
Appendix A: Personal Reflections

Emma van Zoelen

The reason I chose to work on the Smart Mobility project, was because I am interested in complex and intelligent technologies and how they can be applied in society. Autonomous vehicles can of course be considered such a technology. Connected to this, one of my main goals for the project were to keep a constant societal focus. Next to that, my other main goal was a teamwork goal, since it would be my first group project in two years.

For the societal focus goal, I mainly wanted to design for trust and user experience, through literature and field research. I also wanted to write a critical scenario for the future, how our design would fit in a future society. As concrete as that, I didn't reach my goal. During our process though, we did design for the future, requiring us to constantly review our assumptions based on the current situation. I tried to keep the team focused on this societal aspect. Also, I did literature research on designing for trust as well as multi-user interaction, and I feel like I gained more awareness on how autonomous technologies might be implemented in society.

In terms of the teamwork goal, I experienced more difficulties than I expected, which is also part of the reason why I didn't manage to work on the societal goal a lot. Doing a project in a team, especially with half of the team coming from other universities, was quite frustrating sometimes. A lot of time went into long discussions, getting everyone on the same page and making sure everyone understood what was expected of us. I believe this couldn't really have been

avoided, and in the end we managed to use everyone's qualities in an effective way. Part of this was achieved because I constantly watched our planning and task division, trying to keep an overview of the whole process, through which I also achieved the goal as I stated it in my PDP: to design in such a way that everyone's personality would come back in our work.

However, I did notice that I fell back in a pitfall I thought I had overcome in previous group projects. I always used to be a bit of a control freak, trying to take up as many tasks as possible and checking the tasks of the others. In previous projects I learned to be more relaxed in this. However, since I had worked on my own for so long, I noticed that I started taking up tasks that I didn't want to learn about, but felt like I had to do them because then at least they would be done well. I know this isn't a right approach, and now that I'm aware of this pitfall again I will try to focus more on my own learning in the future and be more relaxed about other people's responsibility.

Because of the tasks I took up though, I learned a lot about electronics and the realization of a prototype. Especially in the latter, I found that it is very hard to work on such tasks as a team, and it requires effort to create a good production flow. Often we were waiting for each other to finish. In future products, this is an aspect of the process that might actually need more management.

Last, I gained some insights in design processes related

to group work versus individual work. I found that, while working individually, I can work harder because I make my own decisions. However, I do get stuck sometimes. In groups, discussion and reaching compromises helps to quickly generate concepts and move on, but sometimes you have to wait for each other to continue.

All in all, I learned a lot this semester, though in different areas than I expected beforehand. I believe that my newly gained insights will especially help me to deal with group projects in the future while paying attention to my own development more.

Feng Ye (Steven)

Smart mobility was a new, even a little strange but attractive topic for me when I first started to choose my squad. During my former education experience, I learnt a lot of separate skills like interaction design, conceptual design, service/ business design and UI design, but I never spent a whole semester researching on a leaning-edge but practical topic which is also focused by the world. I was nervous at first, then I noticed that it is a brilliant opportunity for me to bring all my skills together and practice them. I always believe that a new environment could help one make progress, and it turns out so actually. The project taught me skills that complementary to what I've learnt before.

After a semester's learning, I find out what I have learnt is far more beyond what I imagined. I learnt from wise coaches, experts and excellent colleagues, I also learnt from an highly organized, design process, which is really valuable for me, it helps me to establish an organized and clear thinking system which I was lack of. It is a group project so the working pattern is also different from doing a personal project. You need to learn to cooperate with

others, communicate with the team as much as possible. I have to say, the way people think and talk is different between Dutch and Chinese. I read books and articles like "Robert's Rules Of Order"(unfinished) or "10 tips for meeting" and tried to make progress, but when I tried to express myself, my limited language ability drags back my heel, I have to use sketches to express my ideas sometimes - luckily, I am in design faculty, and luckily, I have very supportive colleagues who generously offer me help and always listen to me with patience. Another thing I learnt from this group project is that members should be clear of their positions in the team, you should take over the tasks that you want to learn or where you are good at, so that not only the team could have a higher efficiency, but yourself could also benefit from it.

In the final prototype I was in charge of the interaction of the mainhub and the nerves, which is mainly the light behaviors. I regard light behavior as a silent communication way and I am interested in making a "tacit agreement" between people and product so that people could understand the product naturally. Light is an excellent tool, people understand and influenced by light in a way even they do not noticed. I made researches and tried to combine the light behavior with some specific moving patterns that people familiar to help them understand. Like "crawl - slow - defensiveness" or "flash - fast - assertiveness". And I would say I really enjoy that thinking and creating process.

However, there are some things I tried to do but are not satisfying enough, or problems that I did not aware of. For example, I wanted to practice my user test skills more but did not being proactive to find opportunities for it. I always set myself limitations because of my cowardice and inertia. These are problems that I would pay more attention on and have to overcome in the coming personal projects.

In the end, I would say I am grateful that I could experience such an amazing project, which gives me an opportunity to learn with joy.

Sander Bos

Coming from 'Creative Technology' at the University of Twente I had never done anything with mobility or cars in general, although it interests me. I find cars very fascinating. They have a strong influence on society and people and so when I read about this smart mobility project my choice was made.

The smart mobility project was my first project and collaboration with people at the TU/e so I didn't know what to expect. Everything was new and I didn't know the habits, quality and how things go. Especially in the first quartile this was quite a struggle for me which kept me busy till the end of the semester. I saw this struggle quite heavy in the team. In the beginning I had a hard time working together because I came from a different background. However, when time progressed, the smoother and better our collaboration went which I saw back in our concepts and results.

Since I'm quite good in organizing and structuring I offered myself as being the group leader which was also in line with my PDP. I wanted to learn to be a better (project) leader and improve my skills in organizing and motivating people. But although I was the official project leader, in practice I felt like I wasn't. I maybe held myself back too much because I didn't know what I could expect from the others and how I should lead this first project at a new university because I didn't know what was expected from us. Fortunately I could provide leading and heading to the group in moments where nobody really knew what to do or what was expected. I may not have been the project leader my team members or I wish I could have been, but

I have done what I could do, as well as possible.

As I mentioned before I studied 'Creative Technology' before and during this semester and especially during the project I learned much about how designers work, produce and present their work. My bachelor was far less focused on design and so I was grateful to learn how to work as an industrial designer as well as how to present your product(s) in a professional way. I will definitely use these new skills in future projects and products.

In my PDP I mentioned my interests in Virtual Reality and my plan to include this (if possible) into the project. In the beginnings of the project I was quite positive this would work out well, but unfortunately in the end it didn't fit into the project to include some form of Virtual Reality. Fortunately I have approval to do research regarding Virtual Reality next semester, so I can still develop my skills.

The last thing I would like to address is that I felt totally lost during this semester. I decided to come to Eindhoven because my plans to study VFX didn't work out and so my spirit, motivation and vision were totally lost. I didn't know anymore what I wanted to do during my master and what heading I would like to go. As a team we spent a lot of time together with the project and throughout the semester I slowly got my motivation and vision back. Working on a design project and dive into the user and come up with a product helped me to get back on track. Also learning new ways of how designers think, build and present their work inspired me a lot. Building the concepts and seeing how nice everything came together in the end greatly motivated me and made me think about my future heading.

I believe I learned and developed myself in more subtle ways I didn't expect and couldn't foresee at the beginning of the semester. I developed myself in design thinking and learned new ways of approaching a project from

an industrial designer point of view. For me personally, I would have liked to develop myself a little more regarding my PDP, but considering the circumstances I'm quite happy with my progression and I look forward to the next semester.

Laure Peeters

For my M1.1 project I decided to choose a topic that was completely different from everything I did in my bachelor. My main goal for this project was to just see what would happen if I were to engage in a topic that doesn't necessarily relate to my PI&V. I saw it as a 'last test' for both my PI&V and specialisation expertise areas (EA) before starting with my individual master projects. To provide myself with a bit of guidance I set three more specific goals related to the project.

- Get a clearer idea of design business by focussing on the business aspects of the project.
 - Test my perceived focus EA's (B&E, MD&C) by writing a reflection on how I expressed these within the project.
 - Develop my leadership and project management skills.
- In this reflection I will elaborate on the different goals and how I will use my gained knowledge in the future.

During the project I devoted time to market research. Since we were designing in a future context, the market the design needs to fit into does not exist yet. Therefore looking into trends, developments and market analysis' related to the used technologies was key to help us create a vision for this future market. I discovered that incorporating this research was very valuable for our design process and final design. It gave us a clear context for the design that helped in making design decisions leading to an overall more substantiated design. In future projects I want to keep on incorporating this kind of research from the start of the project in order to create more focused, substanti-

ated and suitable designs for the targeted market.

During the project I found that my focus was on B&E and C&A more so than MD&C. I think that this is mostly because the project didn't contain much content related to MD&C. However I take a lot of interest in giving neat and organised presentations. I took care of the look of the mid term presentations, designing a brand identity for our project and the video that communicates our context and concept in a concise way. I consider this step of summarising all the gathered information and communicating it to an audience in a clear way an essential element of the design process. It makes the entire project look and feel more professional. I want to keep on improving my skills in developing a fitting brand identity for my designs, this will support me in giving convincing presentations which will help me greatly in my future career.

Being in a team with mostly leading personalities I discovered that I am very capable of letting go of control, which I was quite proud of. I would only take the lead in situations where decisions had to be made, after a time of exploration, to keep the team moving. Management wise I focus mostly on creating planning and making sure that the right evidence is available to support our design decisions. I discovered that by not having to take up all of the leading/managing roles in the team I was able to deliver more in-depth/finished content and focus more on my personal learning goals.

This project has mostly been a confirmation of the ideas I had of the direction I want to take during my master and my professional identity. I found that in a context that is unfamiliar to me I am still able to apply my strengths and professional identity.

Unexpected discoveries were that I tend to draw more toward the C&A expertise area, however as explained

before this may also have been due to the nature of the project. Besides that I found that I really enjoy working in a team context. I think that working in a team allows me to let go a bit more, because we have a shared responsibility. I do feel very responsible about my parts of the project and tended to have doubts about whether I contributed enough to the team. Looking back I think that this mostly had to do with the fact that I was exploring a new approach to the design process. It turned out to be very valuable for the overall outcome of the project and it allowed me to learn new skills. I think that working in a team provides me with more freedom to learn and explore, a more relaxed feeling in the design process and a healthy drive to do the best possible for the team, more so compared to doing individual projects. For my upcoming individual projects I want to gather enough partners around me to simulate a similar environment. I feel that it helps ground me as to how the quality of my work is (which is usually fine when I feel that it needs to be better) and allows me to be more relaxed in my work while still being very motivated.

Appendix B: Persona

de Ruijter autonomous car



Members:

a 42 yrs old father Erik
a 39 yrs old mother Jasmine
two children of 11 and 8 yrs old

Education background:

both father and mother are well educated, the two kids are in primary school.

frequently visit places:

Erik' s work place in the city centre
Jasmine' s part-time job in city centre

kid' s school 10km far from home

Yearly revenue: €50,000

location: suburb of Eindhoven

*"On a typical day we eat breakfast together at around 7.30 after which father and the kids leave to either their work or school. I leave 2 hours later at 11.00. " - Mother Jasmine.
"Mother arrives home at around 15.00 while I and my brother arrive at home around 1 hour later at 16.00."
"At 17.00 father is done with work and arrives home at around 18.00. At night father leaves again to meet up with his friends at their favorite bar just outside of the city. He usually takes the taxi since he wants to drink a few beers. At midnight he returns, chats a little with mom and they go to bed. " - kids*

Current Status:

The father is the breadwinner and is the most busy one

The father' s driving time is longest in the family, sometimes has some drink in a bar

The mother deals with home stuffs, driving skill is not satisfying although she should drive to town centre time by time

Dilemma

Too little time for the family to gether together

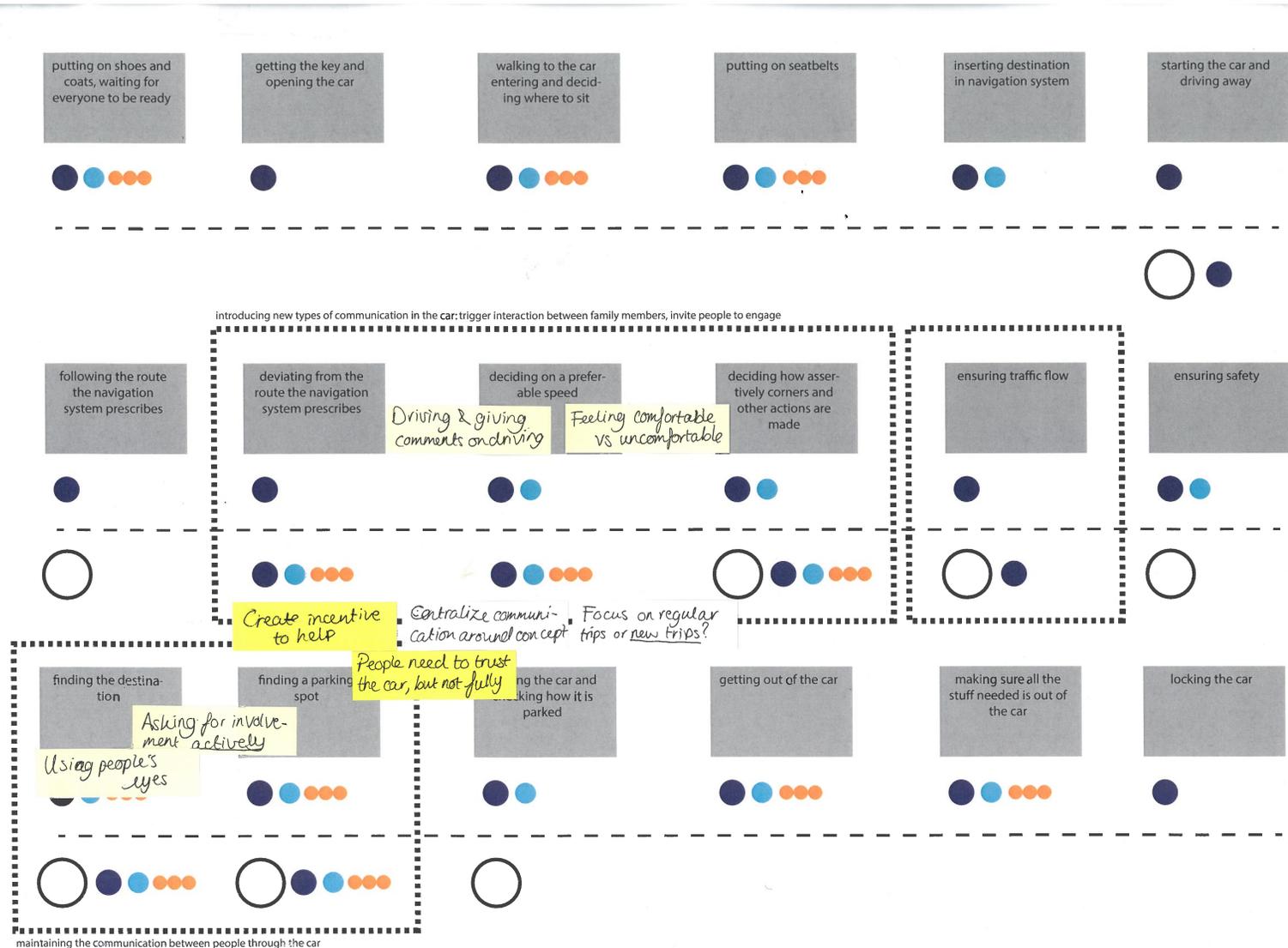
The breadwinner has no self-time, hobbis are limited, once he drink he should take a taxi, which is not quite convinent.

The school distance is too far for children that the everyday-trip costs too much time and energy for the kids

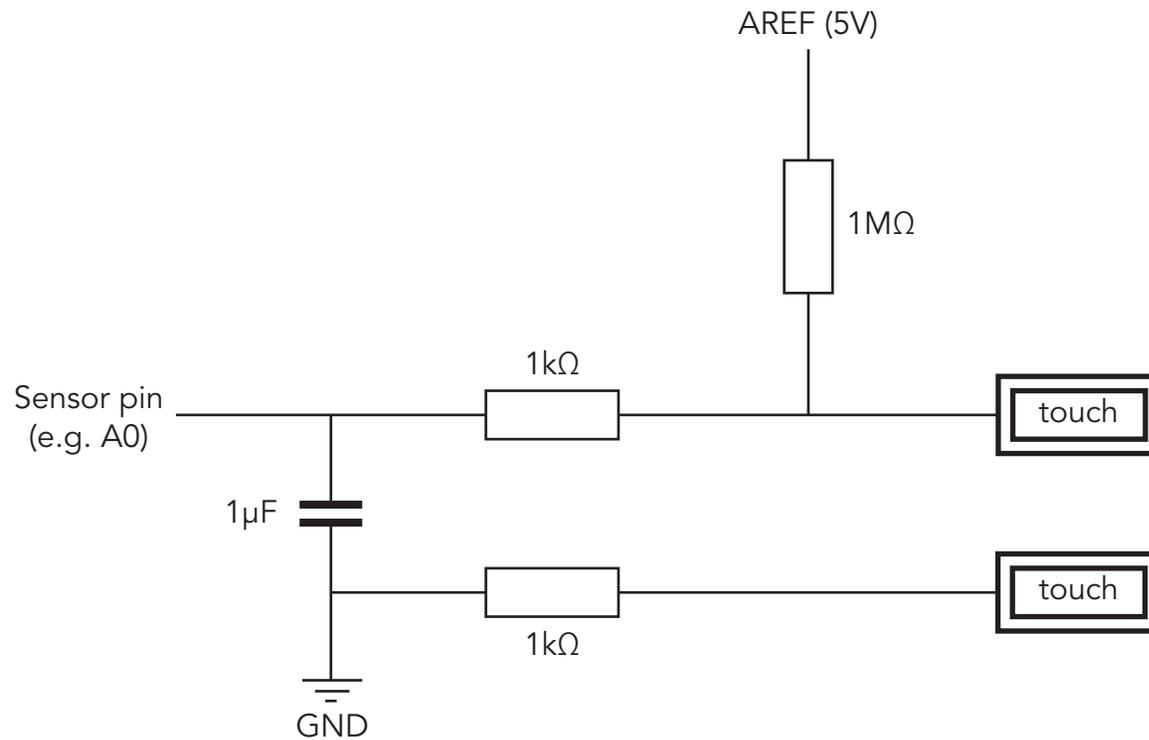
Goals:

More time with family, the father participates in the growth of children, more communication with them, The mother has more self-time and social more with friends, save the time for children so that they can focus more on study and ralex.

Appendix C: Situation Analysis



Appendix D: Circuit Stroke Sensor



Appendix E: Code Callon

```
#include "Adafruit_NeoPixel.h"
#ifdef __AVR__
  #include <avr/power.h>
#endif

#define PIN1      2
#define PIN2      3
#define PIN3      4
#define PIN4      5
#define NUMPIXELS 39//29

Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS,
PIN1, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel pixels2 = Adafruit_NeoPixel(NUMPIXELS,
PIN2, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel pixels3 = Adafruit_NeoPixel(NUMPIXELS,
PIN3, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel pixels4 = Adafruit_NeoPixel(NUMPIXELS,
PIN4, NEO_GRB + NEO_KHZ800);

float r1=0;
float r2=0;
float r3=0;
float r4=0;
float r5=0;
float r6=0;
int flag = 1;
char c;
float m1=0;
float m2=0;

float m3=0;
float m4=0;
float m5=0;
float m6=0;
int initial=0;
/*-----mainhub down-----*/
#include <SoftwareSerial.h>
// software serial #1: RX = digital pin 10, TX = digital pin
11
SoftwareSerial portOne(6, 7);

// software serial #2: RX = digital pin 8, TX = digital pin
9
// on the Mega, use other pins instead, since 8 and 9
don't work on the Mega
SoftwareSerial portTwo(8, 9);

// software serial #3: RX = digital pin 6, TX = digital pin
7
SoftwareSerial portThree(10,11);

//software serial #4: RX = digital pin 4, TX = digital pin 5
SoftwareSerial portFour(12,13);

void setup() {
  // Open serial communications and wait for port to
open:
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for native
```

```

USB port only
}

// Start each software serial port
portOne.begin(9600);
portTwo.begin(9600);
portThree.begin(9600);
portFour.begin(9600);
/*-----led-----*/
#ifdef __AVR_ATtiny85__
if (F_CPU == 16000000) clock_prescale_set(clock_
div_1);
#endif
pixels.begin();
pixels2.begin();
pixels3.begin();
pixels4.begin();
}

void loop() {
// By default, the last initialized port is listening.
// when you want to listen on a port, explicitly select it:
portOne.listen();
Serial.println("Data from port one:");
// while there is data coming in, read it
// and send to the hardware serial port:
while (portOne.available() > 0) {
char c = portOne.read();
Serial.write(c);
/*-----mainhub down-----*/
if(c=='1'){
Serial.println("1");
delay(100);
int j=0;
if(m1>=9){m1=m1-9;}
if(m3>=m1)m2=m3/3;
else m2=m1/3;

```

```

m3=m3+9;
int min1=min(m1,m2);
int min2=min(m2,m3);
int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels.setPixelColor(i, pixels.Color(0,j*5,j*15)); // Moder-
ately bright gre en color.
pixels.show(); }
for(int j=85;j>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){

// pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
int v1= max(0,m1);
int v2=max(j,m2);
int v3=max(j,m3);
pixels.setPixelColor(i, pixels.Color(v1,v2,v3)); // Moder-
ately bright gre en color.
r1=m1;r2=m2;r3=m3;
pixels.show();
pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*
0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
pixels3.show();
pixels4.setPixelColor(i, pixels4.Color(r1*0.
4+r4*0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately
bright green color.
pixels4.show();
//if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}

}}
delay(100);
}
if(c=='2'){
Serial.println("2");
//digitalWrite(wa,LOW);
delay(100);

```

```

//digitalWrite(wa,HIGH);
if(m3>=9){m3=m3-9;}
if(m3>=m1)m2=m3/3;
  else m2=m1/3;
  m1=m1+9;
  int min1=min(m1,m2);
  int min2=min(m2,m3);
  int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels.setPixelColor(i, pixels.Color(j*15,0,0)); // Moder-
ately bright gre en color.
  pixels.show(); }
for(int j=85;j*3>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){
  int v1= max(j*3,m1);
  int v2=max(0,m2);
  int v3=max(0,m3);
  // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
  pixels.setPixelColor(i, pixels.Color(v1,v2,v3));
  r1=m1;r2=m2;r3=m3;// Moderately bright green color.
  pixels.show();
  pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*0.6,r2
*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright green
color.
  pixels3.show();
  pixels4.setPixelColor(i, pixels4.Color(r1*0.4+r4*0
.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
  pixels4.show();
  //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}
  }}}

delay(100);

}

```

```

// blank line to separate data from the two ports:
Serial.println();

// Now listen on the second port
portTwo.listen();
// while there is data coming in, read it
// and send to the hardware serial port:
Serial.println("Data from port two:");
while (portTwo.available() > 0) {
  char c = portTwo.read();
  Serial.write(c);
  if(c=='1'){
Serial.println("1");
//digitalWrite(wa,LOW);
delay(100);
//digitalWrite(wa,HIGH);
int j=0;
if(m1>=9){m1=m1-9;}
if(m3>=m1)m2=m3/3;
else m2=m1/3;
m3=m3+9;
int min1=min(m1,m2);
int min2=min(m2,m3);
int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels.setPixelColor(i, pixels.Color(0,j*5,j*15)); // Moder-
ately bright gre en color.
  pixels.show(); }
for(int j=85;j>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){

  // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
  int v1= max(0,m1);
  int v2=max(j,m2);
  int v3=max(j,m3);
  pixels.setPixelColor(i, pixels.Color(v1,v2,v3)); // Moder-

```

```

ately bright green color.
    r1=m1;r2=m2;r3=m3;
    pixels.show();
    pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*
0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
    pixels3.show();
    pixels4.setPixelColor(i, pixels4.Color(r1*0.
4+r4*0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately
bright green color.
    pixels4.show();
    //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}

}}
delay(100);
}
if(c=='2'){
Serial.println("2");
//digitalWrite(wa,LOW);
delay(100);
//digitalWrite(wa,HIGH);
if(m3>=9){m3=m3-9;}
if(m3>=m1)m2=m3/3;
    else m2=m1/3;
    m1=m1+9;
    int min1=min(m1,m2);
    int min2=min(m2,m3);
    int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels.setPixelColor(i, pixels.Color(j*15,0,0)); // Moder-
ately bright green color.
    pixels.show(); }
for(int j=85;j*3>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){
    int v1= max(j*3,m1);
    int v2=max(0,m2);
    int v3=max(0,m3);

```

```

    // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
    pixels.setPixelColor(i, pixels.Color(v1,v2,v3));
    r1=m1;r2=m2;r3=m3;// Moderately bright green color.
    pixels.show();
    pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*0.6,r2
*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright green
color.
    pixels3.show();
    pixels4.setPixelColor(i, pixels4.Color(r1*0.4+r4*0
.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
    pixels4.show();
    //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}
    }}}

delay(100);
}
}

// blank line to separate data from the two ports:
Serial.println();

// Now listen on the second port
portThree.listen();
// while there is data coming in, read it
// and send to the hardware serial port:
Serial.println("Data from port Three:");
while (portThree.available() > 0) {
    char c = portThree.read();
    Serial.write(c);
    if(c=='3'){
Serial.println("3");
//digitalWrite(wa,LOW);
delay(100);
//digitalWrite(wa,HIGH);
int j=0;

```

```

if(m1>=9){m1=m1-9;}
  if(m3>=m1)m2=m3/3;
  else m2=m1/3;
  m3=m3+9;
  int min1=min(m1,m2);
  int min2=min(m2,m3);
  int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels2.setPixelColor(i, pixels.Color(0,j*5,j*15)); // Mod-
erately bright gre en color.
  pixels2.show(); }
for(int j=85;j*3>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){

  // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
  int v1= max(0,m1);
  int v2=max(j,m2);
  int v3=max(j,m3);
  pixels2.setPixelColor(i, pixels.Color(v1,v2,v3)); // Mod-
erately bright gre en color.
  r1=m1;r2=m2;r3=m3;
  pixels2.show();
  pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*
0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
  pixels3.show();
  pixels4.setPixelColor(i, pixels4.Color(r1*0.
4+r4*0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately
bright green color.
  pixels4.show();
  //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}

}}
delay(100);
}
if(c=='4'){

```

```

Serial.println("4");
//digitalWrite(wa,LOW);
delay(100);
//digitalWrite(wa,HIGH);
if(m3>=9){m3=m3-9;}
if(m3>=m1)m2=m3/3;
  else m2=m1/3;
  m1=m1+9;
  int min1=min(m1,m2);
  int min2=min(m2,m3);
  int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels2.setPixelColor(i, pixels.Color(j*15,0,0)); // Moder-
ately bright gre en color.
  pixels2.show(); }
for(int j=85;j*3>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){
  int v1= max(j,m1);
  int v2=max(0,m2);
  int v3=max(0,m3);
  // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
  pixels2.setPixelColor(i, pixels2.Color(v1,v2,v3));
  r1=m1;r2=m2;r3=m3;// Moderately bright green color.
  pixels.show();
  pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*0.6,r2
*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright green
color.
  pixels3.show();
  pixels4.setPixelColor(i, pixels4.Color(r1*0.4+r4*0
.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
  pixels4.show();
  //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}

}

delay(100);

```

```

}}

}

// blank line to separate data from the two ports:
Serial.println();

// Now listen on the second port
portFour.listen();
// while there is data coming in, read it
// and send to the hardware serial port:
Serial.println("Data from port four:");
while (portFour.available() > 0) {
  char c = portFour.read();
  Serial.write(c);
  if(c=='3'){
Serial.println("3");
//digitalWrite(wa,LOW);
delay(100);
//digitalWrite(wa,HIGH);
int j=0;
if(m1>=9){m1=m1-9;}
  if(m3>=m1)m2=m3/3;
  else m2=m1/3;
  m3=m3+9;
  int min1=min(m1,m2);
  int min2=min(m2,m3);
  int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels2.setPixelColor(i, pixels.Color(0,j*5,j*15)); // Mod-
erately bright gre en color.
  pixels2.show(); }
for(int j=85;j*3>=min3;j=j-10){for(int
i=0;i<NUMPIXELS;i++){

  // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255

```

```

int v1= max(0,m1);
int v2=max(j,m2);
int v3=max(j,m3);
pixels2.setPixelColor(i, pixels.Color(v1,v2,v3)); // Mod-
erately bright gre en color.
  r1=m1;r2=m2;r3=m3;
  pixels2.show();
  pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*
0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
  pixels3.show();
  pixels4.setPixelColor(i, pixels4.Color(r1*0.
4+r4*0.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately
bright green color.
  pixels4.show();
  //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}

}}
delay(100);
}
if(c=='4'){
Serial.println("4");
//digitalWrite(wa,LOW);
delay(100);
//digitalWrite(wa,HIGH);
if(m3>=9){m3=m3-9;}
if(m3>=m1)m2=m3/3;
  else m2=m1/3;
  m1=m1+9;
  int min1=min(m1,m2);
  int min2=min(m2,m3);
  int min3=min(min1,min2);
for(int j=0;j<=17;j++)for(int i=0;i<NUMPIXELS;i++){
pixels2.setPixelColor(i, pixels.Color(j*15,0,0)); // Moder-
ately bright gre en color.
  pixels2.show(); }
for(int j=85;j*3>=min3;j=j-10){for(int

```

```
i=0;i<NUMPIXELS;i++){
  int v1= max(j,m1);
  int v2=max(0,m2);
  int v3=max(0,m3);
  // pixels.Color takes RGB values, from 0,0,0 up to
  255,255,255
  pixels2.setPixelColor(i, pixels2.Color(v1,v2,v3));
  r1=m1;r2=m2;r3=m3;// Moderately bright green color.
  pixels.show();
  pixels3.setPixelColor(i, pixels3.Color(r1*0.4+r4*0.6,r2
*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright green
color.
  pixels3.show();
  pixels4.setPixelColor(i, pixels4.Color(r1*0.4+r4*0
.6,r2*0.4+r5*0.6,r3*0.4+r6*0.6)); // Moderately bright
green color.
  pixels4.show();
  //if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}
  }

delay(100);

}}
}
Serial.println();

}
```

Appendix F: Code Senses

```
#include "Adafruit_NeoPixel.h"
#ifdef __AVR__
  #include <avr/power.h>
#endif

// Which pin on the Arduino is connected to the NeoPixels?
// On a Trinket or Gemma we suggest changing this to 1
#define PIN1      3
// #define PIN2    4
// #define PIN3    5
// #define PIN3    6
// How many NeoPixels are attached to the Arduino?
#define NUMPIXELS 50//29

// When we setup the NeoPixel library, we tell it how many
// pixels, and which pin to use to send signals.
// Note that for older NeoPixel strips you might need to
// change the third parameter--see the strandtest
// example for more information on possible values.
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUMPIXELS,
  PIN1, NEO_GRB + NEO_KHZ800);
//Adafruit_NeoPixel pixels2 = Adafruit_NeoPixel(NUMPIXELS,
  PIN2, NEO_GRB + NEO_KHZ800);
//Adafruit_NeoPixel pixels3 = Adafruit_NeoPixel(NUMPIXELS,
  PIN3, NEO_GRB + NEO_KHZ800);
//Adafruit_NeoPixel pixels4 = Adafruit_NeoPixel(NUMPIXELS,
  PIN4, NEO_GRB + NEO_KHZ800);
int flag=1;
float wa = 6;
int wb = 7;

int wc = 10;
float r1=0;
float r2=0;
float r3=0;
float r4=0;
float r5=0;
float r6=0;

char c;
float m1=0;
float m2=0;
float m3=0;
float m4=0;
float m5=0;
float m6=0;
int initial=0;
/*--stroke variables--*/
int stroke1;
int stroke2;
int stroke3;
int stroke4;

int stroke1_on = 0;
int stroke2_on = 0;
int stroke3_on = 0;
int stroke4_on = 0;

int sensorOutput = 0; //output of the strokesensor

unsigned long timer;
```

```

int stateSensor = 0;

/*--push variables--*/

float brightness = 50;
float fadeAmount = 0.5;

int maxBrightness = 0;

int button1 = 0;
//int button2 = 0;
//int button3 = 0;

/*-----SETUP FUNCTION-----*/

void setup() {
  Serial.begin(9600);
  #if defined (__AVR_ATtiny85__)
  if (F_CPU == 16000000) clock_prescale_set(clock_
div_1);
  #endif
  pixels.begin();
  //pixels2.begin();
  //pixels3.begin();
  //pixels4.begin();
  pinMode(11, OUTPUT);
  pinMode(2, INPUT);
  // pinMode(3, INPUT);
  // pinMode(4, INPUT);
  pinMode(10, OUTPUT);
}

/*-----LOOP FUNCTION-----*/

void loop() {
  int c=0;
  digitalWrite(10, HIGH);

```

```

stroke1 = analogRead(A0);
stroke2 = analogRead(A1);
stroke3 = analogRead(A2);
stroke4 = analogRead(A3);

button1 = digitalRead(2);
// button2 = digitalRead(3);
// button3 = digitalRead(4);

/* Serial.print(stroke1);
Serial.print("\t");
Serial.print(stroke2);
Serial.print("\t");
Serial.print(stroke3);
Serial.print("\t");
Serial.println(stroke4); */

checkSensor();

/* Serial.print("\t");
Serial.print(stroke1_on);
Serial.print(stroke2_on);
Serial.print(stroke3_on);
Serial.println(stroke4_on); */

if (stroke1_on == 1 || stroke2_on == 1 || stroke3_on
== 1 || stroke4_on == 1){

  timer = millis();
  sensorOutput = sensorOutput + 1;
  c=1;
}
/*
if (stroke1_on == 1 || stroke2_on == 1 || stroke3_on
== 1 || stroke4_on == 1 ||button1 == HIGH){

return;
}

```

```

*/

/*-----SENSOROUTPUT to STATE-----*/
if (sensorOutput > 0 && sensorOutput <= 5){
  stateSensor = 1;
}

if (sensorOutput > 5 && sensorOutput <= 10){
  stateSensor = 2; //2 times
}

if (sensorOutput > 10 && sensorOutput <= 15){
  stateSensor = 3; //3 times
}

if (sensorOutput > 15){
  stateSensor = 4; //4 times
}

if (millis() - timer >= 1000){ //inside or outside this
if-clause?
  resetSensor(); //necessary?
  sensorOutput = 0;
}

// Serial.println(sensorOutput);

/*--pushbutton and vibrationmotor part--*/
if (button1 == HIGH){
  stateSensor = 5;
  fadeAmount = 10;
  maxBrightness = 200;
  fade(11);
  c=2;
}

if (button1 == LOW){

```

```

analogWrite(11, 0);

if(sensorOutput == 0){
  stateSensor = 0;
}
}

Serial.println(stateSensor);
//delay(200); //better without?
/*-----nerve-----*/
if(c==1){
//Serial.println("Deffensive");
//digitalWrite(wa,LOW);
//delay(100);
//digitalWrite(wa,HIGH);
int j=0;
if(m1>=9){m1=m1-9;}
if(m3>=m1)m2=m3/3;
else m2=m1/3;
m3=m3+9;
for(int i=0;i<NUMPIXELS;i++){

// pixels.Color takes RGB values, from 0,0,0 up to
255,255,255

pixels.setPixelColor(i, pixels.Color(0,0,0)); // Moder-
ately bright green color.
pixels.show(); }
for(int p1=0; p1<=NUMPIXELS; p1=p1+10){
int p2=p1+3;
for(int j=0;j<50;j++){
for(int p=p1; p< p2;p++){

// pixels.Color takes RGB values, from 0,0,0 up to
255,255,255
pixels.setPixelColor(p, pixels.Color(0,j/3,j)); // Moder-
ately bright green color.

```

```

    pixels.show();
  }}

  for(int j=50;j>=0;j--){
    for(int p=p1; p<=p2;p++){

      // pixels.Color takes RGB values, from 0,0,0 up to
      255,255,255
      pixels.setPixelColor(p, pixels.Color(0,j/3,j)); // Moder-
      ately bright green color.
      pixels.show();
      if(j<=10){for(int i=p2;i<=p2+7;i++){ pixels.
      setPixelColor(i, pixels.Color(0,(10-j)/3,(10-j))); // Moder-
      ately bright green color.
      pixels.show();
      }}}

  }
  }
  for(int j=10;j>=0;j--){
    for(int p=p1; p<=p2;p++){

      // pixels.Color takes RGB values, from 0,0,0 up to
      255,255,255

      for(int i=p2;i<=p2+7;i++){ pixels.setPixelColor(i, pix-
      els.Color(0,j/3,j)); // Moderately bright green color.
      pixels.show();
      }

  }
  }

  }
  Serial.write('1');
  // delay(200);
  flag=1;
  }

```

```

    else if(c==2){
  // Serial.println("Assertive");
  if(m1>=9){m1=m1-9;}
  if(m3>=m1)m2=m3/3;
  else m2=m1/3;
  m3=m3+9;
  for(int i=0;i<NUMPIXELS;i++){

    // pixels.Color takes RGB values, from 0,0,0 up to
    255,255,255

    pixels.setPixelColor(i, pixels.Color(0,0,0)); // Moder-
    ately bright green color.
    pixels.show(); }

    for(int i=0;i<NUMPIXELS+10;i++){

      // pixels.Color takes RGB values, from 0,0,0 up to
      255,255,255

      pixels.setPixelColor(i-11, pixels.Color(0,0,0)); // Moder-
      ately bright green color.
      pixels.show();
      int jj=0;
      for( int ii=i-10;ii<=i-2;ii++){

        pixels.setPixelColor(ii, pixels.Color(jj*3,0,0)); // Mod-
        erately bright green color.
        pixels.show();
        jj=jj+5;}
      jj=50;
      for(int ii=i-2;ii<=i;ii++){

        pixels.setPixelColor(i, pixels.Color(jj*3,0,0)); // Mod-
        erately bright green color.
        pixels.show();
        jj=jj-5;}
      delay(10);
    }
  }

```

```

    if( Serial.read(>0){c=Serial.read(); Serial.
println(c);return;}}
    Serial.write('2');
    flag=1;
    }
else if(c==0){

    int a0 = random()%NUMPIXELS;
    int t0 = random()%2000;
    int t1 = random()%1000;
    if(flag==1){for(int j=0;j<=5;j++){for(int
i=0;i<NUMPIXELS;i++){

    // pixels.Color takes RGB values, from 0,0,0 up to
255,255,255

    pixels.setPixelColor(i, pixels.Color(j,j,j)); // Moderately
bright green color.
    pixels.show();
    stroke1 = analogRead(A0);
    stroke2 = analogRead(A1);
    stroke3 = analogRead(A2);
    stroke4 = analogRead(A3);
    checkSensor();
    button1 = digitalRead(2);
    if (stroke1_on == 1 || stroke2_on == 1 || stroke3_on
== 1 || stroke4_on == 1 ||button1 == HIGH){

return;
}}
    flag=-1;}

    for(int jj=5; jj<=50;jj++){
    for(int ii=a0-2;ii<=a0+2;ii++){
        pixels.setPixelColor(ii, pixels.Color(jj,jj,jj)); // Mod-
erately bright green color.
        pixels.show();
        delay(3);

```

```

        stroke1 = analogRead(A0);
        stroke2 = analogRead(A1);
        stroke3 = analogRead(A2);
        stroke4 = analogRead(A3);
        checkSensor();
        button1 = digitalRead(2);
        if (stroke1_on == 1 || stroke2_on == 1 || stroke3_on
== 1 || stroke4_on == 1 ||button1 == HIGH){

return;
    }
    }
    }
    for(int i=0;i<=t0;i++){
        stroke1 = analogRead(A0);
        stroke2 = analogRead(A1);
        stroke3 = analogRead(A2);
        stroke4 = analogRead(A3);
        checkSensor();
        button1 = digitalRead(2);if (stroke1_on == 1 ||
stroke2_on == 1 || stroke3_on == 1 || stroke4_on == 1
||button1 == HIGH){

return;
    }}
    for(int jj=50; jj>=5;jj--){
        for(int ii=a0-2;ii<=a0+2;ii++){
            pixels.setPixelColor(ii, pixels.Color(jj,jj,jj)); // Mod-
erately bright green color.
            pixels.show();
            delay(3);
            stroke1 = analogRead(A0);
            stroke2 = analogRead(A1);
            stroke3 = analogRead(A2);
            stroke4 = analogRead(A3);
            checkSensor();
            button1 = digitalRead(2);
            if (stroke1_on == 1 || stroke2_on == 1 ||

```

```

stroke3_on == 1 || stroke4_on == 1 || button1 == HIGH)
{

return;
}
}
}
for(int i=0;i<=t1;i++){
    stroke1 = analogRead(A0);
    stroke2 = analogRead(A1);
    stroke3 = analogRead(A2);
    stroke4 = analogRead(A3);
    checkSensor();
    button1 = digitalRead(2);
    if (stroke1_on == 1 || stroke2_on == 1 ||
stroke3_on == 1 || stroke4_on == 1 || button1 == HIGH)
{

return;
}
}

}

/*-----FUNCTIONS-----*/

void checkSensor(){ //check the strokesensor
    if (stroke1 < 900){
        stroke1_on = 1;
    }
    else {
        stroke1_on = 0;
    }

    if (stroke2 < 900){
        stroke2_on = 1;
    }
    else {

```

```

        stroke2_on = 0;
    }

    if (stroke3 < 900){
        stroke3_on = 1;
    }
    else {
        stroke3_on = 0;
    }

    if (stroke4 < 900){
        stroke4_on = 1;
    }
    else {
        stroke4_on = 0;
    }

}

void resetSensor(){ //reset the strokesensor
    stroke1_on = 0;
    stroke2_on = 0;
    stroke3_on = 0;
    stroke4_on = 0;
}

void fade (int motor){ //fade the vibration motor

    analogWrite(motor, brightness);

    brightness = brightness + fadeAmount;

    if (brightness == 50 || brightness == maxBrightness){
        fadeAmount = -fadeAmount;
    }

}

```