Sculpture to show life purpose quotes

“We learn more by looking for the answer
to a question and not finding it than we do from
learning the answer itself.”

~ Llyod Alexander

Written by: Tom Morsink
Date: 16-08-2017
Graduation project for the bachelor Creative Technology

Table of content

"The wisest mind has something yet to learn."
~ George Santayana

Introduction

The project of the "life quote sculpture" was an idea from Clemens Mensink who founded Menperium. To get this idea to fruition he presented his idea to the study Creative Technology as a project that could be used as a bachelor assignment. This was approved and an assignment was drawn up. The assignment was to create a sculpture that displays quotes and was preferred to be kinetic. On top of designing and creating the sculpture, a literature review based on different elements that could be used for the sculpture needed to be carried out based on a research question concerning how the sculpture could be most effective in getting the quotes across to people.

Stakeholders

There are two main stakeholders in this project. These stakeholders are introduced below.

Menperium

Menperium is a small company founded by Clemens Mensink. Its purpose is to get people to actively think about why they do things. What drives them. And in this way help people grow and get the best out of themselves. One of the ways they want to accomplish this is through this project of the quote sculpture. The main objective and concern of the client is a well-designed and working sculpture.

Creative Technology

The bachelor creative technology, or CreaTe for short, is a study at the University of Twente. It focuses on the use of existing technologies in new and innovating ways. The study has a focus of prototyping and creating the designs one can come up with. It therefore was the perfect study to take on this assignment. As it is an academic study, CreaTe is also concerned about the way a design and prototype came to be and how the research question(s) are tested. The main objective of the study is to test if the student working on this project has taken the proper scientific steps to complete this assignment and is ready to finish the bachelor program.

Motive

Over the last couple of years Menperium has gathered several hundreds of inspiring quotes. These quotes are meant to motivate people to think about the essence of a concept or product as well as the core values in one's personal life. To bring these quotes to the attention of people, Menperium wants to create a (kinetic) sculpture that displays these quotes. This sculpture is meant to be placed in an entrance hall or a common gathering ground. In other words, a spacious place where a lot of people come and go every day.

The challenges of this assignment are how to attract and guide the attention of people towards the quotes that are displayed and the design and implementation towards a finished product within the allotted time.

Not only can this sculpture be used to help people think about their lives, which is more the focus of the quotes themselves and less that of the sculpture around it, the sculpture can also be used as a tool for testing what kind of behaviour attracts the attention of people. What can attract attention when people are not searching for it? What stands out? What doesn't? By creating this sculpture tests concerning these questions can be done and might lead to new insights or support existing ideas.

Objective

To grab people's attention and draw it towards the quotes a design with these principles in mind must be made. The focus should naturally lead towards the screen with the quotes, but the sculpture as a whole must be interesting enough to make people want to take a closer look. This can be done by combing shapes, colours and motion into an appealing whole.

"The wisest mind has something yet to learn."
~ George Santayana

Research question
To reach the desired result the following research question was formulated.

> How can a sculpture guide the attention of people to a specific point where a quote is displayed?

To answer this question several sub questions were formulated to support the main research question. These sub questions are:

1. How can colour affect the attention of people?
2. How can shapes affect the attention of people?
3. How can motion affect the attention of people?

State of the art

Literature review
How colour affects attention

Literature shows that more reddish colours attract attention, but this in contrast with the context and dependant on the emotional load of the object or image. For instance, Kuniecki et al. [1] state that a target was identified quicker when it was congruent with a red cue and was identified slower when it was incongruent with a red cue. It was also seen in this experiment that the emotional content of the cue significantly affected the reaction time. Which indicates that a red image with an emotional charge attracts attention?

The study of Pal et al. [2] shows that among warm colours the more reddish colours attract attention. They state that: "distribution of chromatic features (both hue and saturation) corresponding to salient warm colors and non-salient warm colors are different. Moreover, if there is competition between multiple warm colors then the colors with relatively low hue (closer to red) and/or high saturation are more likely to attract our vision." [2]. In other words, in contrast to the surroundings the more reddish colours are more likely to grab the attention.

How shape affects attention

According to Bravo and Nakayama [3], individual shapes don't affect attention significantly, unless people already know what shape they should be looking for. This means that shapes in and on themselves don't hold much value for attracting attention, however size does matter. According to Harris et al. [4], the size of an object relative to its surroundings does have an influence on attention. When an object is big in comparison to its surroundings it attracts more attention.

How motion affects attention

Hillstrom and Yantis already challenged the idea that motion captures attention in 1994. According to their research, motion itself does not necessarily capture attention. What they think captures the attention is "new things". In their research they ruled out the possibility that motion captures attention with an experiment where motion was a distractor and it didn't capture attention. This idea can be supported by the research of Becker and Horstmann [6]. According to them, motion itself does not capture attention. However, if the motion is novel or unexpected, it does capture the attention.

State of the art

Inspiring sculptures

There are a lot of inspiring sculptures to be found. Many carry a strong believe or strong image depicting virtuous behaviour or great accomplishment. The form of these sculptures can range from minimalistic designs to intricate and complex sculptures. One such an example can be found below, in figure 1. It depicts a man sculpting himself from raw material. There is a strong believe behind this sculpture, namely that a person can be anything he or she wants to be if you're willing to work for it.

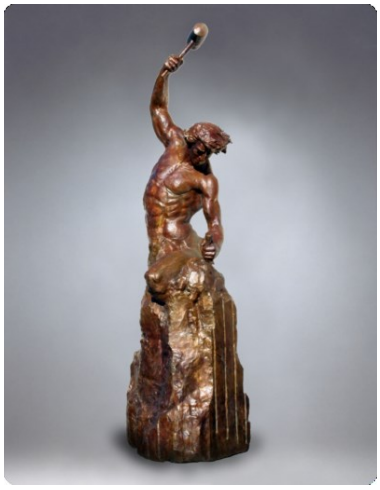

Figure 1. self-made man by Bobbie Carlyle

Figure 2. Girl defying the wall street bull

Another example of a strong inspirational sculpture is the "girl defying the raging bull" on Wall Street as seen in figure 2 above. Thought it was a temporary statue to celebrate international women's day it is still expressing a strong message. At the girls feet is a plaque saying "Know the power of women in leadership – SHE makes a difference". One to support gender equality.

Kinetic sculptures

There are also plenty of kinetic sculptures in this world. These are usually more abstract and more focused on the art of engineering. One famous example is the "strandbeest" from Theo Jansen as can be seen in figure 3. A very intricate piece that walks by the power of the wind. It doesn't communicate a very clear message, but it is inspiring none the less. This is because it is a cleverly designed structure which moves without any motors.

Figure 3. Strandbeest from Theo Jansen

Figure 4. Infinity by David C. Roy

A second example is the Infinity sculpture by David C. Roy (figure 4)[7]. This is also a more abstract piece of art that attracts attention because of its intricate yet simple movements and optical illusions.

Quote sculptures

There are many wall hangings and other forms in which inspirational quotes have been incorporated to make an inspirational piece. And there are, of course, plenty of sculptures with a plaque with a supporting message to help empower the idea that it is supposed to communicate. However, an extensive search for a sculpture that incorporates quotes or a quote into the essence of its design I came up empty. There were no sculptures to be found that lead you towards reading an inspirational quote, which means that the idea of a sculpture that inspires not just by its form but also by several different inspirational quotes is a novel one.

Conclusion

Looking at the state of the art and literature review it is clear that motion can be a big contributor to attract attention. Furthermore creating a sculpture incorporating some form of mechanical construction which creates motion is rather intriguing. Using a reddish colour could also help in attracting attention, but seems to be less important than motion. Thirdly, the shape of an object doesn't seem to matter when there is no internal motivation to look for a shape. However it should be noted that the relative size does matter. Lastly, although inspirational quotes are a growing trend, there seems to have been no endeavour yet to incorporate inspirational quotes into a sculpture that switches between these quotes. Considering these points the most important thing to keep in mind during the design phase is the motions of the sculpture.

Concept design

Since the sculpture needs to attract attention it is important to keep in mind the context in which the sculpture will be placed. Therefore a clear description of the context is given here. The sculpture should be placed in a large open space like an entrance hall. A place where a lot of people pass by. This also means that the people who pass by are probably in a bit of a hurry to get somewhere. This means the sculpture needs to be as noticeable as possible and have a big attraction of people's attention.

Requirements

To form a good concept design, the first thing that needs to be done is set some requirements. These requirements can be formed by looking at the request from Menperium and their requirements, as well as taking into account the results of the literature research. With these factors in mind a list of requirements was made. It is divided into four categories (must have, should have, could have and won't have) based on the feasibility and importance of the specific characteristics. The list of requirements can be found below in Table 1.

Must have	Should have	Could have	Won't have
It must be suitable to place in a large open space	It should be a kinetic sculpture	It could be an interactive installation	It won't have advanced sensor technology
It must display quotes	It should have unexpected movements in it	It could use external information to decide what category is most appropriate to show	
It must regularly change quotes	It should be big in comparison to its surroundings	It could have reddish colours in it to attract more attention	
	It should have different categories of quotes		

Table 1. List of requirements

Users

Though it would be nice to say that the intended audience for this sculpture is every by passer that comes across it, this is not entirely realistic. The first reason for this is that the quotes are all in English. Therefore, the intended audience of the sculpture is those who have a good understanding of the English language. Secondly, the quotes are meant to get people to think critically about themselves and their actions. This can best be achieved when the target audience is able to reflect on themselves and their own actions. These two factors combined form the target audience. These factors are most common in those of higher education, so this shall be the target audience.

Initial designs

Taking into account these requirements some initial sketches were made. These sketches can all be found in Appendix A. In all the initial sketches I tried to convey a certain message. This message was one of wisdom and knowledge that is present in all of the inspiring quotes. The first concept was one of an atom (figure 5). The idea behind it was that all the quotes had a core of truth in them and that the quotes would be displayed on a monitor that was on the core of the atom. Around the core, a track would be hung which would be the track for the "electron" to follow. Two ideation forms were made for this track. One form was by the use of a LED-strip which would shoot a light through itself acting as an electron as depicted in figures 5.1 and 5.2. The other option would be an actual track with a sort of train riding around on the track acting as electron as depicted in figure 5.3. There were also two ideation forms for the core of the atom. One with just one smooth sphere with a screen on it, as seen in figures 5.1 and 5.2, and another one where the neutrons and protons would imagined by using several spheres clustered together as can be seen in figure 5.3. Lastly, figure 5.4 shows two ideas of how the atom would be placed, on a pedestal or hanging from the ceiling.

The second idea was a big mouth that would close, then change the displayed quote and open again (figure 6). The idea behind this concept is the idea that these quotes deserve to be "heard" as they contain a life lesson from the person who once uttered them.

Figure 7 depicts an owl with on its chest a screen that displays the quotes. As the owl is a symbol of wisdom in many cultures it would be a nice idea if such a symbol would show the quotes as little pieces of advice. While discussing this concept the idea came forward that the owl could maybe open and close its wings to make it a more interesting sculpture.

Figure 8.1 and 8.2 show two sketches of an eye. The eye, just like the owl, is in many cultures a symbol, a symbol of enlightenment in this case. The idea is that the eye shows interest in the quotes displayed and "reads them" from the screen.

Figure 9 depicts a book on which the pages display a quote. The idea behind the book is that a book contains valuable information and that it is meant to share its knowledge with the rest of the world.

Figure 10 shows a person lost in thoughts. These thoughts are the quotes which are displayed on a screen at the top of his head, so that people can read the thoughts of this person.

Figure 11 shows an interactive machine. The idea is that you need to turn the wheel and then you can watch the gears work. After a short while turning the gears you would get a dramatic changes on the screen and a new quote would be displayed. The idea was that these quotes are piece of wisdom acquired after working hard to accomplish a certain goal.

Figure 12 shows a brain that is chained down and locked to its own confines. Every time a new quote would be displayed the brain would rattle its chains as if it was trying to break free. This to show that new thoughts and ideas can change your understanding of the world and free your mind of its previous misconceptions.

Figure 13 depicts a giant 42 with a marble track in it. The idea behind the 42 is a reference to a popular book series, the hitchhikers guide to the galaxy, by Douglas Adams. In the books a supercomputer, deep thought, was built to reveal the answer to life, the universe and everything. The answer that came up was 42. The marble track inside the big 42 was meant purely as an attention attractor and doesn't have any meaning in and on itself.

Figure 14 displays a chart where the legend is a screen displaying the quotes. The idea behind this is that these quotes can help guide your way on the path of life.

These ideas were all presented to Menperium and individually discussed. During these discussions the ideas behind the designs were shortly explained and evaluated through open discussion. After every design was individually discussed, the designs were compared to each other. Two designs stuck out in this discussion. The design with the gears (figure 11) and the brain in chains (figure 12). The gears stood out because we believed the motion it shows would be quite intriguing and probably attract attention well. As for the brain, the idea of chains becoming undone with every new quote was a very appealing idea. The brain itself, however, would probably be a bit too graphic to put in an entrance hall. While discussing these designs a new idea sprouted. This was one of a man in chains trying to break free with each new quote or the idea that a figure of a man could be made with gears to show that these quotes got something moving inside of people when they read them. These ideas were roughly sketched on the spot and can be seen in figure 15.

When comparing these initial designs to the requirements set earlier, we can conclude that all design have the potential to fulfil the must have requirements. Looking at the should haves and could haves most of the other requirements can be incorporated into each design. However the requirements for motion and especially the unexpected motions are restricting the choices considerably as these cannot easily be worked into most designs. Keeping these two requirements in mind and after the discussions the idea of a "gear man" came out as a favourite. The reason for this being that it could easily be made to move and even so in unexpected and novel motions.

Concept development

After discussing the initial design sketches with Menperium one of the designs was chosen to further develop, namely the idea of the gear man (figure 13). Some more ideation sketches were made to play with the idea and see what would work well. All sketches of the concept development and detailed design can be found in appendix B. The first iteration of the final design can be seen in figure 16. It shows a gear sculpture in the rough form of a person. The gears in this sketch are attached on the surface of a plate. The plate is also meant to hint more obviously to the form of the sculpture to represent a person. This was also used in the second iteration as can be seen in figure 17. The main difference between these two iterations is the mechanism used to turn the head. In the first iteration this was just a normal spur gear, but in the second iteration this was chosen to be a toothed belt driving a mechanism behind the head. This in order to give an even clearer silhouette of a person. Both the first and the second iteration also have some figures on

their heads. The idea was that these figures could show the current category of quotes. To give the sculpture more body two sketches from a top view perspective were made. These can be found in figure 18. The idea was to create more depth in the sculpture and make it more appealing from the sides. To do this more plates with gears on them would be placed in either a triangle or square form. Lastly a third iteration was made where the gears wouldn't be on plate, but would be mounted on a skeleton that would be hidden by the gears. This would create a more abstract silhouette of a person. This version was made with the thought that a more abstract silhouette could maybe attract more attention by being less obvious and making people take a better look before coming to the realisation that it is the silhouette of a person they are looking at. All of these iterations have a pc monitor, or screen, hanging at around eye level showing the quotes. The choice to display the quotes on a screen stems from the fact that the quotes need to change regularly and the easiest way to do this is through a digital image. Furthermore, with a screen there is a lot of freedom for digital designs or animation. Lastly, a screen can easily be made a part of the sculpture.

These ideation sketches were also discussed and reviewed to choose a final design. In the end the third iteration was chosen. The final design would thus look like a silhouette of a man made out of gears and a screen (figure 19). The reasons for this being that a more abstract sculpture would probably be more intriguing and the focus would be purely on the gears and the screen. An important detail is that the middle of the screen is at about 1.70m in order to put it on about eye level. This way people would automatically notice the screen when looking at the sculpture.

When taking a look at the requirements again, it can be seen that the most important requirements are met (Table 2).

Must have	Should have	Could have	Won't have
It must be suitable to place in a large open space	It should be a kinetic sculpture	It could be an interactive installation	It won't have advanced sensor technology
It must display quotes	It should have unexpected movements in it	It could use external information to decide what category is most appropriate to show	
It must regularly change quotes	It should be big in comparison to its surroundings	It could have reddish colours in it to attract more attention	
	It should have different categories of quotes		

Table 2. checking the requirements. Green = requirement is met, yellow = requirement can be met, red = requirement isn't met.

Detailed design

For the practical implementation a method of getting gears was required. There were several options to be considered, all with their benefits and draw backs. Since the design required some large diameter gears one of the first option, to buy them, was practically shot down in the beginning, as these gears would've had to be custom made. This would mean they would be very expensive. Another option was to 3D print the gears. The advantages of this method would be the high modifiability of the design and the cheap costs. However, the major drawback of this would be that it would take a lot of time to print the required amount of gears. The last option considered was to use a laser cutter. This would also mean a high modifiability of the design and cheap costs, and it would be faster than 3D printing the gears. The restraints of laser cutting would

mainly be the size at which it could cut and the specific material. After careful consideration laser cutting the gears was the preferred method to get the gears.

Bearing in mind the practical constraints, the final design was redrawn digitally as a vector file to get the positioning and size of the gears and the screen right and have a nice overview of how it would look from the front. In figure 20 in Appendix B the digital design can be seen. Using this image a 3D model was made using Fusion 360. A link to the fusion file can be found at [8]. In the model the gears were generated by a plugin of fusion 360. The plugin had several different types of gears, but the most appropriate type for this model was the spur gear. After choosing the type of gear the module and the amount of teeth had to be put in and a gear was generated. To make sure all gears would work with each other the same module, 10, was used for all gears. This module was also chosen as it was rather large and convenient to work with. In total 16 gears were placed in the model with four different sizes. The different sizes were determined by the amount of teeth on a gear (as they all had the same module). The four different gears had either 28, 23, 17 or 11 teeth. These numbers were chosen for two reasons. The first reason is the size. By setting the largest gear to 28 teeth the maximum diameter of the laser cutter (30cm) and the biggest diameter of a gear (30cm) would be the same and thus optimized. The second reason is for the wear on the gears. By taking a number of teeth that ensures a relative prime when meshing together. This is created when the greatest common divisor of the gear teeth is 1. With the amount of teeth for each gear now chosen this holds for every combination of gears within the model.

In the model all gears mesh together, however in reality this won't be the case. To create an unexpected motion in the sculpture, the lining of the gears will be just off for some of them, which creates different groups of gears. This means these groups of gears can be controlled separately by different motors. With this the speed and direction of gears can be different from the other gears while they initially look like they should connect. By incorporating this optical illusion you can create an unexpected motion which, as the literature review concludes, would draw attention. It will probably lead to people taking a closer look to see why their expectation of how the gears should move isn't what is actually happening.

During the final design process some choices were made concerning the backend programming of the sculpture that influences the behaviour of the sculpture. As stated in the requirements the quotes will be split up into categories. These categories are Happiness, morals, personal growth, success and leadership. The categories are used to give the different quotes that are on display a bit more cohesion. In this design the categories aren't made visible, but the change is made visible through motion. The choice was made to let the quotes and the category change with regular intervals. Also the choice to not make the sculpture interactive was made. It was instead decided to let the category change based on the news. To do this news headlines from a site will be checked for topics and based on what is most prominent the category will be picked accordingly. Also, at the end of the design phase it was decided that building a finished product wouldn't be possible in the remaining time of the project. Therefore it was decided that a prototype should be built as a proof of concept. This prototype would have the same functionalities but would be built with cheaper and easier to use materials. The conclusion was that wood would work best for a prototype as it is cheap and easy to work with.

The final design will be a gear structure in the form of a person and will have a screen displaying quotes on its chest. This screen will be at eye level to naturally guide the attention towards the screen. The gears will be moving constantly and at the change of a quote or at the change of the category the gears will perform a patterned motion. This motion speeds up or slows down the turning of the gears and changes the turning direction. After the pattern is completed the gears will again move as they had before the start of the pattern. Lastly it will base the category on the headlines in the news.

Method

To help design a good sculpture a research question and some sub questions were formulated in the beginning of the project. To answer the sub questions a state of the art review has been done and gave some insight in how to make a sculpture that attract attention. Now it can be tested if the findings of the state of the art review were correct. To do this the research question and sub questions should be tested with an experiment.

To test the design with an experiment a method to do so must be set up. Taking into account the literature research and the design of the sculpture the method was devised. The sculpture will be placed at an entrance hall, in this case Hal B at the University of Twente. It will be on display and while it is on display the number of passersby will be counted. Not only the total amount of passersby will be counted but also the number of people who clearly looked at the sculpture while passing it will be noted and the number of people who actually stop to look at the sculpture will be noted down. This can give a good indication how well the sculpture attracts attention in a bottom-up way. In other words, when there is no internal motivation to actively search for the sculpture, but it is noticed due to its own characteristics. Figure 21 shows how the test setup was placed in Hal B.

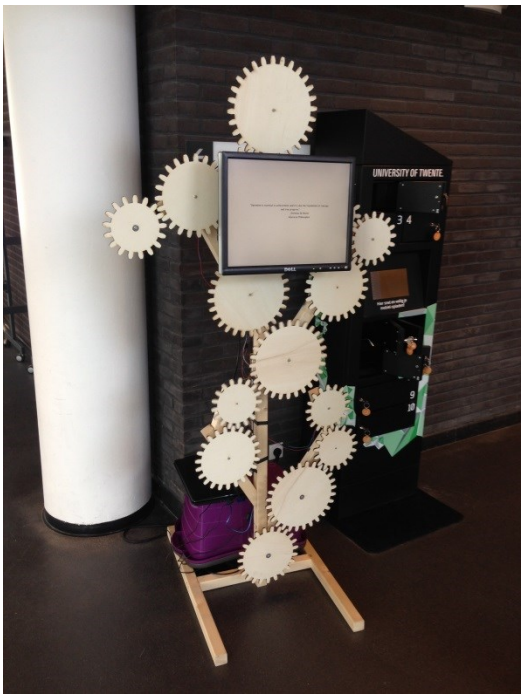

Figure 21. The test setup in Hal B.

To determine which element of the sculpture attracts the most attention three different test set-ups were devised. Based on the literature research the three set-ups were as follows. The first set-up would only display the quotes on a screen while the sculpture itself would be idle. This creates a good measure for the second and third set-up. In the second set-up the sculpture will move with a continuous motion of the gears and display the quotes while in the third set-up the gears will have sudden changes in motion when the quote or the category changes. With these two set-ups the notion of motion attracting attention can be tested. If motion attracts attention then both the second and third set-up would probably attract the same amount of attention. But if only new or sudden motion attracts attention the third set-up should attract more attention. During these tests some comments on observations will also be noted down to help explain the outcome. The test setups mainly check the third sub question, the one about motion, because of the practicality of the tests and because of the findings of the literature research which state that motion is probably a significant part of attracting attention when it's novel or unexpected.

Implementation

The implementation phase consisted mainly of writing the backend of the programs that would run the sculpture and building the sculpture itself.

Programming

For the programming of the sculpture two programs were used, namely processing 3.3.5 for the computer side and Arduino 1.6.5 for the microcontroller. The code written for both programs can be found in appendix C. Both are commented and should be clear to understand.

Arduino code

The Arduino code, which can be seen first in the appendix, is fairly simple. This is partly because Arduino has a stepper library, which makes controlling the stepper motors fairly easy. All you have to do is initiate the stepper and set the speed using stepper.setSpeed(). Then, using the stepper.step() function, you tell the stepper to set as many steps as you specified in the function, and the stepper will set these steps at the speed of the setSpeed function. The Arduino code itself does not determine the behaviour of the gears in terms of timing. It only sets the speed en turns the gears according to the current situation. This current situation is determined in the processing code, which will be explained later on. To set up the Arduino to listen, a serial port is opened at a baud rate of 9600 by the serial.begin function. The Arduino is then ready to receive data through bytes. The incoming byte is the stored in a variable. This variable is then used in the setMotorSpeed function to determine if the motorSpeed and the direction of the motor should be changed. This is done through a series of if loops.

Processing code

The processing code is more complicated compared to the Arduino code. It has a lot of different functionalities and is coded in different tabs in order to keep the code as neat as possible. To keep things clear each tab will be described separately.

In the first tab, quoteDisplay, are the main functions, the setup and draw functions, as well as the function to write the quote on the screen and some of the global variables. In the first tab is also a keypressed function that helped with debugging and testing the code. It was left in there as an easy way to test any improvements. In the setup the program window is set to full screen and a font is loaded into the program. Furthermore the TSV file containing the quotes is loaded into a table. Then a serial connection with the microcontroller is set up at a baud rate of 9600 and the countQuotes function (explained in tab 2) is called. Lastly in the setup the first category is determined by calling the determineCategory function (explained in tab 3). In the draw function the first thing that is done is checking if in the previous draw cycle the quote has changed. If that is the case it calls on the motor control function (explained in tab 5) and sets the Boolean to check if the quote has changed back to false. The reason for this will be explained later on. The draw function also has a timer programmed into it using the millis function. By using the millis function like this one can create a timer which does not pauses the program, but only checks whether or not the specified time has passed or not. If the timer exceeds a certain time the get quote function is called upon and the timer is reset. Lastly the determineCategory and writeQuote functions are called upon in the draw function. The writeQuote function in the first tap draws a new background and the current quote. The quote is positioned using the text function from processing.

In the second tab, count_quotes, the table is analysed on how many quotes per category there are. This is done with a simple for loop. The count is stored in global variables used in other parts of the program. This function is quite useful as it makes it possible to changes the TSV file containing the quotes without having to change the code of the program. This is, of course, on the note that the structure of the TSV file is kept the same.

In the third tab, determine_category, is the code to determine the category based on news headlines. It will check these headlines every day at 8:00 and 13:00 o'clock. At those times there will first be a 1 second delay in the program in order to avoid changing the category more than

once after which the getHeadlines function is called. In order to get the news headlines a separate library for processing needed to be added. This library is Jsoup. Using Jsoup, a webpage's html code can be scraped and parsed into strings. By selecting the appropriate web address (for this project www.nu.nl) and path in the html code the headlines can be parsed into a string. This string can then be split at the spaces into a string array containing all the words that make up the headlines. This array is then compared with five category buzzword arrays to determine which category is more prominent in the news. In table 3 an overview of the buzzwords can be seen. The buzzwords are in Dutch because the website that is scraped is a Dutch news website. When the most prominent category is decided the variable containing the current category is set to it. Lastly the motor control function for the new category is called.

Happiness	Leadership	Morals	Success	Personal growth
gelukkig	leider	goed	vooruitgang	verbetering
blij	president	slecht	succes	eigen
positief	CEO	aanslag	overwinning	persoonlijke
beter	leiderschap	terreur	winnen	ontwikkeling
goed	beleid	moraal	beleggers	hobbels
blijdschap	aansturen	verijdelt	winst	tegenslagen
geluk	premier	justitie	omhoog	tegenslag
			stijgen	groei

Table 3. buzzwords per category

In the fourth tab, get_quote, the new quote to be displayed is determined. This is done by creating a random ID number. This ID number is created by a letter determined by the current category and a randomly generated number. The ID is then checked against the previous ID. If it is the same then the ID goes one number up, or if it is the last ID number go down by one. This is in order to ensure a new quote is displayed and not the same one. Then, if the ID is good to go, the quote variables are changed to the new quote determined from the table.

In the fifth tab, motor_control, the serial library from processing is imported. When the new quote motor control is called upon it first runs the determine category function to minimize the chances of missing it. This is because the motor control will delay the program for a few seconds in order to send a fixed pattern of bytes to the microcontroller. This pattern determines the behaviour of the sculpture and can be changed fairly easily. These delays are also the reason for the Boolean that checks if the quote has changed in the first tab. The new quote won't be drawn until the end of the draw function. And as the motor control function is in between there and delays the whole code the gears would turn before the new quote is displayed if it were not for this construction. This tab has also a new category motor control function which determines the pattern for the gear motions and works in the same way as the new quote motor control function.

Wiring

The wiring for the sculpture consists of 5 wantai 42byghw208 bipolar stepper motors, 5 sn754410ne h-bridges, an Arduino mega and two 12V 3A DC power supplies. And a lot of wires. To connect everything two breadboards were used. This way all connections are clear and can be easily expanded on. The complete schematic made using fritzing can be found in appendix D figure 22. It shows how the five steppers are connected to the Arduino. It is basically connecting 5 steppers as shown in figure 23 except for the potentiometer. The datasheets for the stepper motor and the H-bridge can be found at [10] and [11], respectively.

Although the steppers should only take about 0.5A nominally and thus only one power supply should suffice, there were some problems with steppers not turning even though they should be connected correctly. For this reason a second power supply was placed which seemed to help a bit, but unfortunately didn't solve the problem. Three steppers had trouble turning even

though they were connected properly and had sufficient power. This was even the case when all wiring had been thoroughly checked and had been redone several times. Even attempts at connecting the steppers in any way that could be right did not solve the problem. Even the output pins on the Arduino were changed to see if that would help, but it didn't. In other words, the reason for these stepper motors not working wasn't found.

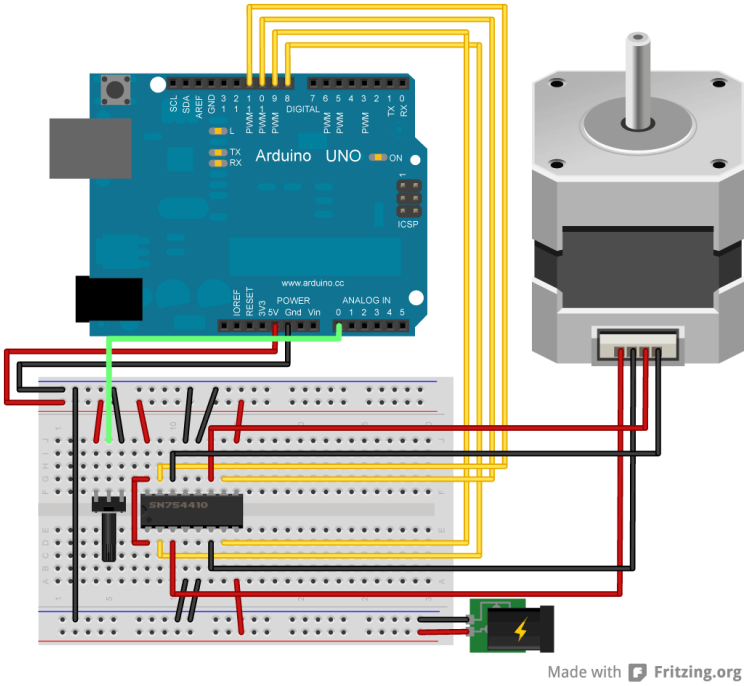

Figure 23: how to connect a stepper using an H-bridge[9]

Building the sculpture

As it was decided to do a prototype of the sculpture instead of a finished product, some freedom was taken while building it in order to make it work. Though this is hardly noticeable when looking at it from the front. Mostly these freedoms were in the frame and the way the sculpture was wired. The slight differences in the frame stem from building the sculpture by hand. To get it as close to perfect as possible, the first thing that was done was laser cut the gears and place them on the floor as an outline for the frame. This can be seen in figure 24 in appendix E, where the other figures of the build are too. Figures 25 and 26 show how the frame is build up bit by bit. The frame was built from pinewood and connected using screws and wood glue in order to create strong joints. Figure 27 shows the finished frame with foot. After the frame was finished the gears were carefully placed one after the other in order to get the spacing right. First the gears that would not be driven directly by one of the motors were placed and the housing for the stepper motors. When all gears and housing were placed the steppers could be placed in their housings. The results of this effort can be seen in figure 28. In figure 29 the remaining gears have been mounted on the steppers and in figure 30 the screen has been placed on the sculpture, which finished it. When looking at the finished sculpture the screen is in sharp contrast with the sculpture itself. This is not necessarily a bad thing as this might attract the attention away from the gears and on to the screen.

As this was just a prototype the wiring was done in such a way that everything was properly connected, but it looked messy. The wiring, which should be neatly tucked away, was now hanging out the back. One of the reasons for this is the large amount of wires that are necessary on a relatively small area, even though the wiring was done in a structured way. An example of how it looked can be found in figure 31. To reduce this wire mess a printed circuit board, or PCB, could possibly be used. Another option could be that the wiring is spread out more making cable management easier and thus clearer.

Testing
As stated earlier, the testing of the sculpture took place in the entrance hall of Hal B at the University of Twente. The sculpture was placed in a prominent position near the main walking routes. The first test setup went without a hitch. The program that displayed the quotes worked fine and there were no disturbances during the test.

The second test had some problems. During the test one of the stepper motors had a problem. It became quite hot and lost its torque. Furthermore the gear on that motor fell off twice. Because of this it was decided that the best thing to do was cut the power to that motor and continue with the remaining motor. As this happened quite early in the test and because of limited time the test was just resumed and not restarted. This is taken into consideration when reviewing the results though. Lastly the third test setup went without problems. For testing purposes the time to switch quotes was reduced to one minute instead of five.

Results experiment

The sculpture
As was described in the implementation the sculpture had some trouble with the stepper motors which couldn't be dealt with in the allotted time. Nevertheless the prototype was completed and up and running. Menperium was happy with the prototype when speaking to them.

The tests
As stated in the method, three test setups were run. Below each one is discussed separately and will be compared in the conclusion.

Experiment 1: no motion
Time: from 11.45 till 14.00 ~ 2 hours 15 minutes
Date: 19-07-2017

Passersby	1388	100%
Did not look	1174	84.58%
Looked while passing	187	13.47%
Stopped to look	27	1.95%

Comments: some of the people who stopped to look were more interested in the sculpture and how it was made then in the quotes. All who looked at it also looked at the screen, but didn't necessarily read the quote.

Experiment 2: constant motion
Time: from 10.43 till 13.20 ~ 2 hours 37 minutes
Date: 20-07-2017

Passersby	1096	100%
Did not look	862	78.65%
Looked while passing	181	16.51%
Stopped to look	53	4.84%

Comments: 11.15 one motor stopped working
Some problems with one of the gears of the above mentioned motor. It fell of twice. All who looked at it also looked at the screen, but didn't necessarily read the quote.

Experiment 3: abrupt change in motion when quote changes
Time: from 10.20 till 13.50 ~ 3 hours 30 minutes
Date: 21-07-2017

Passersby	1215	100%
Did not look	916	75.39%
Looked while passing	248	20.41%
Stopped to look	51	4.20%

Comments: All who looked at it also looked at the screen, but didn't necessarily read the quote.

Significance

Using the website [12] to calculate the probability values of a two-tailed t-test the following results were found. Comparing to a critical p value of 0.05 both the second and third experiment had a significant difference of the number of people who looked at it while passing, respectively p=0.0341 and p=<0.0002. Also the amount of people who stopped to look is significantly different at p=<0.0002 for the second experiment and p=0.0008 for the third experiment. Both compared to the first experiment. When comparing the second experiment with the third experiment we find that the number of passersby that looked while passing is significantly different, namely p=0.0161. The difference between the people who stopped to look at the sculpture is not significant. It has p=0.4599.

Conclusion experiment

When looking at the research data the sub question of how motion can affect attention can be answered as followed. The significant differences between the first experiment and the second/third experiment suggest that motion attracts attention. And the significant differences between the second and third experiment suggest that sudden motion attracts even more attention than a continuous motion. Though the difference of people who stopped was not significant between the second and third experiment, this does not mean that it attracts more or less attention. It mainly means that people aren't more or less interested in the sculpture because of these sudden movements in comparison to continuous motion. As the focus of these tests were attracting attention, and not necessarily grabbing interest, this doesn't matter for the outcome. The fact that a significantly larger percentage of people's attention where drawn during the third experiment means that novel or unexpected motion attracts bottom-up attention. Then to answer the main research question:

> How can a sculpture guide the attention of people to a specific point where a quote is displayed?

It is of a significant importance that the sculpture has moving parts in it as they attract the attention. When the attention is drawn it is also drawn to the screen where the quote is displayed. This is probably because the screen is at eye level and is in sharp contrast with the rest of the sculpture.

Discussion

This was a big project for the amount of time allowed. Completely developing such a big project from the ground up with no real direction other than "it should be a sculpture that shows quotes and it should be big" was very taxing. This had influence on testing the method as this had to be done during the vacation period. This means that there were probably less people who passed by the sculpture. However, with over a 1000 passersby for each test setup this shouldn't be a problem. The fact that only one person was available to keep track of the passersby the counts might be slightly off, but again, on over 1000 people this shouldn't have any consequences for the

data. Another influence on the tests that were run is that the statue did make some noise. This noise wasn't prominent when there was some buzzle in the rest of the surroundings but when it was quiet it was present in such a way that it may have attracted extra attention. There was unfortunately not enough time to test another setup where the noise would not be of influence. However, as it wasn't a prominent noise it could be neglected. For future works the noise could be taken into account and tested.

Conclusion

As stated earlier, this project was a bit too big to properly handle. This created a lack of time to finish in the normally allowed time. Thus extra time to finish up the project was necessary. Nevertheless a working prototype was delivered and the research question was answered. Furthermore the client, Menperium, was happy with the delivered prototype. The sculpture attracted attention and people read the quotes that were on display. Thus it can be concluded that the sculpture and this project were successful.

Recommendations

As in every project there were a few ideas that couldn't be implemented due to a lack of time. Therefore I have some recommendations or future works. For instance, an animation of gears on the screen could be implemented to fill up the now rather empty screen. These gears could even be an indicator of the current category by using colour or gear shapes for example. Another recommendation I would like to make for the screen is the screen ratio. Now it was too square which meant the quotes were either too small or chopped up. A widescreen ratio would allow for bigger text without chopping up the quotes into small parts which don't read well. With these improvements the sculpture would probably attract even more attention.

References

[1] Kuniecki, M., Pilarczyk, J., & Wichary, S. (2015). The color red attracts attention in an emotional context. An ERP study. *Frontiers in human neuroscience*, *9*.
[2] Pal, R., Mukherjee, J., & Mitra, P. (2012, December). How do warm colors affect visual attention?. In *Proceedings of the Eighth Indian Conference on Computer Vision, Graphics and Image Processing* (p. 24). ACM.
[3] Bravo, M. J., & Nakayama, K. (1992). The role of attention in different visual-search tasks. Attention, Perception, & Psychophysics, 51(5), 465-472.
[4] Harris, A. M., Remington, R. W., & Becker, S. I. (2013). Feature specificity in attentional capture by size and color. *Journal of Vision*, *13*(3), 12-12.
[5] Hillstrom, A. P., & Yantis, S. (1994). Visual motion and attentional capture. Attention, Perception, & Psychophysics, 55(4), 399-411.
[6] Becker, S. I., & Horstmann, G. (2011). Novelty and saliency in attentional capture by unannounced motion singletons. Acta psychologica, 136(3), 290-299.
[7] http://www.woodthatworks.com/kinetic-sculptures/infinity, website from David c. Roy about his sculpture infinity, 16-08-2017
[8] http://a360.co/2sRdJQ9, Link to the 3d model, Autodesk, 16-08-2017
[9] https://www.arduino.cc/en/uploads/Tutorial/bipolarKnob_bblg.png, image of stepper wiring, Arduino, 29-07-2017
[10] https://www.openimpulse.com/blog/wp-content/uploads/wpsc/downloadables/42BYGHW208-Stepper-Motor-Datasheet.pdf, datasheet of the wantai 42byghw208 stepper, openimpulse, 17-07-2017
[11] http://www.ti.com/lit/ds/symlink/sn754410.pdf, datasheet of the sn754410ne h-bridge, Texas Instruments, 30-07-2017
[12] http://vassarstats.net/propdiff_ind.html, The Significance of the Difference Between Two Independent Proportions, VassarStats, 31-07-2017

Appendixes

Appendix A

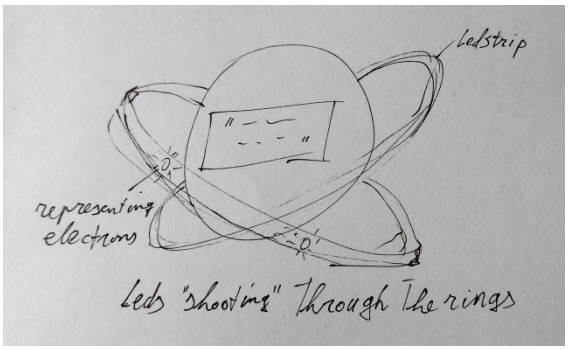

Figure 5.1 first atom sketch

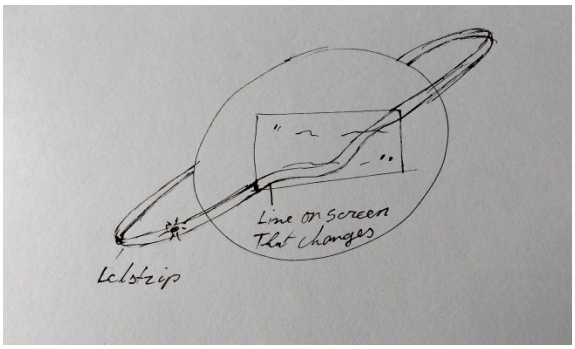

Figure 5.2 ideation sketch atom

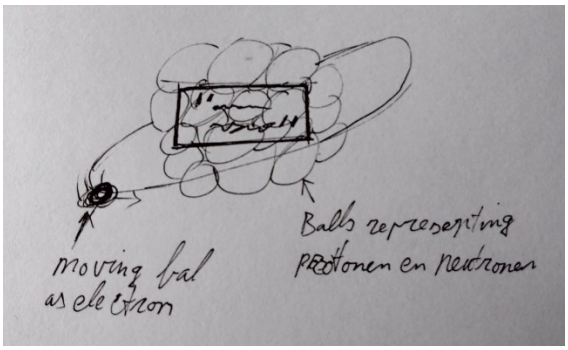

Figure 5.3 second ideation sketch atom

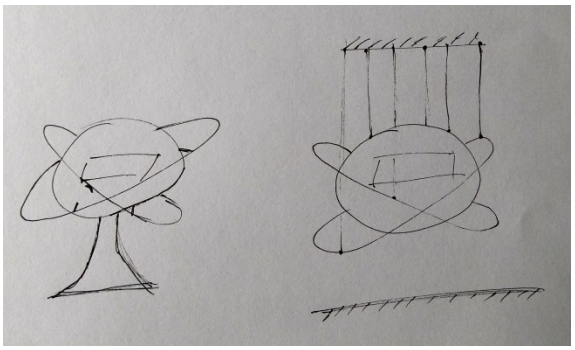
Figure 5.4 ideation how to place

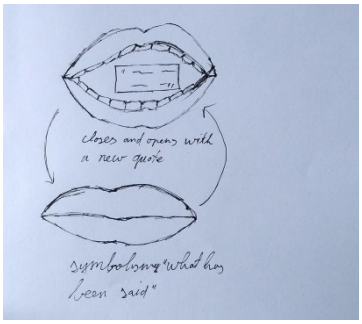

Figure 6. Mouth

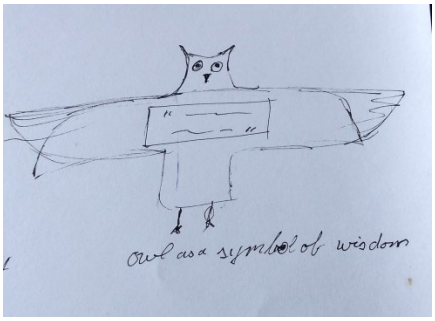

Figure 7. Owl

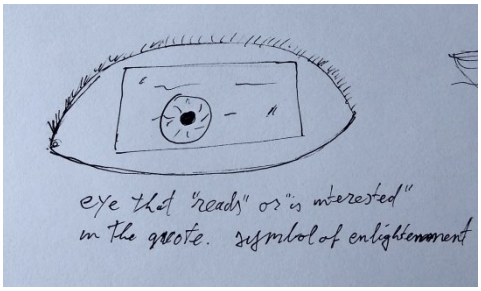

Figure 8.1 Eye, first concept

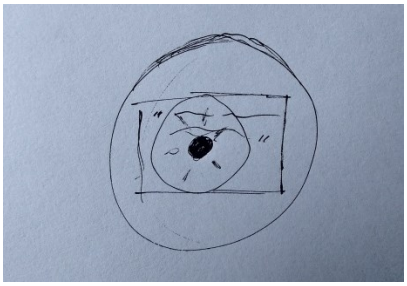
Figure 8.2 Eye, ideation sketch

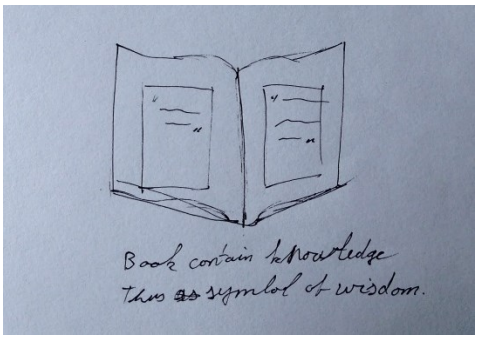

Figure 9. Book

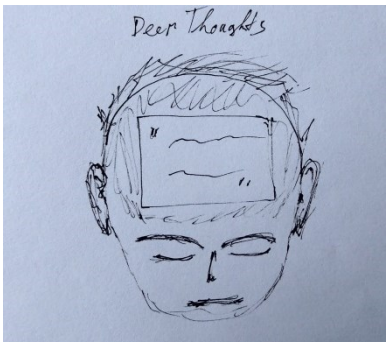

Figure 10. deep thoughts

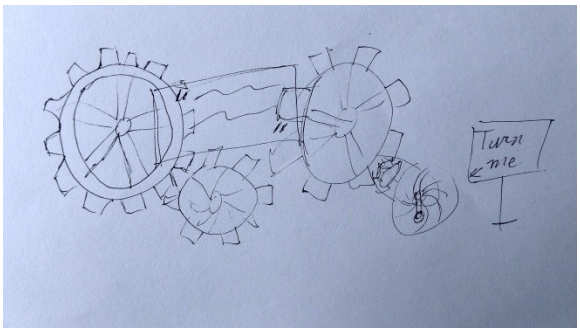

Figure 11. gear machine

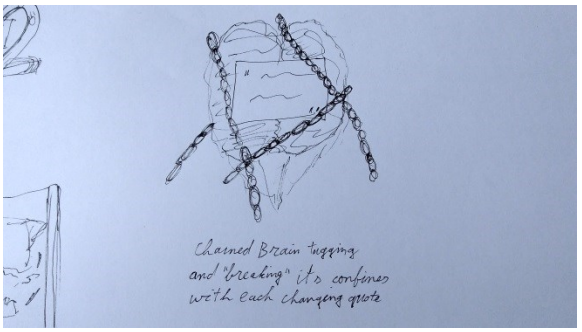

Figure 12. chained brain

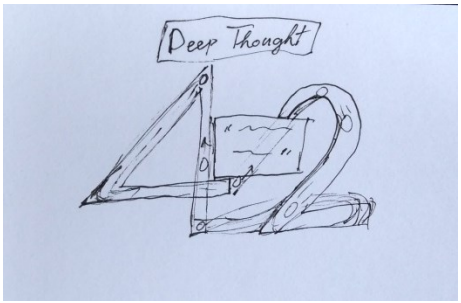

Figure 13. deep thought's answer

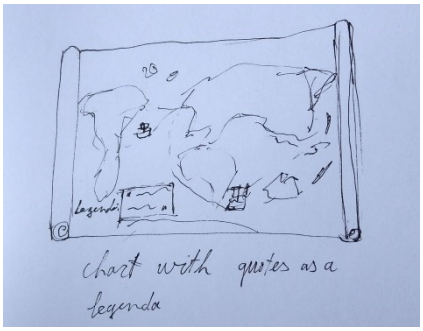

Figure 14. Chart

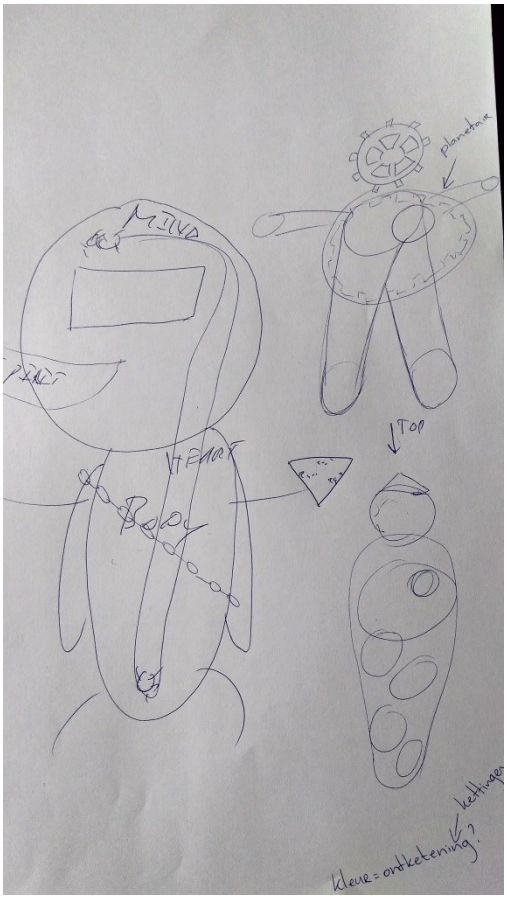

Figure 15. gear / chained person

Appendix B

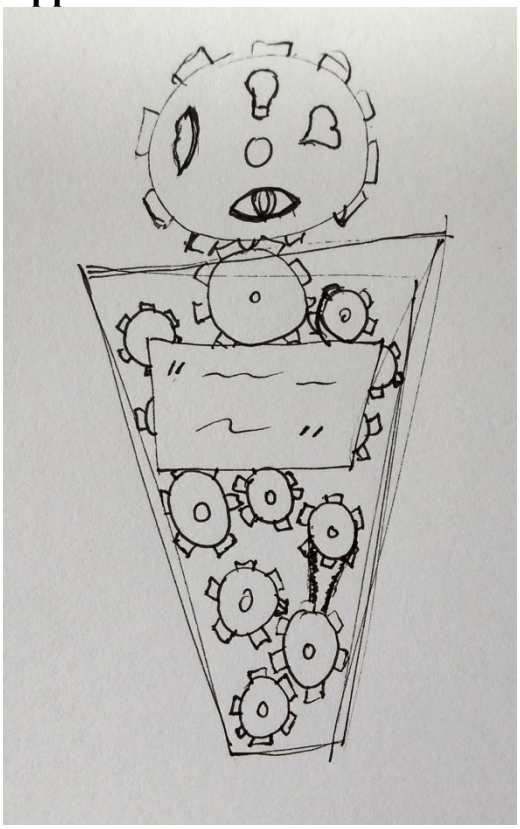

Figure 16: first iteration of the final design

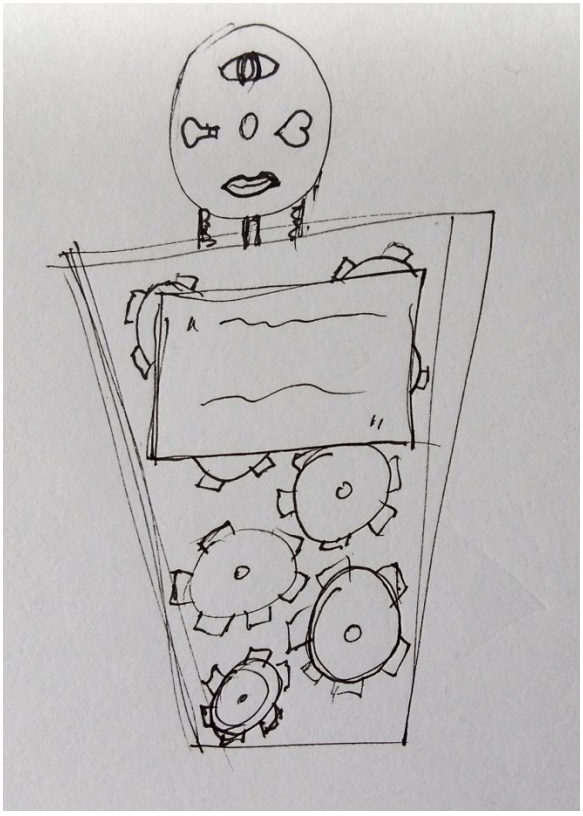

Figure 17: second iteration of the final design

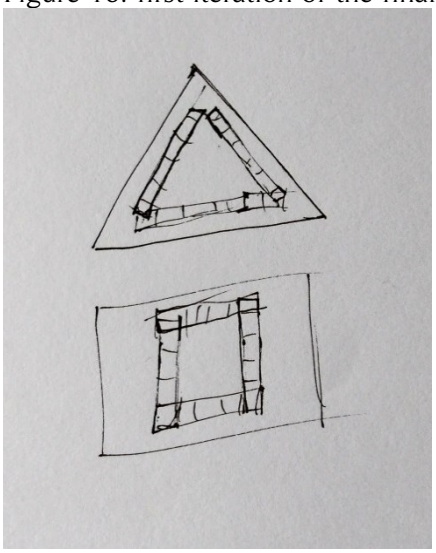

Figure 18:

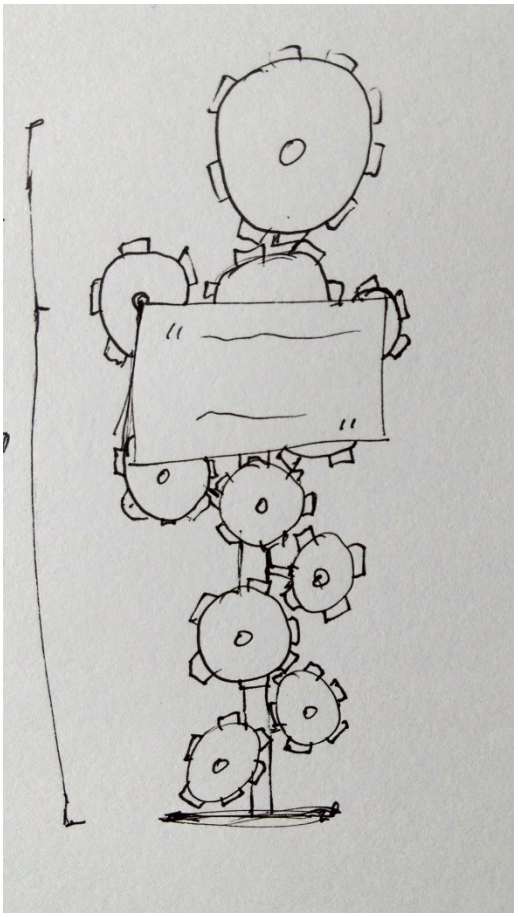

Figure 19: third iteration of the final design

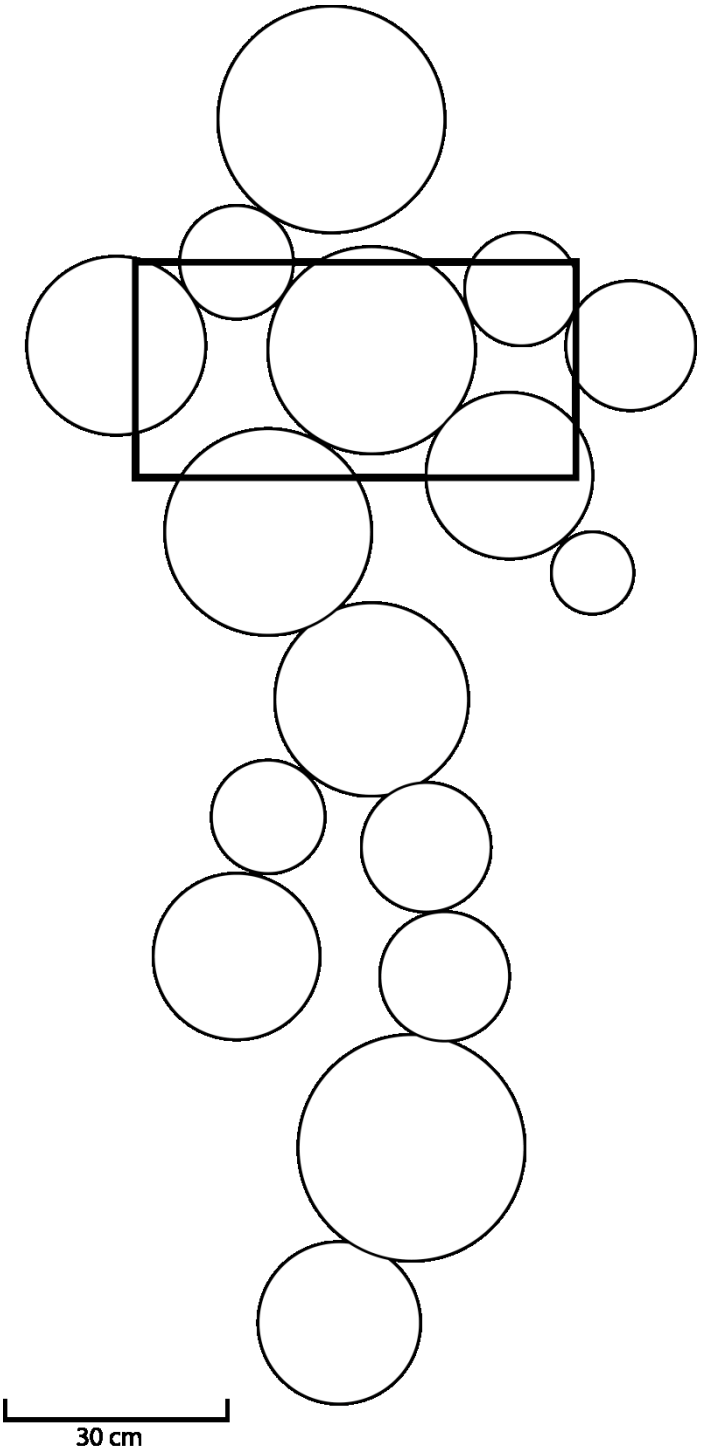

Figure 20: Digital design and positioning

Appendix C

Arduino code:

```
#include <Stepper.h>

const int stepsPerRevolution = 200;  // change this to fit the
number of steps per revolution for your motor


// initialize the stepper library on pins 8 through 11:
Stepper stepperNumber1(stepsPerRevolution, 8, 9, 10, 11);
Stepper stepperNumber2(stepsPerRevolution, 22, 24, 26, 28);
Stepper stepperNumber3(stepsPerRevolution, 31, 33, 35, 37);
Stepper stepperNumber4(stepsPerRevolution, 38, 40, 42, 44);
Stepper stepperNumber5(stepsPerRevolution, 47, 49, 51, 53);

int stepCount = 0;  // number of steps the motor has taken
int motorSpeed1 = 50;   //initial motor speeds
int motorSpeed2 = 50;
int motorSpeed3 = 50;
int motorSpeed4 = 50;
int motorSpeed5 = 50;

int s1direction = 1;    //initial direction
int s2direction = 1;
int s3direction = 1;
int s4direction = 1;
int s5direction = 1;

int incomingByte;       //set incomingByte

void setup() {
  //begin the serial communication at a baudrate of 9600
  Serial.begin(9600);
}

void loop() {
  incomingByte = Serial.read();

  setSpeeds();

  moveSteppers();
}

void setSpeeds(){
  if (incomingByte == 'n') {      //speed up the motors to 70rpm
    motorSpeed1 = 70;
    motorSpeed2 = 70;
    motorSpeed3 = 70;
    motorSpeed4 = 70;
    motorSpeed5 = 70;
  }

  if (incomingByte == 'q') {      //slow down the motors to 25rpm
and reverse the direction
    motorSpeed1 = 25;
```

```
    motorSpeed2 = 25;
    motorSpeed3 = 25;
    motorSpeed4 = 25;
    motorSpeed5 = 25;
    s1direction = -1;
    s2direction = -1;
    s3direction = -1;
    s4direction = -1;
    s5direction = -1;
  }

  if(incomingByte == 'N'){         //stop the motors
    motorSpeed1 = 0;
    motorSpeed2 = 0;
    motorSpeed3 = 0;
    motorSpeed4 = 0;
    motorSpeed5 = 0;
  }

  if (incomingByte == 'C') {       //slow down the motors to 10rpm
and reverse the direction
    motorSpeed1 = 10;
    motorSpeed2 = 10;
    motorSpeed3 = 10;
    motorSpeed4 = 10;
    motorSpeed5 = 10;
    s1direction = -1;
    s2direction = -1;
    s3direction = -1;
    s4direction = -1;
    s5direction = -1;
  }

if(incomingByte == 'R'){         //speed up the motors to 80rpm and
restore the direction
    motorSpeed1 = 80;
    motorSpeed2 = 80;
    motorSpeed3 = 80;
    motorSpeed4 = 80;
    motorSpeed5 = 80;
    s1direction = 1;
    s2direction = 1;
    s3direction = 1;
    s4direction = 1;
    s5direction = 1;
  }

  if(incomingByte == 'c'){       //restore the speed and direction of
the motors
    motorSpeed1 = 40;
    motorSpeed2 = 40;
    motorSpeed3 = 40;
    motorSpeed4 = 40;
    motorSpeed5 = 40;
    s1direction = 1;
    s2direction = 1;
```

```
    s3direction = 1;
    s4direction = 1;
    s5direction = 1;
  }
}

void moveSteppers(){                //put the steppers into action
  // set the motor speed:
  if (motorSpeed1 > 0) {
    stepperNumber1.setSpeed(motorSpeed1);
    // step 1/100 of a revolution:
    stepperNumber1.step(s1direction * stepsPerRevolution / 100);
  }
  if (motorSpeed2 > 0) {
    stepperNumber2.setSpeed(motorSpeed2);
    // step 1/100 of a revolution:
    stepperNumber2.step(s2direction * stepsPerRevolution / 100);
  }
  if (motorSpeed3 > 0) {
    stepperNumber3.setSpeed(motorSpeed3);
    // step 1/100 of a revolution:
    stepperNumber3.step(s3direction * stepsPerRevolution / 100);
  }
  if (motorSpeed4 > 0) {
    stepperNumber4.setSpeed(motorSpeed4);
    // step 1/100 of a revolution:
    stepperNumber4.step(s4direction * stepsPerRevolution / 100);
  }
  if (motorSpeed5 > 0) {
    stepperNumber5.setSpeed(motorSpeed5);
    // step 1/100 of a revolution:
    stepperNumber5.step(s5direction * stepsPerRevolution / 100);
  }
}
```

Processing code:

First tab (quoteDisplay):

```
//the following code is one that loads a tsv file with quotes.
//in this code the tsv file must be in the data folder of the
sketch.
//these quotes must be displayed on a screen and changes over time.

Table table;

int m;
int reset = 0;
int timeToSwitch = 5*60*1000;    //time between switching of the
quotes
boolean switchQuote = false;
boolean startUp;


void setup() {

  //size(1300, 700);
  fullScreen();
```

```
  background(255);

  fill(0);              //fill the color of the text black

  PFont timeless;                                        //get a nice
font
  timeless = createFont("Timeless-Bold.ttf", 30);
  textFont(timeless);
  textAlign(CENTER);

  //String[] fontList = PFont.list();                    //check the
fonts on the computer
  //printArray(fontList);

  table = loadTable("Quotes.txt", "header, tsv");     //import the
tsv containing the quotes and puts them in a table

  println(Serial.list());
  myPort = new Serial(this, Serial.list()[0], 9600);

  countQuotes();

  startUp = true;
  determineCategory();
  startUp = false;

  println(table.getRowCount() + " total rows in table");
}

void draw() {
  if (switchQuote == true) {      //to draw the new quote the draw
cycle must first be finished
    motorControlNewQuote();     //because of the delays in the
motorControl function the motion would be finished before the new
quote is displayed
    switchQuote = false;        //while it should be done on the cue
of a new quote being displayed, hence the if statement
  }

  m = millis()-reset;           //create an interval of X time
  if (m >= timeToSwitch) {      //set an interval for showing a new
quote
    if (currentCategory.equals("happiness")) {
      happiness();
    }
    if (currentCategory.equals("leadership")) {
      leadership();
    }
    if (currentCategory.equals("morals")) {
      morals();
    }
    if (currentCategory.equals("succes")) {
      succes();
    }
    if (currentCategory.equals("personal growth")) {
      personal();
```

```
    }
    println("the current category is " + currentCategory);
    reset = millis();
    switchQuote = true;
  }

  determineCategory();

  writeQuote();        //draw the quote
}

void writeQuote() {
  background(255);
//refresh the background so only 1 quote is on display at a time
  text('"' + quote + '"' + '\n' + "                              ~" +
name + '\n' + "                            " + profession //The
quote that will be displayed and the person who said it underneath
the quote
    , width/6, height/2.5, width/1.5, height/1.3);
//the positioning of the qoute, x position, y position, x size, y
size
}

//for testing purposes
void keyPressed() {
  if (key == 'h') {
    happiness();
  }
  if (key == 'l') {
    leadership();
  }
  if (key == 'm') {
    morals();
  }
  if (key == 's') {
    succes();
  }
  if (key == 'p') {
    personal();
  }
}
```

Second tab (count_quotes):

```
int countH = 0;                    //count_quotes variables
int countM = 0;
int countP = 0;
int countS = 0;
int countL = 0;

void countQuotes() {               //function to determine the amount
of qoutes for every different category, in the table that is loaded

  for (TableRow happy : table.findRows("happiness", "Category")) {
    countH = countH +1;
  }
  println("amount of happiness quotes: " + countH);
```

```
  for (TableRow leader : table.findRows("leadership", "Category")) {
    countL = countL +1;
  }
  println("amount of leadership quotes: " + countL);

  for (TableRow morals : table.findRows("morals", "Category")) {
    countM = countM +1;
  }
  println("amount of morals quotes: " + countM);

  for (TableRow succes : table.findRows("succes", "Category")) {
    countS = countS +1;
  }
  println("amount of succes quotes: " + countS);

  for (TableRow pers : table.findRows("personal growth",
"Category")) {
    countP = countP +1;
  }
  println("amount of personal growth quotes: " + countP);
}
```

Third tab (determine_category):

```
// using Jsoup
// For this program you have to download jsoup-1.7.1.jar from
http://jsoup.org/download
// and add it via Sketch>Import Library>Add Library

import org.jsoup.Jsoup;
import org.jsoup.nodes.Document;
import org.jsoup.select.Elements;
import org.jsoup.nodes.Element;

//buzz word counters to determine the current category based on buzz
words in the news
int happinessCounter = 0;
int leadershipCounter = 0;
int moralsCounter = 0;
int succesCounter = 0;
int personalCounter = 0;

//in the string arrays below the buzz words for the different
category are determined and stored
String[] happinessBuzzWords = {"gelukkig", "blij", "positief",
"beter", "goed", "blijdschap", "geluk"};
String[] leadershipBuzzWords = {"leider", "president", "CEO",
"leiderschap", "beleid", /*"minister",*/ "aansturen",
/*"ministerie",*/ "premier"};
String[] moralsBuzzWords = {"goed", "slecht", "aanslag", "terreur",
"moraal", "verijdelt", "justitie"};
String[] succesBuzzWords = {"vooruitgang", "succes", "overwinning",
"winnen", "beleggers", "winst", "omhoog", "stijgen"};
String[] personalBuzzWords = {"verbetering", "eigen",
"persoonlijke", "ontwikkeling", "hobbels", "tegenslagen",
"tegenslag", "groei"};
```

```
String currentCategory = "happiness";
//currentCategory stores which category should be displayed. It
starts with happiness because it should have an initial value to
directly start with showing quotes

void determineCategory() {
  //everyday at 8:00 and at 13:00 the category is determined or when
starting up the program
  if (hour() == 8 && minute() == 0 && second() == 0 || hour() == 13
&& minute() == 0 && second() == 0 || startUp == true) {
    delay(1000);                      //delay for a second to avoid
running this function multiple times in one second
    happinessCounter = 0;             //reset the counters
    leadershipCounter = 0;
    moralsCounter = 0;
    succesCounter = 0;
    personalCounter = 0;

    getHeadlines();                   //getHeadlines is the function
that scrapes the web and checks the current headlines

    //the if else loops check which category is bigger at the moment
    if (happinessCounter > leadershipCounter && happinessCounter >
moralsCounter && happinessCounter > succesCounter &&
happinessCounter > personalCounter) {
      currentCategory = "happiness";        //set the current
category
      happiness();                          //show a new quote in
the new category
    } else if (leadershipCounter > happinessCounter &&
leadershipCounter > moralsCounter && leadershipCounter >
succesCounter && leadershipCounter > personalCounter) {
      currentCategory = "leadership";        //set the current
category
      leadership();                          //show a new quote in
the new category
    } else if (moralsCounter > happinessCounter && moralsCounter >
leadershipCounter && moralsCounter > succesCounter && moralsCounter
> personalCounter) {
      currentCategory = "morals";        //set the current category
      morals();                          //show a new quote in the
new category
    } else if (succesCounter > happinessCounter && succesCounter >
leadershipCounter && succesCounter > moralsCounter && succesCounter
> personalCounter) {
      currentCategory = "succes";        //set the current category
      succes();                          //show a new quote in the
new category
    } else {
      currentCategory = "personal growth";        //set the current
category
      personal();                                 //show a new quote
in the new category
    }
    println(currentCategory + " is bigger");
```

```
    reset = millis();                    //reset the reset so the
new quote is displayed as long as usual

    motorControlNewCategory();
  }
}

void getHeadlines() {
  // doc contains the result from the website request
  Document doc;
  // this is the actual request, including error handling
  try {
    doc = Jsoup.connect("http://www.nu.nl").get();
//connect to the news website nu.nl and gets the sourcecode as a doc
    //doc = Jsoup.connect(url).get();
  }
  catch(IOException e) { // Connection error
    e.printStackTrace();
    doc = null;
  }
  catch(IllegalArgumentException e) {
    println("Bad URL!");
    doc = null;
  }

  if (doc != null) {
    //this is the command that selects the data from the result
expression
    Elements headlines = doc.select("#main > div.column-
content.clearfix > div.column.first");          //selects the path
that has the headlines in it
    //the rest here is only to get the data in a nice format, times
is a list
    if (headlines != null) {
      //timeString contains then the first string of the list times
      String headlineString = headlines.text();
      //this is to separate the data into strings containing just
one word by seperating the sting at every space
      //this way each word can be checked later on
      String[] splitResult = split(headlineString, ' ');

      if (splitResult.length > 0) {
        //printArray(splitResult);
        for (int i = 0; i < splitResult.length; i++) {
//these for loops go through each splitResult and checks them
against all the buzz words in the second for loop
          for (int h = 0; h < happinessBuzzWords.length; h++) {
//check for happiness buzz words
            if (splitResult[i].equals(happinessBuzzWords[h])) {
//check if a word from the headlines is equal to one of the
buzzwords
              happinessCounter++;
//count the amount of buzz words in the headlines
              //println(splitResult[i] + " " + i + " is the same as
" + happinessBuzzWords[h]);
            }
```

```
            for (int l = 0; l < leadershipBuzzWords.length; l++) {
//check for leadership buzz words
              if (splitResult[i].equals(leadershipBuzzWords[l])) {
//check if a word from the headlines is equal to one of the
buzzwords
                leadershipCounter++;
//count the amount of buzz words in the headlines
                //println(splitResult[i] + " " + i + " is the same
as " + leadershipBuzzWords[l]);
              }
            }
            for (int m = 0; m < moralsBuzzWords.length; m++) {
//check for morals buzz words
              if (splitResult[i].equals(moralsBuzzWords[m])) {
//check if a word from the headlines is equal to one of the
buzzwords
                moralsCounter++;
//count the amount of buzz words in the headlines
                //println(splitResult[i] + " " + i + " is the same
as " + moralsBuzzWords[m]);
              }
            }
            for (int s = 0; s < succesBuzzWords.length; s++) {
//check for succes buzz words
              if (splitResult[i].equals(succesBuzzWords[s])) {
//check if a word from the headlines is equal to one of the
buzzwords
                succesCounter++;
//count the amount of buzz words in the headlines
                //println(splitResult[i] + " " + i + " is the same
as " + succesBuzzWords[s]);
              }
            }
            for (int p = 0; p < personalBuzzWords.length; p++) {
//check for personal growth buzz words
              if (splitResult[i].equals(personalBuzzWords[p])) {
//check if a word from the headlines is equal to one of the
buzzwords
                personalCounter++;
//count the amount of buzz words in the headlines
                //println(splitResult[i] + " " + i + " is the same
as " + personalBuzzWords[p]);
              }
            }
          }
        }
        /*println("happiness counter is: " + happinessCounter);
//check if the counter and getHeadlines function works properly
        println("leadership counter is: " + leadershipCounter);
        println("morals counter is: " + moralsCounter);
        println("succes counter is: " + succesCounter);
        println("personal counter is: " + personalCounter);*/
      }
    }
  }
}
```

Fourth tab (get_quote):

```
int happyIDOld = 0;                //variables to store previous IDs
to check new IDs
int leaderIDOld = 0;
int moralsIDOld = 0;
int succesIDOld = 0;
int personalIDOld = 0;

String id;
String quote;
String name;
String category;
String profession;

void happiness() {    //get a random qoute in the category happiness
  int happyID = int(random(1, countH));
  if (happyID == happyIDOld) {      //make sure the new quote to be
displayed is not the same as the old one
    happyID = happyID +1;          //when the random new quote is the
same as the old one take the next quote
    if (happyID > countH) {          //make sure the next qoute
actually exists, otherwise go back in the table to a quote that does
      happyID = happyID -2;
    }
    //println("happyID +1");
  }
  happyIDOld = happyID;

  TableRow row = table.findRow("H" + happyID, "ID");

  id = row.getString("ID");                            //get the info
from the table and put it into strings
  quote = row.getString("Quote");
  name = row.getString("Name");
  category = row.getString("Category");
  profession = row.getString("Profession");
}

void leadership() {    //get a random qoute in the category
leadership
  int leaderID = int(random(1, countL));
  if (leaderID == leaderIDOld) {
    leaderID = leaderID +1;
    if (leaderID > countL) {
      leaderID = leaderID -2;
    }
    //println("leaderID +1");
  }
  leaderIDOld = leaderID;

  TableRow row = table.findRow("L" + leaderID, "ID");

  id = row.getString("ID");
  quote = row.getString("Quote");
  name = row.getString("Name");
```

```
  category = row.getString("Category");
  profession = row.getString("Profession");

  //println(name + " " + category + " (" + quote + ") has an ID of "
+ id);
}

void morals() {    //get a random qoute in the category morals
  int moralsID = int(random(1, countM));
  if (moralsID == moralsIDOld) {
    moralsID = moralsID +1;
    if (moralsID > countM) {
      moralsID = moralsID -2;
    }
    //println("moralsID +1");
  }
  moralsIDOld = moralsID;

  TableRow row = table.findRow("M" + moralsID, "ID");

  id = row.getString("ID");
  quote = row.getString("Quote");
  name = row.getString("Name");
  category = row.getString("Category");
  profession = row.getString("Profession");

  //println(name + " " + category + " (" + quote + ") has an ID of "
+ id);
}

void succes() {    //get a random qoute in the category succes
  int succesID = int(random(1, countS));
  if (succesID == succesIDOld) {
    succesID = succesID +1;
    if (succesID > countS) {
      succesID = succesID -2;
    }
    //println("succesID +1");
  }
  succesIDOld = succesID;

  TableRow row = table.findRow("S" + succesID, "ID");

  id = row.getString("ID");
  quote = row.getString("Quote");
  name = row.getString("Name");
  category = row.getString("Category");
  profession = row.getString("Profession");

  //println(name + " " + category + " (" + quote + ") has an ID of "
+ id);
}

void personal() {    //get a random qoute in the category personal
growth
  int personalID = int(random(1, countP));
```

```
  if (personalID == personalIDOld) {
    personalID = personalID +1;
    if (personalID > countP) {
      personalID = personalID -2;
    }
    //println("personalID +1");
  }
  personalIDOld = personalID;

  TableRow row = table.findRow("P" + personalID, "ID");

  id = row.getString("ID");
  quote = row.getString("Quote");
  name = row.getString("Name");
  category = row.getString("Category");
  profession = row.getString("Profession");

  //println(name + " " + category + " (" + quote + ") has an ID of "
+ id);
}
```

Fifth tab (motor_control):

```
import processing.serial.*;

Serial myPort;

int s = 1000;

void motorControlNewQuote(){      //when a new quote is displayed
let the sculpture do something to attend people on a new quote being
displayed
  determineCategory();            //before running the delays, check
if it's time for a new category

  myPort.write('n');              //send bytes to the
microcontroller to alter the speed and/or direction of the motors
  delay(1*s);
  myPort.write('q');
  delay(4*s);
  myPort.write('c');
}

void motorControlNewCategory(){    //when the category gets updated
show the change also with the sculpture
  myPort.write('N');
  delay(1*s);
  myPort.write('C');
  delay(3*s);
  myPort.write('R');
  delay(1*s);
  myPort.write('c');
}
```

Appendix D

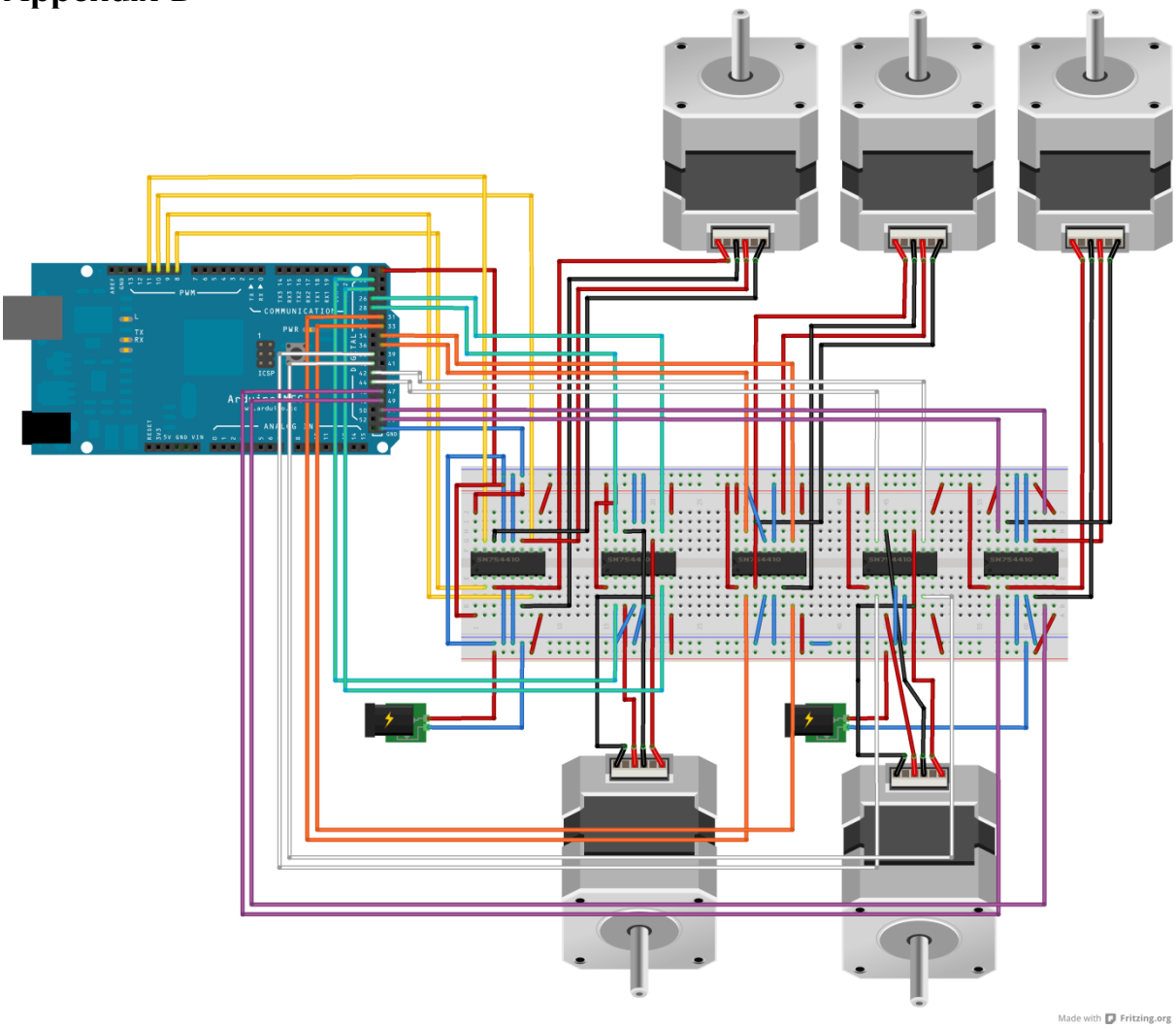

Figure 22: overview of the entire schematics

Appendix E

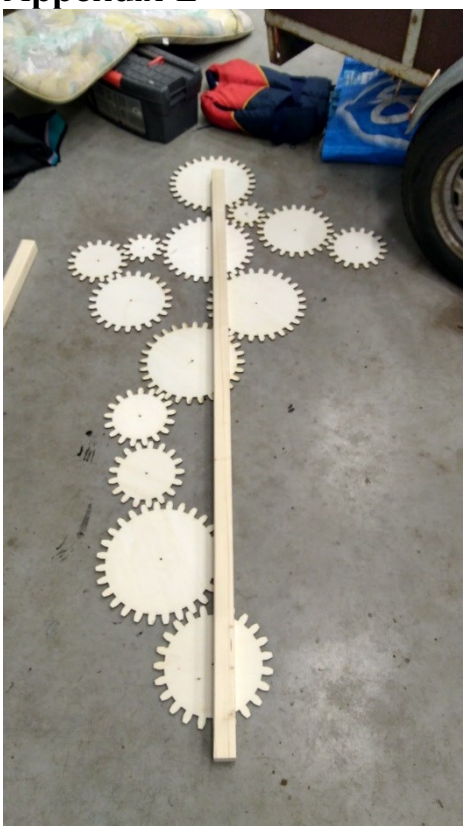

Figure 24: Outlined and first beam of the frame

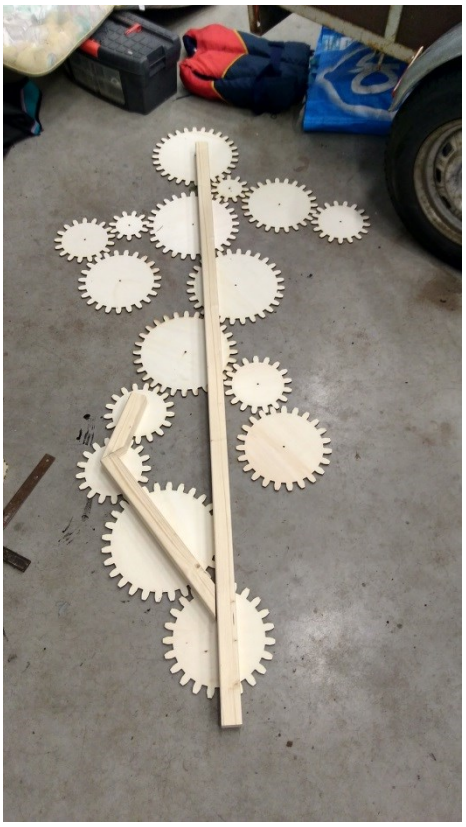

Figure 25: First branch in the frame

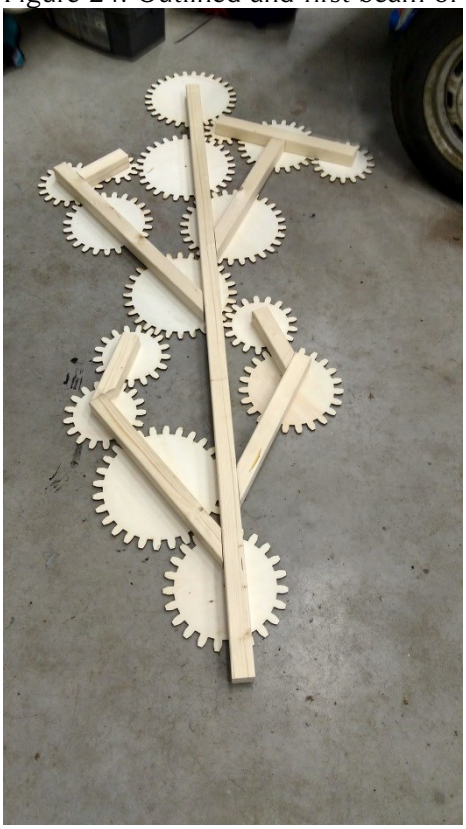

Figure 26: All branches of the frame

Figure 27: Frame finished

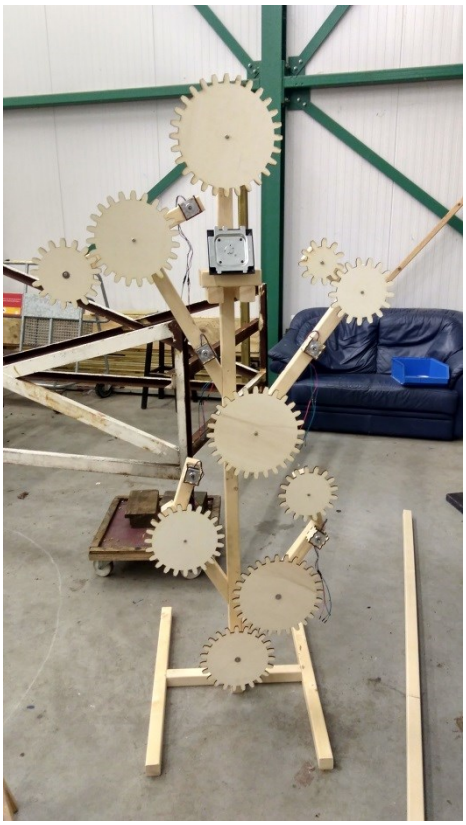

Figure 28: All gears except the motor gears on frame

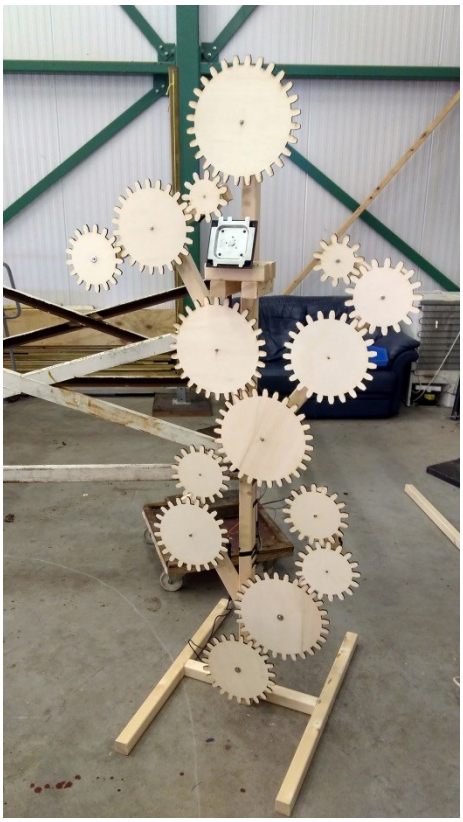

Figure 29: All gears on frame

Figure 30: Finished sculpture

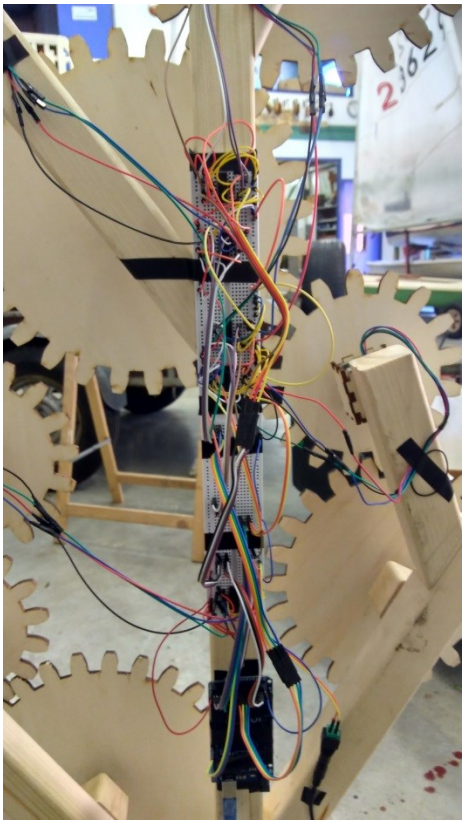

Figure 31: Image of the wiring on the back of the sculpture