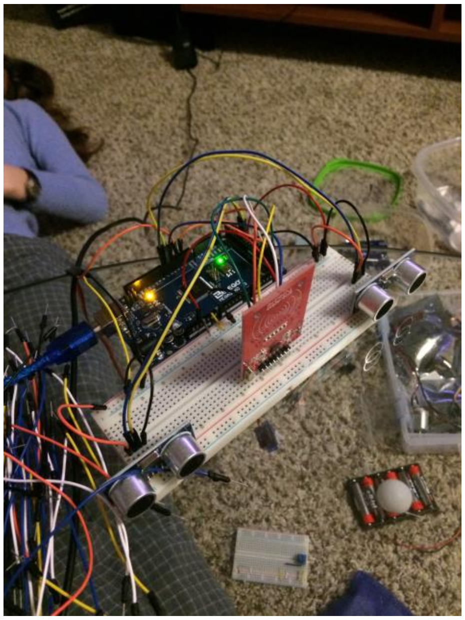

Figure 10*. Ultrasonic Sensors for Left and Right Eyes with RFID Sensor*

The chosen ultrasonic sensors operate by sending out a high frequency sound and timing how long it takes the echo of the sound to reflect back. When the sensor retrieves the sound back, the time it took to echo back can be converted into a specific distance the object it reflected off of is away. This type of sensor can be used in a variety of different applications, but for Dean Browning's case, she uses the sensor to trigger servo motion in her eyes if an object is detected within less than 20 cm of her sensors. This sensor is wired to an Arduino Mega, as seen in Figure 11.

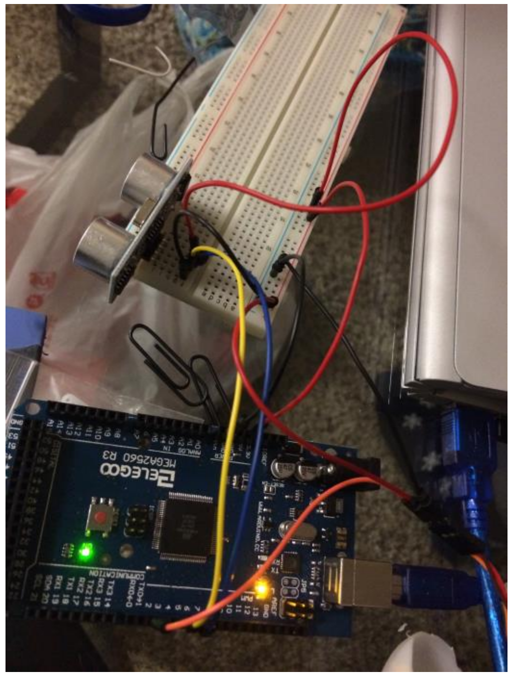

Figure 11*. Ultrasonic Sensor for Left Eye Control*

To even get Dean Browning to begin speaking, though, someone in the audience must have what we termed as "VIP" access. Obviously, not just anyone can talk to the Dean whenever they please, so only the select individuals with VIP access will be able get the dean to start talking and interacting with the audience. To achieve this, an RFID sensor will be utilized. The team decided that two cards would have access to activate the Dean's animations and voice. If an incorrect card is held in front of the sensor, the person will be denied access to the Dean. Once the correct access card is detected, Dean Browning will awaken, and her following sensors and animations will be activated. This includes her talking and the ability of her to attentively follow her audience.

The RFID sensor operates using radio waves to transfer data from a tag to a reader. Each RFID tag has an integrated circuit and antenna, which communicates with the RFID sensor. Each tag has a unique signal that the sensor can be programmed to read. By approximating a given tag to the sensor, the sensor will read if the card matches the required specifications matching that specified in the physical code of the RFID sensor. In the case of Dean Browning, once the sensor reads an access granted card, the Dean will begin interacting with her audience.

To accurately simulate the Dean speaking, the recorded soundtrack of the Dean's voice will be synchronized to the motions of the jaw in order to approximate the way a person is actually talking. This will be initiated as soon as the correct tag is placed in front of the RFID sensor and access is granted.

Lastly, the second joint that will be designated to move to make the animatronics face more realistic will be the eyes. As the Dean is a very respectable and educated woman, she will always look at the observer when she speaks, therefore, the two ultrasonic sensors that will be connected to the servo motors to control the eye movement will be triggered when the observer moves around the Dean. When the ultrasonic sensor that is placed on the left side of the Dean's facial structure is triggered, both servo motors will rotate the Dean's eyes 45 degrees to the left to turn and look at him or her. Similarly, when the ultrasonic sensor that is placed on the right side is triggered, both servo motors will rotate the eyes 45 degrees to the right. However, if both ultrasonic sensors are triggered, it means the observer is right in front of the Dean and thus her eyes will look directly ahead. Figure 10 depicts the general setup that will be utilized, featuring an ultrasonic sensor on both the left and right side of the breadboard and the RFID sensor in between.

Section 6: Programming for Interaction

The first interaction to program was the RFID sensor, as this is the very first interaction the audience will have with the Dean and the interaction that will trigger other interactions. To do this, the MFRC522 library needed to be downloaded, as that is the library associated with the RFID sensor. Before trying to code anything, an example called "DumpInfo" needs to be performed in order to obtain the access code associated with PICC card that came with the RFID sensor. Once the unique access code is obtained from the serial monitor of the example, this access code can be implemented into the actual code for the Dean, where a code from any RFID tag that was not equivalent to that of the associated PICC would cause the serial monitor to output "Access Denied." If the right PICC was sensed by the RFID reader, the code would print "Access Granted," and the first interactions with the Dean, including voice, mouth, and eye tracking, begin. Since this is the primary interaction, it is the statement used in the main void loop. Additionally, in checking for access, the code is written as a Boolean statement, meaning there is only true or false to the conditions presented within the Boolean statement [4].

The ultrasonic sensors placed on the base of the structure on both sides of the face will detect objects and cue the eyes to move in that direction. The code on the Arduino as one or both ultrasonic sensors are triggered displays a message stating the distance the object detected is from the sensor. For instance, a hand is placed in front of the right eye sensor and the Arduino will read "Object detected 20 cm away from right eye." This simultaneously triggers the servo motors and both eyes will turn and look to the right in this case. The same is the case for the left eye except for having the eyes turn to the left. However, if both or no sensors are triggered, the eyes will remain stationary and look straight ahead as the object or person detected should be standing in front of the Dean. The code on the Arduino was a sweep command so that when one sensor is triggered, the servo motors for the eyes will rotate the eyeballs attached to the eye joint horizontally 20 degrees to look right and 30 degrees to look left. The eyes will continue looking in the triggered direction and remain there until the object or person is no longer triggering the sensor, which will cue the servos to return the eyeballs to their original position of looking straight ahead. These ultrasonic sensors were coded using the NewPing library download, defining each respective sensor's echo and trig pins, and in the eyeBalls function through multiple if and else if statements. These statements were the part of the code responsible for providing the aforementioned eye movement.

To have the high torque servo motor actuate the jaw for speaking, the code added to the Arduino included the addition of mouthServo, which was used to call commands to that particular servo. This servo was the component featured in two different functions within the code--mouthUp and mouthDown. These two functions were responsible for moving the mouth up and down, respectively, and each was achieved with a for loop, calling the mouth to either raise or lower in steps of 5 with a slight delay of 10 milliseconds on each. This code triggered the motor to raise and lower the detached jaw for speaking purposes. The delay between starting and ending positions was very small in order to have this process happen at a fast-enough pace to more accurately simulate a person speaking.

The timing for the entire speaking interaction is triggered as soon as access is granted with an if else statement inside the original void loop. The general overview of what this does is once access is granted to the Dean, the code triggers the if portion of the void loop, which is only valid for 5.75 seconds, and the mouthUp, mouthDown, and eyeBalls functions are called. Once the 5.75 seconds are up, the else portion of the void loop is triggered and only the eyeBalls function is called. This 5.75 seconds was determined by the length of the complete audio portion the Dean says.

Lastly, developing the code for the voice command was probably the most rigorous portion of the code. The team opted not to use an MP3 shield or similar components that would add to the circuit of the Arduino and the main Arduino code. Instead, a backhanded way of playing the audio file was developed. After lots of research, it was determined that there was a way to somehow come up with a program to listen to the COM port on the Arduino and trigger an audio file to play through the computer when the external program "heard" a certain trigger phrase on the serial monitor of the Arduino. Many of the already built programs on the internet and Arduino forums for this had expired or their websites had been sold, leaving no program to accomplish the needed requirements until another programming open source software called Processing was found.

Through Processing, a Java based programming language, a tutorial for a similar function on the internet was found and the code was created and adjusted to fit the needs of Dean Browning. Since the team aimed to have the voice portion of the project at the very start of interaction, the trigger phrase the Processing program was listening for was "UID tag : 9C 80 53 73." This means that once the Dean senses the right access from this specific tag, she becomes available to talk with the VIP who scanned in and received permission to speak with her. Once the Processing program receives that trigger phrase from the Arduino serial monitor, it will trigger the imported audio file titled "DEAN_VOICE.wav" to play. This will happen simultaneously with the start and end of the mouth movements [5].

All the code utilized to create this project can be found within Appendix A. Additionally, the link to the YouTube video displaying the final working version of the Dean can be found using the link below.

Link to Dean Browning's Long-Lost Twin: https://youtu.be/iHOl3xcqUYQ

Section 7: Lessons Learned and Suggestions

7.1 Lessons Learned

1. The biggest lesson the team learnt, and hence, why it is number one on this list is to always save your files to more than one location. This may seem like common sense, but it is a lesson that this team overlooked until the flash drive had already corrupted every file on it, including all the Arduino code and every single simulation and drawing sheet for one senior design drawing package that was due three days later. If nothing else is learned from this report, please save the trauma and save any file to multiple locations.

2. It is important to have good friends with a variety of skills. For this team, one of the teammates had a roommate that was a computer science major. That was extremely helpful in troubleshooting any code errors, learning how to combine different parts of the code in one, and having a sound mind to consult when the team wanted the Dean to something particular.
3. Black magic within the code may just be possible. With very little knowledge on any coding before this, the team learned just what kind of miracles code is capable of, including listening, communicating, and triggering files to play on the computer from the Arduino.

7.2 Suggestions

1. Even though it is extremely difficult, try to allocate more time to not getting behind on this project and being proactive with the milestones. With all the other courses and work load the team had on their plate, the animatronics project ended up lower on the priority list earlier in the week, which led to late nights of extreme stress to try to pull off each milestone before the start of the lab. Even if it is just for an hour earlier in the week, that could save some future pain of procrastination.
2. When starting the project, it was an important step for the team to start by setting out the functions the face will do and labeling them as either mandatory or goal features. This is actually extremely helpful in the process of starting the face, because it is helpful in prioritizing throughout the project on where time is spent.
3. Always have a backup plan for the original backup plan for the actual plan. Knowing where to go when things inevitably go wrong or time is running low is always useful.

Section 8: Personnel and Bill of Materials

8.1 Personnel

The tasks for the project were split as shown in Table 1, but each team member contributed to every area of the project. The report writing, general troubleshooting, and idea generation was contributed to equally by each team member.

Table 1. *Personnel Task Distribution*

Task	***Main Personnel***	***Secondary Personnel***
Building the Supporting Structure	Victoria Wahlen	Jordan Traxler
Mounting the Components	Jordan Traxler	Victoria Wahlen
Research for Project	Sabrina Hamdan Shepard	Victoria Wahlen
Creation of the Code	Victoria Wahlen	Jordan Traxler
Physical Wiring and Integration to Arduino	Jordan Traxler	Sabrina Hamdan Shepard
Aesthetics and Beauty	Sabrina Hamdan Shepard	Victoria Wahlen

8.2 Bill of Materials

The costs for the project are reflected in Table 2, but these costs are solely based on material used for the project. The cost of labor was not added to this estimate due to the lack of funding in the $90 budget to pay three mechanical engineers.

Table 2. *Bill of Materials to Create Dean JoAnn Browning*

Item No.	*Description*	*Manufacturer*	*Quantity*	*Unit ($)*	*Total ($)*
1	Mega 2560 Super Starter Kit for Arduino Seesi	Seesi from Amazon	1	$64.50	$64.50
2	Wood and Screws for Structure	Jordan's Mom	N/A	$0	$0
3	Joints and Brackets	Jordan and Jordan's Mom	N/A	$0	$0
4	RFID Sensor and Keys	Seesi Kit	1	$0	$0
5	Ultrasonic Sensor	Seesi Kit	2	$0	$0
6	Servo Motor	Seesi Kit	2	$0	$0
7	High Torque Servo Motor	William Carrol	1	$0	$0
8	Mini-Breadboards	Seesi Kit	2	$0	$0
9	Breadboard	Seesi Kit	1	$0	$0
10	Arduino Mega 2560	Seesi Kit and Lab	1	$0	$0
11	Wires	Seesi Kit	N/A	$0	$0
12	Cardboard	Victoria's House	N/A	$0	$0
13	3-D Face Mask and Paint	Hobby Lobby	1	$3.89	$3.89
14	Paint for Supporting Box	Michael's	2	$4.99	$9.98
15	Paint Brushes	Jordan's Mom	3	$0	$0
Total Cost of Project					***$78.37***

Acknowledgements

This project would not have been possible without the help and assistance of the following: Austin Carpenter for coding troubleshooting and guidance, Angelica Traxler for the beautiful paint job on the outside of the Dean's case and the hospitality when the team was working on the project, William Carrol for the high torque servo motor a day before the project was due, and Sara Shepard de Hamdan for the dinner that carried the team through a long night before the project was due.

REFERENCES

[1] J. Brunken, “CoconutHead -- A Neophyte's Arduino-Based Animatronic Project,” YouTube, 05-Dec-2013. [Online]. Available: https://www.youtube.com/watch?v=CT9iG5XxCSw. [Accessed: 21-Oct-2018].

[2] Brainergiser. “How to make your own Animatronics Robot at home”. Youtube, 2 Jun, 2015. [Online]. Available: https://www.youtube.com/watch?v=2bN3ZSUr-QQ [Accessed: 21 Oct, 2018].

[3] Instructables.com. (2018). *Talking Animatronic Robot Head*. [Online] Available at: https://www.instructables.com/id/Talking-Animatronic-Robot-Head/ [Accessed 21 Oct. 2018].

[4] A. Mukherjee, “Security Access Using RFID Reader,” Arduino Project Hub, 07-May-2016. [Online]. Available: https://create.arduino.cc/projecthub/Aritro/security-access-using-rfid-reader-f7c746. [Accessed: 20-Nov-2018].

[5] T. Pulver, “How to Trigger an Audio File with an Analog Sensor Using Processing and Arduino,” GitHubGist, 07-Dec-2015. [Online]. Available: https://gist.github.com/timpulver/5ba4a29cddd543b4a900. [Accessed: 29-Nov-2018].

Appendix A: Codes

A1.1 Arduino Code

```
#include <SPI.h>
#include <MFRC522.h>
#include <NewPing.h>
#include <Servo.h>// include servo library

#define SS_PIN 53
#define RST_PIN 5
#define leftTrig 31
#define leftEcho 33
#define rightTrig 7
#define rightEcho 6


MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create MFRC522 instance.
NewPing leftSensor(leftTrig, leftEcho, 100);
NewPing rightSensor(rightTrig, rightEcho, 100);


Servo rightEyeServo;// declare servo name type servo
Servo leftEyeServo;
Servo mouthServo;
boolean accessGranted = false;
unsigned int startTime;
void setup()
{
  Serial.begin(9600);   // Initiate a serial communication
  SPI.begin();      // Initiate  SPI bus
  mfrc522.PCD_Init();   // Initiate MFRC522
  Serial.println("Approximate your card to the reader...");
  Serial.println();
  rightEyeServo.attach(2);// attach your servo
  leftEyeServo.attach(3);
  mouthServo.attach(25);
  mouthServo.write(45);
  // put your setup code here, to run once:
}

void loop() {
  if (!accessGranted) {
    accessGranted = checkAccess();
  }
  else {
```

```
    unsigned int curTime = millis();
    if (curTime - startTime < 5750) {
      eyeBalls();
      mouthDown();
      mouthUp();
      delay(50);
    }
    else {
      eyeBalls();
    }
  }
}

boolean checkAccess() {
  // Look for new cards
  if ( ! mfrc522.PICC_IsNewCardPresent())
  {
    Serial.println("No card present");
    delay(900);
    return false;
  }
  // Select one of the cards
  if ( ! mfrc522.PICC_ReadCardSerial())
  {
    return false;
  }
  //Show UID on serial monitor
  Serial.print("UID tag :");
  String content= "";
  byte letter;
  for (byte i = 0; i < mfrc522.uid.size; i++)
  {
     Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
     Serial.print(mfrc522.uid.uidByte[i], HEX);
     content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
     content.concat(String(mfrc522.uid.uidByte[i], HEX));
  }
  Serial.println();
  Serial.print("Message : ");
  content.toUpperCase();
  if (content.substring(1) == "9C 80 53 73"|| content.substring(1) == "E5 AE 4F 26") //change here
the UID of the card/cards that you want to give access
  {
    Serial.println("1200");
    startTime = millis();
    return true;
```

```
  }

 else   {
    Serial.println(" Access denied");
    return false;
  }
}

void eyeBalls() {

  unsigned int distanceLeft = leftSensor.ping_cm();
  unsigned int distanceRight = rightSensor.ping_cm();

  Serial.print("Left eye: ");
  Serial.print(distanceLeft);
  Serial.println();

  Serial.print("Right eye: ");
  Serial.print(distanceRight);
  Serial.println();

  if (canSee(distanceLeft) && canSee(distanceRight)) {
    Serial.println("Both eyes");
    leftEyeServo.write(90);
    rightEyeServo.write(90);
  }
  else if (canSee(distanceLeft) && !canSee(distanceRight)) {
    Serial.println("Left eye only");
    leftEyeServo.write(70);
    rightEyeServo.write(70);
  }
  else if (!canSee(distanceLeft) && canSee(distanceRight)) {
    Serial.println("Right eye only");
    leftEyeServo.write(120);
    rightEyeServo.write(120);
  }
  else if (!canSee(distanceLeft) && !canSee(distanceRight)) {
    Serial.println("None");
    leftEyeServo.write(90);
    rightEyeServo.write(90);
  }

  delay(200);

}
```

```
boolean canSee(int distance) {
  if (distance > 0 && distance <= 20) {
    return true;
  }
  else {
    return false;
  }
 }

void mouthDown() {
  for (int i = 45; i >= 0; i=i-5) {
    mouthServo.write(i);
    delay(10);
  }
}

void mouthUp() {
  for (int i = 0; i <= 45; i=i+5) {
    mouthServo.write(i);
    delay(10);
  }
}
```

A1.2 Processing Code for Audio

```
import processing.serial.*;
import ddf.minim.*;

Minim minim;
AudioPlayer player;

int lf = 10;    // Linefeed in ASCII
String myString = null;
Serial myPort;  // The serial port
int sensorValue = 0;

void setup() {
  // List all the available serial ports
  printArray(Serial.list());
  // Open the port you are using at the rate you want:
  myPort = new Serial(this, Serial.list()[0], 9600);
  myPort.clear();
  // Throw out the first reading, in case we started reading
  // in the middle of a string from the sender.
  myString = myPort.readStringUntil(lf);
```

```
  myString = null;
  // we pass this to Minim so that it can load files from the data directory
  minim = new Minim(this);
  // loadFile will look in all the same places as loadImage does.
  // this means you can find files that are in the data folder and the
  // sketch folder. you can also pass an absolute path, or a URL.
  // Change the name of the audio file here and add it by clicking on "Sketch —> Import File"
  player = minim.loadFile("DEAN_VOICE.wav");
}

void draw() {
  // check if there is something new on the serial port
  while (myPort.available() > 0) {
    // store the data in myString
    myString = myPort.readStringUntil(lf);
    // check if we really have something
    if (myString != null) {
      myString = myString.trim(); // let's remove whitespace characters
      // if we have at least one character...
      if(myString.length() > 0) {
        println(myString); // print out the data we just received
        // if we received a number (e.g. 123) store it in sensorValue, we sill use this to change the background color.
        try {
          sensorValue = Integer.parseInt(myString);
          // As the range of an analog sensor is between 0 and 1023, we need to
          // convert it in order to use it for the background color brightness
          int brightness = (int)map(sensorValue, 0, 1023, 0, 255);
          background(brightness);
        } catch(Exception e){}
        if(myString.equals("UID tag : 9C 80 53 73")){
          if(player.isPlaying() == false){
            player.play();
          }
        }
      }
    }
  }
}
```