Exploring the Design Space of Tangible Interfaces

Christopher Heiden
Student ID: 120249

ABSTRACT

Tokens and constraints are a major part of the wide topic of tangible user interfaces. By touching, pushing or rotating three-dimensional objects, an underlying dataset can be changed, and the system itself reacts to the user input. It provides users with the option not only to use one or two fingers at once. Hands can be used for more precise interaction and can deal with many different shapes and forms; however, people do not use these possibilities by interacting with a common computer or laptop or smartphone, because of the two-dimensional interaction and representation layer.

The combination and the interaction with tokens and constraints are in focus of the project. Therefore, theoretical approaches to design and build a tangible user interface are explored. Moreover, bodystorming is used to find interaction ideas. With these approaches, a new tangible user interaction system is built and used to create abstract instruments.

Keywords
Human-Computer Interaction, Tangible User Interfaces, UI, Tokens & Constraints, Interaction, Sound Generation, Bodystorming

INTRODUCTION

Common two-dimensional user interfaces are used in a wide range of applications. To interact with these kinds of applications, screen and input devices like a keyboard or a mouse or both together are used. On the other hand, Humans are not used to interacting within a two-dimensional space because, in the real world, humans are surrounded by three-dimensional objects. So, there is a mismatch between two-dimensional interfaces and interaction on digital devices and three-dimensional interaction with objects in real life. Still, people became used to work in two-dimensional space and just using one or two fingers over the years.

The idea of changing this standard and using three-dimensional spaces is not new. This project tries to explore the possibilities of tangible user interfaces by using music.

RELATED WORK

There are many approaches to design and build tangible user interfaces; however, all approaches share the same idea. All agree that a physical object represents data and changing its object state changes the value of the object it represents.

One approach to build a tangible user interface is using tokens and constraints. Tokens are physical objects with an identifier, so a digital system can identify and handle the token that stores information. On the other hand, a constraint is an area that restricts the user from doing specific interaction gestures like moving the token to one specific position or rotating it. Therefore, B. Ullmer, H. Ishii, and R. J. K. Jacob present grammar that can be used to explain interactions with the token and what users would expect from the system.[1]

B. Ullmer, H. Ishii and R. J. K. Jacob state that token and constraint systems are used to interact with abstract and digital information that has no inherent physical representation, nor any intrinsic physical language for its manipulation. They draw on our basic knowledge about the behavior of the physical world and change the understanding of the relationship between physical tokens and constraints. Therefore, two interactions are defined, association and manipulation. [1]
First, there is the association which means that the user can place a token in or on the constraint, so the system can identify the token. [1]
Second, manipulation is the movement of a token like rotation or shifting. [1]
Adding multiple tokens at once is possible, so multiple tokens can be used on one constraint or even nested tokens could be possible. [1]
To describe the mapping of physical relationships to digital interpretations, a grammar table has been presented (see Table 1).

Physical Relationship	*Interaction Event*	*Digital Interpretations*
presence	Add remove	binding

Position	move	Scalar
Sequence	Order change	Query ordering
Proximity	Proximity change	Relationship strength
Connection	Connect/disconnect	Logical flow
Adjacency	Adjacent/ no-Adjacent	Boolean, axes
nesting	nesting	grouping

Table 1 shows a smaller grammar list that has been defined by authors B. Ullmer, H. Ishii and R. J. K. Jacob. These grammars are used for the mapping of physical relationships to digital interpretations. [1]

A similar paper that tries to find an approach to handle tangible user interactions and their interaction methods is called "The TAC Paradigm: Specifying Tangible User Interfaces". In this paper written by O. Shaer, N. Leland, E. H. Calvillo-Gamez and R. J. K. Jacob discuss the TAC paradigm, challenges for tangible interaction systems, and how to evaluate them. [2]
The TAC paradigm "identifies the common components and properties sufficient for specifying the structure and functionality of a wide range of TUIs." [2] Therefore, tangible interaction systems are defined as a set of physical objects and digital information. This concept has five key properties. [2]
First, coupling is used to couple digital information and a physical object, so it can be defined as a token. A toke is a physical object that represents digital information. [2]
Second, there is a relative definition between a token and a constraint that limits the behavior of the token or both. [2]
Third, a "TAC is created when a token is physically associated with a constraint". [2]
Fourth, interaction and manipulation of the physical object are interpreted. This process is automated regards to its constraint (computational interpretation). [2]
Fifth, manipulation can happen "discretely, continuously or in both ways." [2]
Besides, other keys of a TAC system are presented.
For example, the "system [should] allow[.] users to design the information architecture" [2]. Therefore, simultaneous and cooperative work can be explored. Furthermore, it should "allow[.] users to construct a structure" [2] that visualizes information graphically on a display. The structure is built upon blocks. Each block is aware of its neighbors. The tokens and their constraints use physical constraints "to express, manipulate and visualize queries." [2] By manipulation, tangible feedback works by a remote user. [2]

Though these two papers focus only on tokens and constraints and how to handle them, other approaches are also used in Human-Computer Interaction.

One is called "MCRpd-pattern" that has been explained in the paper "Emerging Frameworks for Tangible User Interfaces." This paper discusses the technological extension of an object and how it is coupled to other its object. Therefore, it introduces MCRpd-pattern.
"M" stands for model and is the underlying digital database of information. "C" is the controller that is the input device. This is directly linked to rep-p (physical representations) that is "computationally coupled to underlying digital information (model)" [and] "[t]he physical representations are perceptually coupled to actively mediated digital representations (rep-d). [3]
In addition to this, the paper points out many examples and explains how the tokens are coupled. Furthermore, the paper explains in detail why and how these tokens are coupled, and categorized projects by using names like Spatial, Constructive, Relational and Associative. [3]
Coupling in this context means that additional information are shown around or next to the token. [3]
Spatial in this context means that tokens have a relationship to each other and express this relationship by their special position to each other. So, not the absolute position in a world is important, the relative position to another token is. [3]
The constructive approach is the "middle ground between spatial and relational approaches" [3]. It involves modular elements that can be used together. [3]
Relational, means that tokens can be linked together and the sequence, how these tokens are linked is important and change the outcome of the system. [3]
Finally, the associative group binds all other tangible user interfaces that cannot be linked to the other three groups. [3]

Another paper that has been written by Steve Harrison, Deborah Tatar, and Phoebe Sengers and is called "The Three Paradigms of HCI". These parameters are called Human-Factors, Classical Cognitivism/Information Processing Based, Third/Phenomenologically-Situated Paradigm. The authors split the paradigms and defined all of them and explained how thinking can improve the tangible interface. [4]
The Human-Factor paradigm has the goal to optimize the interaction between the human and the computer. So, the shape of the props, the possible interaction methods and how to avoid errors are defined. [4]
In the Classical Cognitivism/Information Processing Based paradigm that test the actual system and try to improve it by research. [4]
Last but not least, the Phenomenologically-Situated Paradigm adds some emotional and cultural perspectives into the system. So here, the focus is not the system itself but rather than where and who is using it at some point. [4]

CLASS PROCESS

In the first couple of meetings, the class read about several different approaches to deal with tangible user interfaces (see RELATED WORK), worked on video prototypes to present their ideas and learned about bodystorming[1]. At first, the group read about interaction with physical objects and how interaction designers can use them to deal with data that is bind to objects. Therefore, the papers “Token+constraint systems for tangible interaction with digital information” [1] and “The TAC Paradigm: Specifying Tangible User Interfaces” [2] were given to read and to present in a poster presentation (see Table 2).

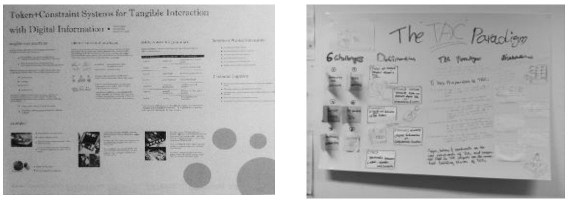


Table 2 shows the two posters that the group designed and presented. They show a summary of the papers “Token+constraint systems for tangible interaction with digital information”[1] and “The TAC Paradigm: Specifying Tangible User Interfaces”[2].

These papers and their posters have been presented and discussed. Furthermore, clay has been used for prototyping. Therefore, vocabulary has been defined which had to be presented in an object. One example would be ‘resolution’. Props have to be designed to symbolize the word (see Figure 1). Furthermore, the project group did some brainstorming sessions to scrabble some ideas about how an interactive system could look like for a chosen word.

Figure 1 shows the outcome of the clay prototyping session.

Furthermore, to improve the understanding of spatial, constructive, relational and associative tangible user interfaces, one for each category had to be found and presented. Therefore, two weeks have been claimed for this. In the first week, the group split up into two groups and thought and presented their group ideas in a small presentation (see Figure 2).

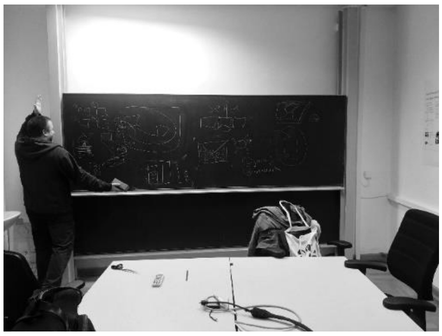


Figure 2 shows some ideas one group had in the idea and design process for the video prototype that has to be filmed later.

The first idea of one group was sound editing and how to make it tangible. The main problem of sound editing is that it is difficult to work cooperatively and simultaneously with other people. There is usually only one mouse and keyboard that is linked to a computer that can be used as an input device. Therefore, it is difficult to work together. The idea is to improve this by using tokens and constraints.

The second idea of the same group was about work-life-balance in a company and how to visualize how much work someone has. Therefore, a heart symbolizes the amount of work and balls are used to define tasks that can be placed around the heath. According to the number of balls and the heath shape changes. Moreover, the balls change their colors if the deadline comes closes.

Finally, there was an idea about robots that can be programmed with puzzle parts to find a door to leave a maze. Therefore, the two teams are needed. The first team programs the maze and how it looks like. The second team programs the movement of the robot that is inside the maze. The goal is to find a way out with the robot. The programming can be done by puzzle parts. The puzzle parts are color-coded that define the height level of a maze, borders or the code structure like red puzzle parts are used to define a loop.

Tips and what very group has to improve for each idea has been given. One idea of every group had to be video prototyped in the second week.

In the decision process, what the group wanted to video prototype, the group discussed everything and decided to build the maze-robot-idea. However, after some research, several example projects with nearly the same principles have been found[5], so the group decided to build the SoundEditer-Idea. The other group built a driving system that reacts to the actual situation on the streets according to weather, time of a day.

[1] Bodystorming is a method to find new inspiration and ideas for prototyping. This approach uses role-play elements to interact with an object, a prop or the actual prototype to gather information about how people use the device. The underlying relationships between the human, the surrounding, and the object will be explored.[41]–[43]

To film the ideas, the groups had to cut wood pieces and buy some paper and print out stuff. After preparation, the authors filmed the interaction with the system to show the functionalities. The outcome had to be presented and the whole group discussed what every team must improve. After discussing everything, “bodystorming” has been introduced and a new paper[4] had to be read.

Next time, the whole group looked at the possibilities of accessible old technologies. Later, the paper has been discussed and ideas had to be found for the final project idea. Therefore, the students brainstormed in the same session. However, two more weeks have been claimed to find inspiration. Some students met, for example, with other people to find creativity. One person that has been asked studies law. In this meeting, the study program and the different books of law have been discussed. In addition to this, the person explained how a law case can be structured and how she is dealing with such cases. She also explained where she is studying (library or at home) and that she is learning alone most of the time. With this knowledge, a new idea has been developed, how a law student could use a tangible user interface.

Ideas that became in focus were a fighting game with real swords, the creation of abstract music instruments and sound creation by forest sounds and plants.

The sword idea is a tangible game. Two or more players draw cards that show a specific element like water or fire. These tokens have to be placed on the handle of the sword. The sword can be used to attack the other players, and according to the elements, the player makes different damage. The sword is divided into three areas (attack, defense, and void). Depending on the players’ hit, the opponent points are reduced, and at some point, the element gets destroyed and a new element must be drawn. If the play does not have any cards anymore, the other player has won.

The other ideas (abstract music instrument) uses objects that usually do not have an instrumental background; however, these objects get a musical, an augmented meaning. Any object can be used to create music if a sensor module is placed on the object. The objects/abstract instruments are constraints. At the same time, it uses a token to identify the instrument. If the technical system is placed on the token, music is generated. A token can be added to activate the object to play some abstract music, so nothing comparable to already existing music instrument sounds.

The last idea that was in the pool of favorites was the forest sound idea. Therefore, plants can be placed on the shelf and depending on where the plant comes from, different forest music is played. If the user wants to change the volume, he/she can rotate the shelve.

In the final meeting, the whole group decided which idea/ ideas were the best and what is worth building and implementing. Therefore, all favorite ideas have been discussed again to find out their potential (see Figure 3). Finally, the group decided to build and implement the music instrument idea. After this decision, further process has been debated.

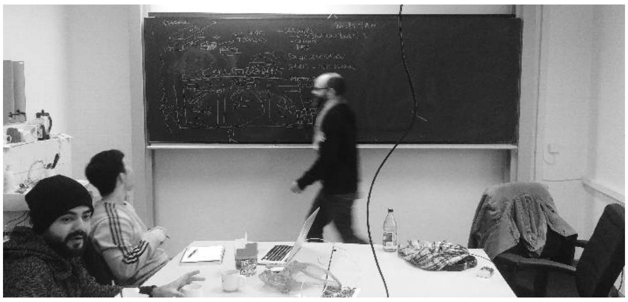


Figure 3 shows sketches of the music instrument and the sword game idea, so the whole group can find out the potential of both.

After the decision, the group met every Monday and talked about the progress and problems. Furthermore, designs have been discussed, how to program and work with a microcontroller[2] and their communication protocols and how sound generation can be programmed.

At the end of the semester, the project prototype had to be presented. Therefore, two videos have been produced. The first video shows all functionalities of the actual prototype. The second video shows the project vision and future work that could be done.

PROJECT PROCESS

1. PROGRAMMING

To define sensors that should be used for the abstract instruments sound generator project, the group did a

[2] Microcontrollers are an extremely small computer, logic systems which have become possible to be due to the progress of modern micro-technologies. The heart of the controller is the microprocessor which executes the commands and calculations. The advantage of microcontrollers is their flexibility, so they can be extended by sensors like an acceleration sensor. This flexibility is possible because of several input/output ports on the microcontroller and analog/digital converters. There are many different types of microcontrollers, which may differ in chipset, memory, and interfaces. Well-known microcontrollers are the ESP8266, ESP32 or the Arduino family. [44], [45]

bodystorming session and defined objects that could be used to generate sound. After around 15 minutes of finding objects in the lab, the ideas have been debated and how the interaction with the object would look like and what kind of sound the object could make. After thinking about objects and how people could use them, the group mapped this experience and interaction gestures with sensors that can detect these specific movements. For every interaction, the group came up with what can be detected by an accelerometer[6], [7], a distance sensor[8], and an orientation sensor[9], [10] (see Figure 4). The interaction should not be restricted, so a wireless connection has been prioritized. This would allow the user to interact freely because no cables would stop the user to do a specific movement.

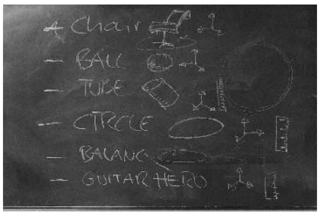


Figure 4 shows the objects that could be used to generate sound and their sensor values that can define the possible gestures users can do with the objects.

To generate the sound, the group decided to use Pure Date[11]. This software allows the user the opportunity to generate sound by interaction.

Before starting to implement, students had to learn how to program a microcontroller and how a microcontroller works in general. Not every student had experience with microcontrollers, so those students have taught the other members and gave the others a small introduction. In this session, the Arduino Uno[12] has been programmed, so it can gather the distance and acceleration values.

During the vacations, one team member implemented a network with an ESP8266[13]. This one could send UDP packages over WIFI. Pure Data can receive the packages and interpret the OSC data. Because of the different microcontrollers, the implementation of the accelerometer and the ultrasonic distance sensor had to be changed. However, after programming the network, Pure Data could not be received any data.

New sensors have been used to detect the distance to the user and the acceleration. Before, sensors like the hc-sr04 (distance sensor)[8] and the hmc5883l (acceleration sensor)[14] have been used. Instead, the chips LSM303DLHC[15] and the VL53L1X[16] are used. This specific acceleration chip (LSM303DLHC) that has been used can also be programmed to detect the orientation. Because of this change, the software had to be reprogrammed. In addition, the new sensors had to be soldered for testing reasons. There were no pins attached to the sensors before, so soldering was mandatory for testing. Furthermore, the microcontroller has been programmed, so it can use the RFID[3] reader (RFID-RC522[17]) to read tags. This chip can detect NFC tags, so it can be used to define the instrument and its sounds. At the beginning of the project, the idea was that users can also use modifiers, so the user can change the sound of an instrument. Therefore, two RFID readers would have been needed. After a long discussion, the group agreed that the focus of the project should be to identify an instrument with RFID tags.

After coding and testing every sensor separately, the microcontroller and the sensors have been soldered. Therefore, a circuit board, an RFID reader, the LSM303DLHC and a VL53L1X sensor has been linked together (see Figure 5).

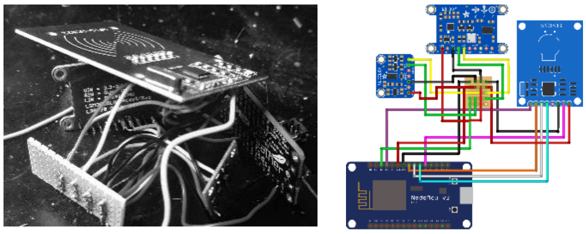


Figure 5 shows the soldering process of a NodeMCU 1.0 with an RFID reader, an LSM303DLHC, a VL53L1X sensor, and a circuit board.

At this point in the project an ESP8266 has been used; however, an ESP32[18] has been bought because this microcontroller is faster in processing than an ESP8266. Furthermore, a battery plugin is soldered onto the board, so the power supply is easier to handle. After soldering the sensors and the new microcontroller, the code did not work. It seemed the microcontroller entered a rebooting loop. For troubleshooting, all wires have been disconnected from the broad. Every sensor has been added separately, so the problem could be found. However, no problem has been found in the code if only file (ino-file) has been used. So, the code objects have been removed. After troubleshooting and soldering everything, more flexible wires have been used (see Figure 6).

[3] Radio-frequency identification (short RFID) refers to a technology that is capable of sending information wirelessly via radio waves.[46], [47]

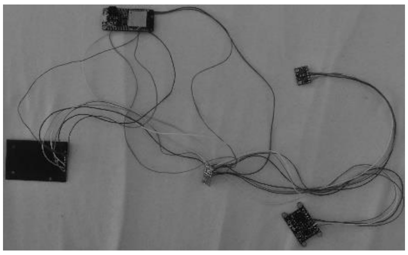


Figure 6 shows the soldering of every sensor with the microcontroller and a circuit board with more flexible wires.

Still, after implementing the acceleration sensor, the gyroscope, the RFID-RC522 chip did not work. That's why I^{2}C[19] communication has been tried out. This would have helped to structure the wires because then every chip would use the same communication protocol. However, it did not work out, so the UART[20] communication has been used. Even this did not work, and the SPI[21] did not work either with the ESP32.

Two major problems have been occurred after using the ESP32. First, the RFID reader did not work anymore, and second, no classes could be used. That's why the ESP32 has been removed and the ESP8266 1.0 has been used again.

After changing the microcontroller, soldering, and programming classes, the ESP8266 always started to enter the Software Watchdog mode. That's why the yield() function and the ESP.wdtDisable() function has been used to deal with the Software Watchdog[22]. This could not solve the problem, so it was a hardware problem. That's why the wires between the ESP and the sensors have been removed and later linked again with a fresh circuit board. This worked; however, the I^{2}C communication for the RFID chip did not, so the SPI communication protocol has been used again.

At the same time, the WIFI network has been reprogrammed because the sensor values must be shared with Pure Data. After weeks of trying to receive any information, another microcontroller has been added to the project and a network between two ESP8266 has been programmed.

The microcontroller with the sensors on it is the server and the other ESP8266 that is linked to a computer is the client. The server, that is linked to the abstract instrument shares its data with a client. The client is linked to a computer, so Pure Data can receive serial outputs. The whole code logic can be found in Figure 7.

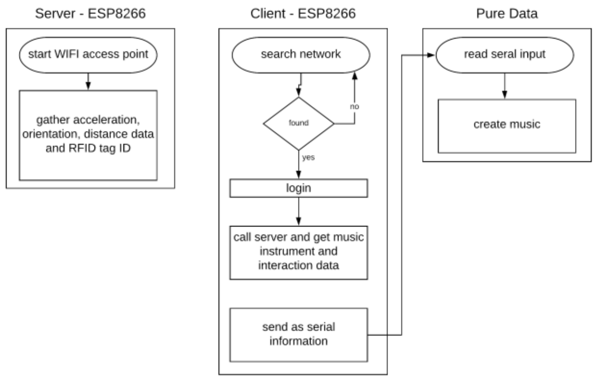


Figure 7 shows the interaction between programs. The server gathers the acceleration, orientation and distance values. Furthermore, it creates a wireless network. If the Client is connected to the network, it calls for the interaction data and processes them, so Pure Data can use the data to generate music.

The communication between the client and the server and the code processing takes around five seconds. This takes too long, so an ESP32 has been tested as a client. However, this could not fix the time delay. Without the RFID reader, the client can read the server messages faster (see Figure 8). However, the delay was around two seconds long.

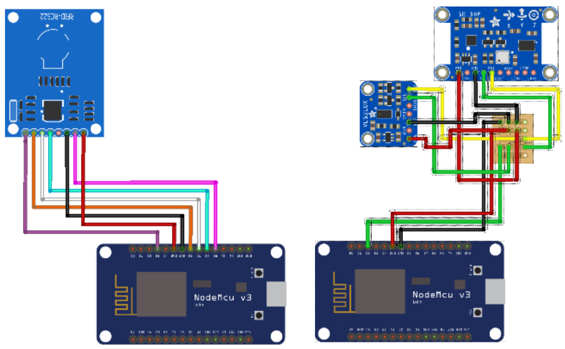


Figure 8 shows the wiring of the client (see left microcontroller) that is wired to an RFID reader and the server (see right microcontroller) that is wired to a circuit board and sensors like the LSM303DLHC and the VL53L1X.

With this the code process changed as well. The server created a wireless network, so anyone can enter it and call for the interaction data. The client tried to enter this network, and if it found it, it asks for the RFID tag and calls the server to get the interaction values. After gathering the data, it sends it to Pure Data (seen Figure 9).

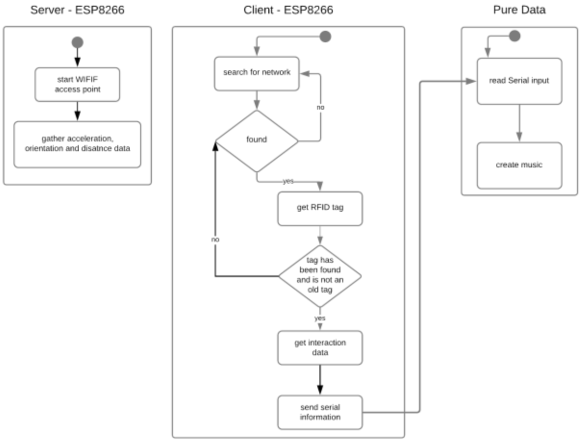


Figure 9 shows the new code processing. The client tried to find the network the server created and then reads the RFID tag and then calls for the interaction data from the server. After gathering all information, these information get send to Pure Data to generate sound.

After a consultation, all agreed that a delay of even two seconds would be too much for an interaction system that should have an immediate response. So, a WIFI network is no option or at least the HTTP protocol is not. That's why a serial connection was used instead. Only one microcontroller is used for the project again. This one receives distance, orientation, and acceleration data. In addition, the RFID reader has been linked to this microcontroller again, so it can detect RFID and NFC tags. However, the RFID tag is only read if a button[23] is pressed. The button is new and is used as an initiator. Only if the Button has been pushed once, the RFID reader reads the tag on the object. Furthermore, a USB chip has been wired to the ESP8266. Therefore, a USB TTL Serial cable has been used.[24] This chip uses UART for communication. For uploading code and receiving serial information over this TTL Serial Cable, another USB driver has to be installed[25]. After compiling the code, the FLASH and the RST button must be pushed before uploading the code. Then, the user must release the button RST on the microcontroller first and the FLASH button. After uploading the code, the RST button must be pushed.

In the end, one microcontroller is used to gather all the necessary information. It is liked to an RFID reader that is used to know the abstract instrument, an acceleration and orientation sensors to get the motion and a distance sensor to get the distance between the module and the user. A button has also been added, so it only reads the new RFID tag if it is pressed once (see Figure 10). After wiring and adding everything in the sensor module, it has been noticed that NFC tags cannot be recognized by the RFID reader anymore. The distance between the reader and the NFC tag is too large. That's why RFID cards have to be used to identify the abstract instruments instead.

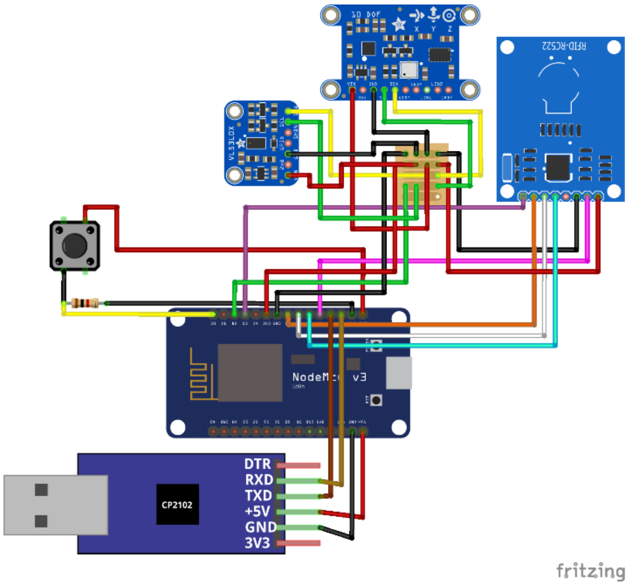


Figure 10 shows the final wiring of the NodeMCU 1.0 with TTL USB, RFID reader, LSM303DLHC, VL53L1X, a button, and a circuit board.

The code processing has changed while working on the wiring and removing the wireless network. One microcontroller is linked to all sensors and chips (see Figure 10). The button is used to call the RFID reader to get the tag ID. If a tag has been found it sends specific music data according to the tag ID to Pure Data (see Figure 11).

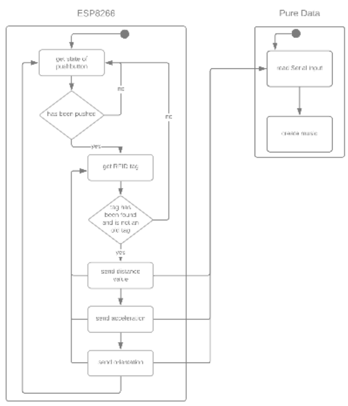


Figure 11 shows the code processing. One microcontroller is linked to all sensors and chips. If the button is pressed, the RFID reader reads the tag ID. According to the ID, the different interaction values are sent.

The final documentation with all their images and codes can be found on Github.[26] The changes can that have been made over the time can be found on Github as well.[27]

2 SOUND GENERATION

To understand sound waves and how sound can be created, an introduction of sound generation has been given. Therefore, the open-source platform Axoloti[28] has been used. In this session, the filters, pitches, or the envelope mode, for example, has been discussed and explained (see Figure 12).

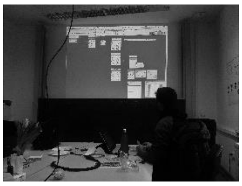



Figure 12 shows the music generation session. It included the Axoloti and how to program sound generation.

After this session, group members decided to use Pure Data[11]. A software solution has been prioritized because it felt more natural to the group using it. All group members have a background in programming. A hardware solution like the Axoloti was also discussed; however, this would add another hardware device users would need to buy. Pure Data instead is for free and open source.

At the beginning of the project, a wireless network between a microcontroller and Pure Data needed to be programmed. Wireless was the prioritized communication method because this would help to use the microcontroller with all necessary sensors freely. To develop such a network the patch MrPeach[29] had to be added. However, the group could not find out how the patch can be integrated even if many online sources have been found. Because of this problem, a wireless connection between Pure Data and a microcontroller like the ESP8266 could not be created, so a direct wireless connection was not possible.

That's why another microcontroller has been added to this project. This one is directly linked to the computer. The microcontroller receives the interaction data and sends it to Pure Data. Therefore, the serial port is used. To receive serial information the patch Comport[30] has to be loaded. This patch can be used to receive raw serial data, so this data can be used for music creation. However, this did not work because of the amount of data that has to be sent at once.

That's why another patch has been used to read data from the serial port. The object that can be used to receive MIDI data is called notein[31]. MIDI has been used because it was the only way Pure Data could receive information from a microcontroller. However, the user of the system must download and install two programs, LoopMIDI and Hairless.

LoopMIDI is going to create a virtual MIDI device. There is a default software in Windows that does it, but it often crashes and LoopMIDI is more stable.
The program Hairless is going to take the serial data and send it to the MIDI device.

In addition to these problems, there was another one that came up when instrument sounds needed to be added to the project. Abstract sounds like a sine wave sound were not difficult to generate; however, to generate instrument sounds, vst-files must be added in Pure Data. Those files are audio files that are used for the piano, cello, violin and flute sounds, for example. In order to play vst-files, plugins must be used. A plugin for Pure Data has been found and some example vst-files as well; however, the files could not be run.

As the last step, the final sound files had to be programmed and tweaked. Therefore, the final abstract instruments had to be defined. So, the group decided to present a chair, a stick, and a rope as the abstract instruments. The corresponding sound generation Pure Data files create plucked strings, snare drum, and phasor sounds.

By making sound with the chair, the acceleration and the gyroscope data can be used. This could be useful because the rotation and relative position of the chair can be found with the gyroscope sensor values, and the acceleration is used to find out the velocity of the chair movement. On the other hand, the distance, gyroscope, and the acceleration sensors can be used for the stick instrument. By interacting with a stick, many interaction gestures are possible, so a wide range of sensor values could be used to detect them. It could be used as a guitar, and the distance between the fingers and the end of the stick has to be detected or as a sword and the relative position and the velocity with the gyroscope and the acceleration has to be found. The same goes for the rope as an abstract instrument. The user could use it as a whip or a Lasso, for example, and that interaction possibilities have to be found. With these data, plucked strings, snare drum, and phasor sounds can be created with Pure Data.

3 DESIGN THINKING

In the meanwhile, the team also met to discuss how the sensor module could look like. The sensor module defines all sensors and the microcontroller and all other technical stuff that is included inside a case.

First, everyone tried to find their designs and ideas. Days later, the group met and discussed them. In this design thinking process, several ideas and drawings have been discussed. The shape but also the connection between the module and the instrument has been debated (see Figure 13).

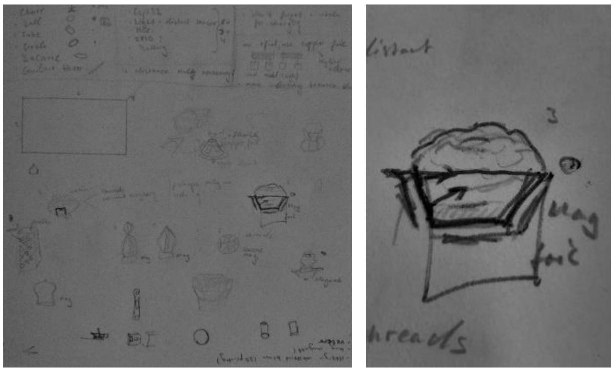


Figure 13 shows the sketched ideas of how the sensor module could look like and the favorite idea.

The group thought that the module should not have the shape of a common module like a rectangle or a cylinder. Instead, it should look interesting, so people want to interact and experience what it is. At the end of the meeting, it has been decided to design an abstract form of a brain. The brain concept is not linked to music. Instead, it has the meaning of including everything necessary to generate the sound. Other objects/the instruments/ the constraints that are attached to this module are like the body of the device. This, at least, was the intention and the metaphor as a concept (see Figure 13).

Also, the connection between the module and the constraints were part of the discussion. Two ideas were prioritized.
The first one was using magnets. This idea has a positive aspect that no connections have to be shown. However, this also is the biggest drawback because people would not know how to link everything (affordance issues). Furthermore, the magnets have to have a big magnetic effect; otherwise, the module would lose the connection to the object when the user is doing some fast movements.
Because of this problem, other linking mechanisms were tried to be found. Hook-and-loop fastener[4] was another idea. This not only provides the user a hint about how to connect devices, but the connection is also very strong, so even faster movements should not be difficult to handle. However, also there the direction of how to connect the constraint with the module is not defined.

After the meeting, anyone tried to model the abstract form of a brain version in a 3D program like Fusion 360[32] or Maja[33]. This helped each other to understand how the module could look like. Furthermore, it also helps to create a printable version later because it can be exported into a 3D printer readable file. A first version can be found in Figure 14.

Figure 14 shows the first version of a 3D model of the sensor module.

In another meeting, the 3D model has been discussed and what must be changed. The main problem with this model is that it is symmetry. This leads to the problem that users do not know how to connect it to an object. Since only one distance sensor is used, the direction of the module is important; otherwise, the interaction values are wrong.

Furthermore, the connection between the module and the object was part of the discussion. To hold everything in place in place, an asymmetric design has to be found. The base can be connected via cable ties, for example.

At the same time one of the team members did some further changes in the 3D model (see Figure 15). The main difference compared to the first version (see Figure 14) is the top. It is more flattened, so the height of the overall model shrinks. Furthermore, more space for the sensors has been added, so all devices fit in the sensor module. In addition, an opening and closing mechanism have been added, so the device can be debugged if needed.

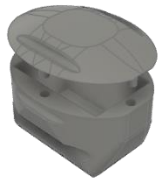


Figure 15 shows the second iteration of the abstract form of a brain.

In a third iteration of the sensor module design, some space has been added to the top, so the RFID chip could also be added there. This version has been 3D printed. The whole process can be found in Table 3.

[4] Hook-and-loop fasteners consist of two components. Two lineal fabric strips can be attached to each other's surfaces. The first component features tiny hooks, the second features smaller loops, so if they are connected, they are bind temporarily.

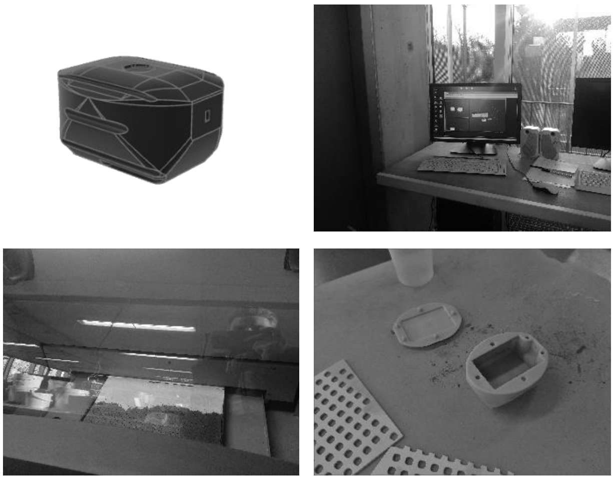


Table 3 shows images that show the printing process.

After printing out the sensor model, sensors have been checked if the sensors fit; some space could be removed. This led to a shrunken model. Furthermore, some sensors did not fit, so some bigger holes have been added. The sensors did not fit because there was a misunderstanding in the width and height values of some sensors. The datasheets used inches and not centimeters or millimeters. In addition, holes for the Micro-USB cable have been added. So, the battery that is linked to the microcontroller can be charged. Also, the space on the back of the top has been added again because the group decided to have the RFID reader on the ground. However, the print was hardly comparable with the module that has been designed on the computer (see Figure 16).

Figure 16 shows a new 3D printing.

In addition, the connection base (see Figure 17) that connects the object with the sensor module has been designed. However, it has not been printed because a new printer had to be found first. This base has four holes. These are used to wire the base to the instrument. Furthermore, a slider bar has been added. This holds the sensor model in place even if users move the model fast.

Figure 17 shows the base model that has been designed.

Because of using a server-client wireless network for the communication between two ESP8266 microcontrollers, a receiver module had to be designed. The client is placed in the inside of it. Later, the client is linked to the computer by a USB cable (see Figure 18).

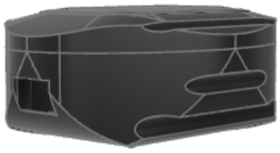


Figure 18 shows the receiver module for the ESP8266 client.

However, after all these changes in the programming part (see 1. Programming) and what kind of communication is used, the receiver module does not need to be printed yet.

The sensors module is printed with some slight changed. First and foremost, the top of the module is not around anymore. The reason for this is the new printer that has been used. It would be difficult for the printer and the outcome would probably look bad, so a flat top is printed instead. Some smaller changes in the body of the sensor module have been done as well. In printing process can be found in Table 4.

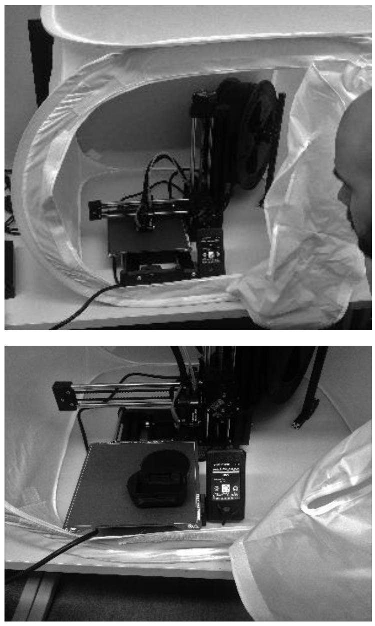
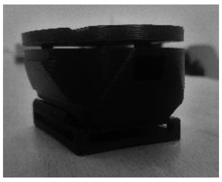


Table 4 shows the final printing process.

The printer uses Polylactide (PLA)[34]. This material shape change if the temperature is higher than around max. 50°C. The final sensor module can be seen in Figure 19.



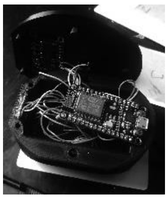


Figure 19 shows the final sensor module with all sensors and the microcontroller included.

4. BODYSTORMING

In order to get more inspiration on how to use the prototype and to find more constraints, a bodystorming session has been held. Like mentioned before, a bodystorming session has been done before (see 1. Programming); however, only the group participated. This time, more and more new inspiration has been found from people who have different backgrounds. In total, three students from the Bauhaus University have participated, two males and one female. The average age of these three people was 31.3. The first participant studies Human-Computer Interaction but also has a background in Design. The second, on the other hand, is studying Media Arts and Design and has a background in Music. The third participant is also studying Media Arts and Design. The bodystorming session has been held between 1 and 3 pm.

To prepare everything, the Human-Computer Interaction Lab has been chosen to hold the session. In the lab, participants can find a wide range of devices and interaction properties.

A brief introduction to the project has been given. Sensors have been defined, so the participants knew what kind of interactions are possible. In addition, the sensor module has been presented and explained how it works and how it was to be connected to generate sound. An example has been used to explain a bodystorming session and how the device could be used. Therefore, any device in the room could be used to come up with new ways to interact with the sensor module.

In total, every participant had three sessions of 10 minutes. After this time, the participants and the observer talked about their ideas (see Table 5 in the Appendix).

There were some good ideas, some ideas which the group had originally thought of but did not focus on those in the end and some ideas which were out of the scope for this project, e.g. user 2 had an idea where the sensors are placed inside the surface or floor; however, the project is based around the fact that the sensor module is mobile and the data is gathered by the sensor modules movement, so it did not fit in the original idea of the project. Another idea from user 3 which was out of our project scope is to use vibrations. For this, an additional sensor has to be included. There were few ideas where the sensor module is placed inside a ball and that ball, in turn, is moved around, thrown or used to play with another person. However, the sensor module is fragile, and a solution for safety could not be found, perhaps foam material or Styrofoam could be used, so the module is completely snug inside which in order does not allow the sensor module to move and also gives it protection from hard impacts.

Some nice ideas that came out of this session were having a sensor module in a ball that is attached to a string and the other end of the string is attached to a paddle. Another idea was that the sensor module is attached to a rope and two people hold the rope and generate waves. Another idea was when a sensor module is inside a stress ball and that ball can be squeezed or thrown around.

CONCLUSION

In this project, many problems occur when programming the music generation or the microcontroller: the RFID, wireless connection and MrPeach, the time delay of the wireless communication, the loading of vst-files.

The RFID-RC522 reader can technically be used via three different serial protocols (UART, I^2C, and SPI[21]). All protocols have been tried but none of them worked with the ESP32 Feather. After connecting every pin correctly, the RFID reader did not respond. After two weeks of working on it, all agreed that the ESP32 should not be used anymore. Instead, the NodeMCU ESP8288 is used. With this microcontroller, everything worked well. The mistake that has been made was never found.

The sound generation over WIFI stopped the group for a long time as well. The MrPeach patch could not be included properly, so a second microcontroller has been added as a receiver. A wireless network between the ESP that was attached to the constraint/instrument and the ESP that is directly linked to the computer has been implemented to solve this problem. In this client-server network, the client received the interaction values from the server. The client sends serial data over a cable. However, the Comport patch that should have been used for this purpose cannot handle these numbers of data at once, so the notein object has been used. Later the wireless network has been removed because the delay was too long.

The problem that kept us from using a wireless connection was the communication delay. The client needed too much time to receive the interaction data from the server. To solve the problem, the ESP32 has been used as a client; however, there was no improvement. In the end, a longer cable has been used to transfer the information. However, there are ways to overcome this problem.

One of these solutions could be code/ software. Perhaps the used libraries are slow because it should not take so long. Libraries like ESP8266WiFi.h[35],

ESP8266WiFiMulti.h[36] and ESP8266HTTPClient.h[37] should be included and tested. Also, the ESPAsyncWebServer library[38] could be tried out.
Another solution to this problem could be a different sensor. The NRF24L01 chip[39] could be tested, for example.

Adding vst-files to Pure Data was a major problem, and several days had to be used to find the problem without much success. A solution to this problem was never found; otherwise, there have been found some instructions on the Internet.

However, even with these many problems, the aim could be reached, and a working prototype could be developed.

REFLECTION

The class and its final project were interesting, the group agreed. The author came across many different topics and problems that had to be solved in a given time. Before working on the project, the class helped to learn more about tangible user interaction and Human-Computer Interaction. Still, the author would say that the project should start sooner by finding the final project idea and working on it. Before working on this project, the project meetings were not as productive as they could be. Still, this also gave time to get to know each other and build a group. A solution for this could be a project that includes two semesters. Then more theory could be taught and more time for a proper idea is given. In a class called “Idea meets technology” students had a whole semester to find a good idea. This additional time could be used to read about tangible user interaction approaches and design theory and also about finding an idea.

The project idea was never a question. We all agreed which this was the best idea, so there was no disagreement in this group. Also, another discussion like the design discussion was discussed friendly and with respect. This shows the good communication the team had.

In this project, the author had the chance to touch many different topics at once, like 3D printing, music generation and how it works and programming of different microcontrollers and sensors.

Every week and near to the deadline every day they could talk about their achievements and what they did. Splitting the team is sufficient; however, this also creates a problem because no one knows everything about the project.

Working in a project like this allowed us to have valuable experiences not only in situations where everything worked as planned but also in some bad cases whereas things were happening not as expected and how we faced those problems.

Most problems with the electronics could be fixed even if it took a bit of time. Soldering all sensors and devices should be done later in the project. Changes in the sensors or which microcontroller use which sensors appears often in the middle of the project and if the sensors are soldered already, it takes a long time to unsolder everything. However, this gave me more practice. The group could deal with problems well and always tried to find a solution by themselves first. Still, not every problem could be fixed sufficiently.

The wireless network was a big problem in the programming process. The delay of around five seconds could not be shrink to an immediate response of the system. Regarding the documentation, the time delay should have been tested more precisely. Therefore, the function millis()[40] should have been used to test the HTTP response time.

The number of participants with different backgrounds in a bodystorming session was too low. There was not too much overlap in their opinions; however, we could find out many different aspects and how we should generate the sound. Still, the bodystorming session gave the group some further inspiration for attachments and constraints. However, three participants are perhaps not enough and more different participants with different backgrounds.

REFERENCES

[1] B. Ullmer, H. Ishii, and R. J. K. Jacob, “Token+constraint systems for tangible interaction with digital information,” 2005.

[2] O. SHAER, N. LELAND, E. H. CALVILLO-GAMEZ, and R. J. K. JACOB, “The TAC Paradigm: Specifying Tangible User Interfaces,” 2004.

[3] B. Ullmer and H. Ishii, “Emerging Frameworks for Tangible User Interfaces,” Cambridge, 2000.

[4] S. Harrison, D. Tatar, and P. Sengers, “The Three Paradigms of HCI,” Blacksburg, Ithaca, 2007.

[5] O. Shaer and E. Hornecker, “Tangible User Interfaces: Past, Present, and Future Directions,” Glasgow, 2010.

[6] “How Accelerometer works? Interface ADXL335 with Arduino,” *Last Minute Engineers*, 2020. [Online]. Available: https://lastminuteengineers.com/adxl335-accelerometer-arduino-tutorial/. [Accessed: 20-Feb-2020].

[7] Dejan, “MEMS Accelerometer Gyroscope Magnetometer & Arduino,” *howtomechatronics*, 2018. [Online]. Available: https://howtomechatronics.com/how-it-works/electrical-engineering/mems-accelerometer-gyrocope-magnetometer-arduino/. [Accessed: 20-Feb-2020].

[8] "Ultrasonic Ranging Module HC - SR04," *Elecfreaks*. [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf. [Accessed: 09-Jan-2020].

[9] Dejan, "Arduino and MPU6050 Accelerometer and Gyroscope Tutorial," *howtomechatronics*, 2019. [Online]. Available: https://howtomechatronics.com/tutorials/arduino/arduino-and-mpu6050-accelerometer-and-gyroscope-tutorial/. [Accessed: 20-Feb-2020].

[10] Cbenson, "Arduino 5 Minute Tutorials: Lesson 7 - Accelerometers, Gyros, IMUs," *Robotshop*, 2019. [Online]. Available: https://www.robotshop.com/community/tutorials/show/arduino-5-minute-tutorials-lesson-7-accelerometers-gyros-imus. [Accessed: 20-Feb-2020].

[11] "Welcome To The Pure Data Site!," *Institute of Electronic Music and Acoustics*, 2020. [Online]. Available: https://puredata.info/. [Accessed: 04-Jan-2020].

[12] "Getting Started with Arduino UNO," *Arduino.cc*, 2020. [Online]. Available: https://www.arduino.cc/en/Guide/ArduinoUno. [Accessed: 15-Feb-2020].

[13] M. Yuan, "Getting to know NodeMCU and its DEVKIT board," *IBM Developer*, 2017. [Online]. Available: https://developer.ibm.com/tutorials/iot-nodemcu-open-why-use/. [Accessed: 04-Jan-2020].

[14] "3-Axis Digital Compass IC HMC5883L," *Honeywell*, 2014. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/HMC5883L_3-Axis_Digital_Compass_IC.pdf. [Accessed: 05-Feb-2020].

[15] "Ultra-compact high-performance eCompass module: 3D accelerometer and 3D magnetometer," *STMicroelectronics*, 2014. [Online]. Available: https://www.st.com/resource/en/datasheet/DM00027543.pdf. [Accessed: 09-Jan-2020].

[16] "A new generation, long distance ranging Time-of-Flight sensor based on ST's FlightSense™ technology," *STMicroelectronics*, 2018. [Online]. Available: https://www.st.com/resource/en/datasheet/vl53l1x.pdf. [Accessed: 09-Jan-2020].

[17] "Security Access using MFRC522 RFID Reader with Arduino," *Random Nerd Tutorials*, 2020. [Online]. Available: https://randomnerdtutorials.com/security-access-using-mfrc522-rfid-reader-with-arduino/. [Accessed: 19-Feb-2020].

[18] "Adafruit HUZZAH32 – ESP32 Feather Board," *Adafruit*, 2017. [Online]. Available: https://www.adafruit.com/product/3405. [Accessed: 19-Feb-2020].

[19] "I2C Info – I2C Bus, Interface and Protocol," *i2c.info*, 2020. [Online]. Available: https://i2c.info/. [Accessed: 23-Mar-2020].

[20] Jimblom, "Serial Communication UARTs," *Sparkfun*, 2020. [Online]. Available: https://learn.sparkfun.com/tutorials/serial-communication/uarts. [Accessed: 23-Mar-2020].

[21] MIKEGRUSIN, "Serial Peripheral Interface (SPI)," *Sparkfun*, 2020. [Online]. Available: https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all. [Accessed: 23-Mar-2020].

[22] N. Santos, "ESP8266: Watchdog functions," *Techtutorialsx*, 2017. [Online]. Available: https://techtutorialsx.com/2017/01/21/esp8266-watchdog-functions/. [Accessed: 05-Feb-2020].

[23] F. Schäffer, *Sensoren am Arduino*, 1.Auflage. Elektor-Verlag GmbH, 2016.

[24] "FTDI Cable – USB to RS232 Converter," *Components 101*, 2018. [Online]. Available: https://components101.com/cables/ftdi-cable-usb-to-rs232-converter. [Accessed: 22-Feb-2020].

[25] "Delock USB 2.0 to Serial TTL Converter with 6 open wires 1.8 m (3.3 V)," *Delock*, 2020. [Online]. Available: https://www.delock.com/produkte/G_83117/merkmale.html. [Accessed: 12-Mar-2020].

[26] C. Heiden, "Final Documentation," *GitHub*, 2020. [Online]. Available: https://github.com/ChrisHeiden/Doc_Tangible_Interaction. [Accessed: 23-Mar-2020].

[27] C. Heiden, "Projectflow," *GitHub*, 2020. [Online]. Available: https://github.com/ChrisHeiden/Tangible_Interfaces_Project. [Accessed: 23-Feb-2020].

[28] "Axoloti," *Axoloti*, 202AD. [Online]. Available: http://www.axoloti.com/. [Accessed: 15-Jan-2020].

[29] "Mr.Peach," *PureData*, 2020. [Online]. Available: https://download.puredata.info/mrpeach. [Accessed: 19-Feb-2020].

[30] "Comport," *PureData*. [Online]. Available: https://puredata.info/downloads/comport. [Accessed: 08-Feb-2020].

[31] "Midi," *PureData*, 2020. [Online]. Available: http://write.flossmanuals.net/pure-data/using-midi/. [Accessed: 19-Feb-2020].

[32] "Fusion 360 - Integrierte CAD-, CAM- und CAE-Software.," *Autodesk*, 2019. [Online]. Available: https://www.autodesk.de/products/fusion-360/overview#banner. [Accessed: 05-Dec-2019].

[33] "Maja - Software für Animation, Modellierung, Simulation und Rendering in 3D," *Autodesk*, 2020. [Online]. Available: https://www.autodesk.de/products/maya/overview. [Accessed: 15-Jan-2020].

[34] R. G. HILL, "Biomaterials, Artificial Organs and Tissue Engineering," *Science Direct*. [Online]. Available: https://www.sciencedirect.com/topics/medicine-and-dentistry/polylactide. [Accessed: 20-Feb-2020].

[35] I. Grokhotkov, "ESP8266WiFi.h," *GitHub*, 2019. [Online]. Available: https://github.com/esp8266/Arduino/blob/master/libraries/ESP8266WiFi/src/ESP8266WiFi.h. [Accessed: 23-Mar-2020].

[36] M. Sattler, "ESP8266WiFiMulti.h," *GitHub*, 2019. [Online]. Available: https://github.com/esp8266/Arduino/blob/master/libraries/ESP8266WiFi/src/ESP8266WiFiMulti.h. [Accessed: 23-Mar-2020].

[37] Jeroen88, "BasicHttpClient.ino," *GitHub*, 2018. [Online]. Available: https://github.com/esp8266/Arduino/blob/master/libraries/ESP8266HTTPClient/examples/BasicHttpClient/BasicHttpClient.ino. [Accessed: 25-Jul-2019].

[38] "ESPAsyncWebServer," *GitHub*, 2019. [Online]. Available: https://github.com/me-no-dev/ESPAsyncWebServer.

[39] "nRF24 Series," *Nordic Semiconductor*, 2020. [Online]. Available: https://www.nordicsemi.com/Products/Low-power-short-range-wireless/nRF24-series. [Accessed: 23-Mar-2020].

[40] "millis()," *Arduino.cc*, 2019. [Online]. Available: https://www.arduino.cc/reference/en/language/functions/time/millis/. [Accessed: 23-Mar-2020].

[41] "UXD Method 11 of 100: Bodystorming," *Typepad*, 2011. [Online]. Available: https://dux.typepad.com/dux/2011/04/uxd-method-11-of-100-bodystorming.html. [Accessed: 24-Feb-2020].

[42] B. Martin and B. Hanington, *Universal Methods of Design*. Rockport Publishers, 2012.

[43] A. Endmann, H. Fischer, and M. Krökel, *Mensch und Computer 2015 - Usability Professionals: Workshop*. De Gruyter Oldenbourg, 2015.

[44] C. Valens, *Mikrocontroller verstehen und anwenden : schnell und einfach mit Arduino und Elektor-Shield*, 1. Aufl. Aachen: Elektor-Verl., 2014.

[45] T. Ungerer, *Mikrocontroller und Mikroprozessoren : mit 41 Tabellen*, 2., überar. Berlin [u.a.]: Springer, 2007.

[46] M. Bolic, D. Simplot-Ryl, and I. Stojmenovic, *RFID Systems: Research Trends and Challenges*. Wiley, 2010.

[47] C. K. Haley, L. A. Jacobsen, and S. Robkin, *Radio Frequency Identification Handbook for Librarians*. Libraries Unlimited, 2007.

APPENDIX

ideas / Users	User 1	User 2	User 3
1	Attaching the sensor module to a rope and two people will hold the rope on each end. They will generate waves, like a Sine wave, on the rope which will trigger sound generation	The sensor module is inside a ball that is attached to a string. The user can roll the ball on a surface or spin the string in a circular motion to wrap it around one's arm	The sensor module is attached to a steel drum and a brush is used to strike the drum. The vibrations caused will trigger the sensor module and also the distance of the brush from the sensor module will be taken into account when generating sound
2	Having two sensor modules, one will be on a stick which will be used as a guitar and the other one will be in the user's hand. Every time the two modules touch, it would generate sounds. The idea was inspired by Guitar Hero	The sensor module is inside a ball, and the ball is held in place by using a cup-like object. This allows the ball to be rotated like a trackball	The sensor module is inside a ball and the ball will be covered in bubble wrap. The popping of the bubble wrap will activate the sensors and generate sound
3	The sensor module will be inside a small and light ball, which would be thrown around using sticks with a net, similar to lacrosse, and depending on speed/acceleration of the ball, the sensor module would generate different sounds.	The sensor module is inside a ball which is similar to a stress ball. Users can squeeze it and throw it which will bounce back.	The sensor module is attached to a chalkboard and the vibration that is generated while writing with a chalk will be used to create sensor modules.
4	The sensor module is inside a ball and the ball is attached to a string. The other end of the string is attached to a paddle, similar to a table tennis paddle. The aim is to hit the ball in the target which could be the center of the paddle, which would generate certain sounds and every time the user hits the ball on the other part of the paddle, the generated sound is different	Install the sensors on a surface or floor and when an object rolls on it, or someone steps on it, it triggers the vibration sensor. The intensity will dictate how the sound is generated	

Table 5 shows the idea of the participants in every session.