



Geïntegreerde proef

Cocktailmaker

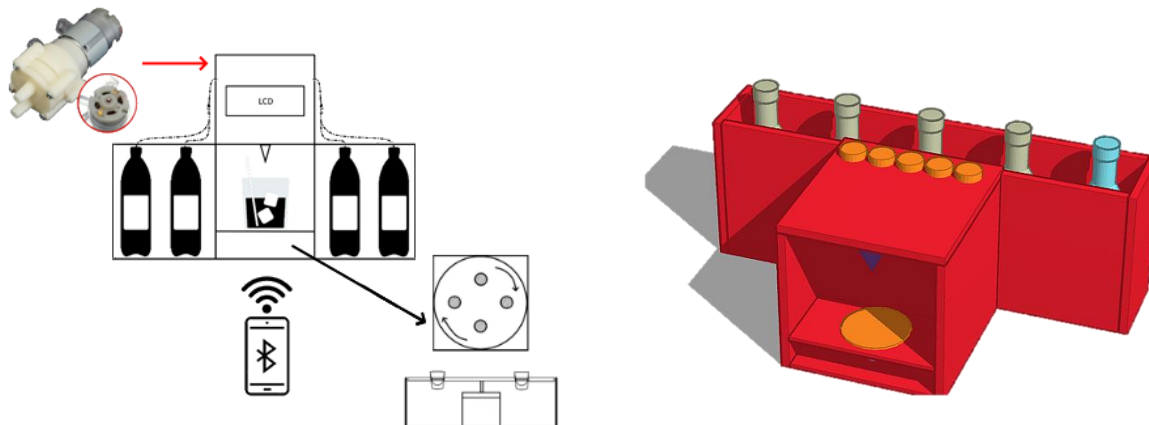
Leerling: Lukas Olivier 6EE
Begeleider: Hertens Karel, VTI
Schooljaar: 2020 - 2021
Prizma Campus VTI

Abstractum:

Elke leerling in het 6^{de} jaar Elektriciteit-Elektronica in het Campus VTI krijgt de opdracht om een eindproject te creëren waarbij de geziene leerstof wordt omgezet in de praktijk. De eerste stap was het brainstormen rond een idee. Dit gebeurde al in de zomervakantie. In september stelde iedereen zijn ideeën voor en begon het denkwerk rond de concrete uitvoering van deze project. Ik wou iets maken dat ik ook nadien nog zou kunnen gebruiken. Ik koos ervoor om een automatische cocktailmaker te maken, met behulp van Arduino.

Voorwoord:

Nadat het idee van een automatische cocktailmaker vaststond begon ik met een eerste schets en aansluitschema. Daarna startte ik met het schrijven van het Arduino programma. Ik ontwierp een app om de cocktails te kunnen bestellen en linkte die met een database. In 2021 begon ik aan het samenstellen van de kast waar alle hardware en flessen in verwerkt zitten. De overige tijd gebruikte ik om alle bugs uit het programma / app te halen en de beste recepten voor cocktails uit te testen. Ik vond het een zeer interessant en leuk project om aan te werken en blijf tot op vandaag ten volle achter mijn keuze staan. Ik heb veel kennis opgedaan over het communiceren met bluetooth, opstellen van dataservers, apps ontwerpen en 3D tekenen in Solid Edge.



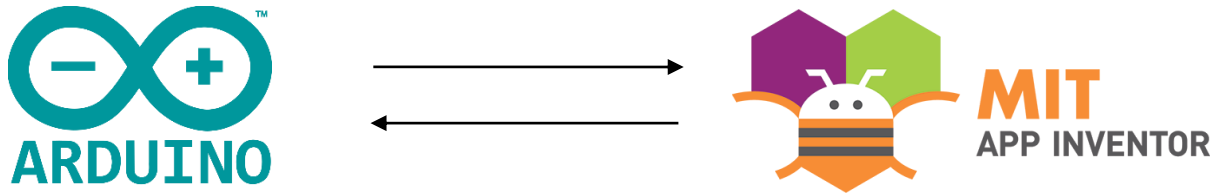
Eerste schetsen over de werking & ontwerp

Inhoud

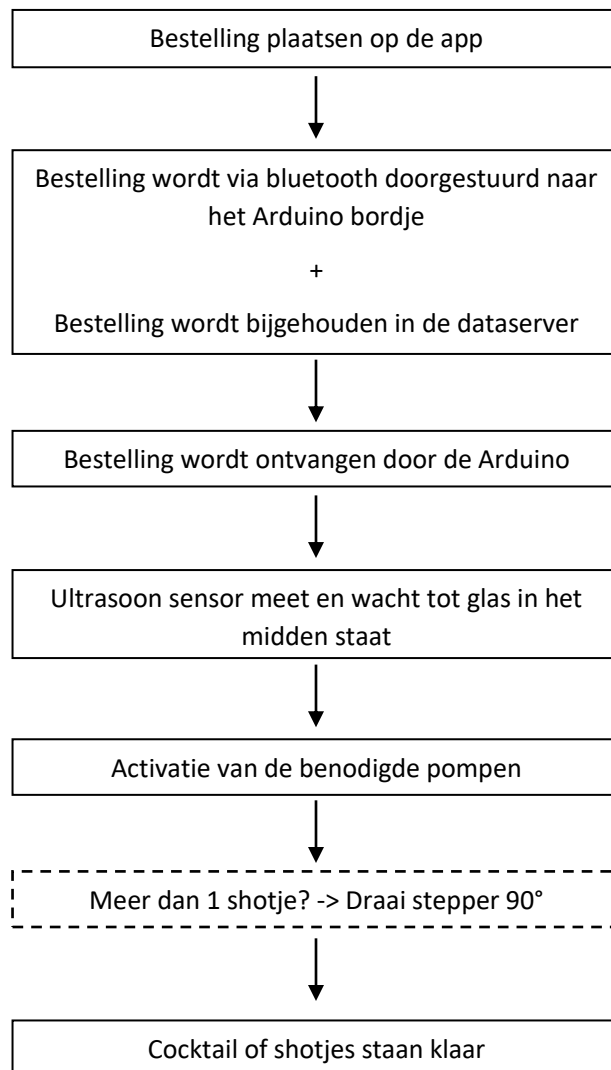
Abstractum:.....	3
Voorwoord:	3
1.0 Software:.....	5
1.1 Blockschema	5
1.2 Arduino Programma	6
1.3 MIT app.....	12
1.3.1 Login scherm	13
1.3.2 Registratie scherm.....	14
1.3.3 Bestelscherm	15
1.3.4 Customcocktailscherm	16
1.3.5 Adminscherm	17
2.0 Hardware:	18
2.1 Nema17 Stappenmotor	18
2.2 L298N DC Motor Driver	19
2.3 Ultrasoon Sensor	20
2.4 Relaisbord	21
2.5 DC pomp	22
2.6 LCD-Display + I ² C.....	23
2.7 Bedradingsschema	25
3.0 Externe onderdelen:	26
3.1 Houten kast	26
3.2 Slangen	27
3.3 Draaitafel	27
3.4 3D-Shields	28
Besluit:.....	29
Dankwoord:	29

1.0 Software:

Het programma werd geschreven in C++ met behulp van Arduino. C++ is een oude maar bruikbare programmeertaal waarin veel mogelijk is. De app werd ontworpen via MIT App Inventor, een browser gebaseerde app maker die werkt met “blokken”.



1.1 Blockschema



1.2 Arduino Programma

```
// Stepper
#include <Stepper.h>
const int stepsPerRevolution = 200; //360/1,8
int pos = 0;
Stepper myStepper(stepsPerRevolution, 5, 6, 9, 10);
boolean moveClockwise = true;

// Bluetooth
#include <SoftwareSerial.h>
SoftwareSerial EEBlue(0, 1); // RX | TX

// LCD
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x38, 16, 2);
```

Invoegen van bibliotheken

```
// Pompen
int Hbrug = 19;
int Vodka = 18;
int Cranberry = 20;
int Sprite = 8;
int Gin = 21 ;
```

Poorten toekennen

```
//ultrasoon sensor
int pingPin = A5;
long duration, cm;
long afstand;
int newvalue;
int oldvalue;
int naast;
int midden;
```

Variabelen toekennen
voor de ultrasoon sensor

```
void setup()
{
    //begin seriële (bluetooth transmissie)
    Serial.begin(9600);
    Serial1.begin(9600);

    //opstart tijd voor brainbox arduino leonardo
    delay(1750);

    // Gebruik poorten 19-23 als OUTPUT
    pinMode(18, OUTPUT);
    pinMode(19, OUTPUT);
    pinMode(20, OUTPUT);
    pinMode(21, OUTPUT);
    pinMode(8, OUTPUT);

    // Relaisbord is laag actief
    digitalWrite(18, HIGH);
    digitalWrite(19, HIGH);
    digitalWrite(20, HIGH);
    digitalWrite(21, HIGH);
    digitalWrite(8, HIGH );
```

De poorten die gekoppeld
zijn met het relaisbord
instellen als uitgang en
deze uitgangen hoog
zetten. (Het relaisbord is
actief laag).

```
// Setup voor LCD
lcd.init();
lcd.backlight();
lcd.setCursor(0, 0);

// Setup voor Servo
myStepper.setSpeed(15);
}
```

LCD activeren
draaisnelheid van servo
instellen op 15

```
void loop()
{
  cocktail = 0;
  if (Serial1.available() > 0 )
  {
    cocktail = Serial1.read();
    lcd.clear();
  }

  else
  {
    lcd.setCursor(1, 0);
    lcd.print("Maak een keuze");
    lcd.setCursor(2, 1);
    lcd.print("via de app!");
  }
}
```

Als de Arduino een
bluetooth signaal
binnenkrijgt dan wordt de
data (een letter) aan de
variabele "cocktail"
toegekend.

Als er geen data
binnenkomt dan verschijnt
er op de LCD dat er
gewacht wordt op een
keuze.

```
// Cocktails:
switch (cocktail) {

  //Vodka sprite
  case 'a':
    Ultrasoon(10);
    VodkaPomp(1000);
    GinPomp(0);
    CranberryPomp(0);
    SpritePomp(3000);
    einde();
    break;
```

Iedere letter komt
overeen met een
bepaalde cocktail. De
cocktail wordt
samengesteld door
bepaalde pompen een
bepaalde tijd te activeren.

```
// Gin Shot
case 'f':
  shot = true;
  VodkaPomp(0);
  GinPomp(1000);
  CranberryPomp(0);
  SpritePomp(0);
  break;
```

Bij een shot wordt een
pomp 1 seconde
geactiveerd.

```
// Custom Cocktail
case 'g':
    Ultrasoon(10);
    custom();
    einde();
    break;

// Spoelen
case 'n':
    VodkaPomp(5000);
    GinPomp(5000);
    CranberryPomp(5000);
    SpritePomp(5000);
    break;
```

Bij de custom cocktail worden er geen pompen geactiveerd aangezien dit in een andere functie door de klant zelf bepaald wordt.

Om de pompen en slangen te spoelen worden alle pompen 5 seconden geactiveerd.

```
void Ultrasoon(int afstand) {
    while (cm != afstand) {
        newvalue = cm;
        if (newvalue != oldvalue) {
            oldvalue = newvalue;
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("Center het glas");
            naast = afstand - cm;
            midden = 7 + naast;
            lcd.setCursor(midden, 1);
            lcd.write(255);
            lcd.setCursor(midden + 1, 1);
            lcd.write(255);
        }
        pinMode(pingPin, OUTPUT);
        digitalWrite(pingPin, LOW);
        delayMicroseconds(2);
        digitalWrite(pingPin, HIGH);
        delayMicroseconds(5);
        digitalWrite(pingPin, LOW);
        pinMode(pingPin, INPUT);
        duration = pulseIn(pingPin, HIGH);
        cm = duration / 29 / 2;
        delay(100);
    }
    lcd.clear();
    cm = 0;
}
```

De ultrasoon sensor meet de afstand tussen zichzelf en het glas. De LCD toont de positie van het glas aan. Zolang het glas niet in het midden staat zal de ultrasoon blijven meten. Eenmaal het glas in het midden staat kan er gepompt worden.




```
void CranberryPomp(int  
  InschakeltijdCranberry)  
{  
  if (InschakeltijdCranberry > 0)  
  {  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("  Cranberry");  
    lcd.setCursor(0, 1);  
    digitalWrite(Cranberry, LOW);  
  
    for (int i = 0; i <= 16; i++)  
    {  
      lcd.write(255);  
      delay(InschakeltijdCranberry / 16);  
    }  
  }  
}
```

Het programma doorloopt elke pomp. We nemen hier de cranberry als voorbeeld.

Bij de digitalWrite wordt het relaisbord aangestuurd waardoor de pomp activeert voor een bepaalde inschakeltijd die op pag7 werd toegekend.

Er wordt een laadbalk weergegeven op de LCD terwijl er gepompt wordt.

De totale tijd wordt verdeeld over de 16 segmenten van de LCD.



```
void VodkaPomp(int InschakeltijdVodka) {
  int getal;
  if (InschakeltijdVodka > 0) {
    getal = 1;
    if (shot == true)
    {
      while (Serial1.available() == 0)
      {
        lcd.setCursor(0, 0);
        lcd.print("Aantal?");
      }
      if (Serial1.available() > 0 )
      {
        getal = Serial1.parseInt();
        Serial.println(getal);
      }
    }
    for (int i = 0; i < getal; i++)
    {
      Ultrasoon(11);
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("      Vodka");
      lcd.setCursor(0, 1);
      digitalWrite(Vodka, LOW);
      for (int i = 0; i <= 16; i++)
      {
        lcd.write(255);
        delay(InschakeltijdVodka / 16);
      }
      digitalWrite(Vodka, HIGH);
      if (shot == true)
      {
        delay(500);
        digitalWrite(Hbrug, LOW);
        myStepper.step(50);
        delay(1000);
        digitalWrite(Hbrug, HIGH);
      }
    }
  }
  lcd.clear();
}
```

Wanneer een cocktail wordt gedefinieerd als shot (pag.7) dan ziet de functie er wat anders uit.

De Arduino wacht opnieuw op een bluetooth signaal in vorm van een getal. Dit getal bepaalt het aantal shots er worden gemaakt.

Wanneer het shotglas gevuld is wordt de servo geactiveerd en draait deze 50 stappen of 90°.

Het getal dat eerder werd ontvangen door de Arduino, bepaalt het aantal keer dat dit proces doorlopen wordt.

```

void custom()
{
    int drank;
    int lcdteller;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("    Max drank");
    while (drank != 'z')
    {
        if (Serial1.available() > 0 )
        {
            drank = Serial1.read();

            switch (drank) {
                case 'h':
                    digitalWrite(Vodka, LOW);
                    break;
                case 'i':
                    digitalWrite(Gin, LOW);
                    break;
                case 'j':
                    digitalWrite(Sprite, LOW);
                    break;
                case 'k':
                    digitalWrite(Cranberry, LOW);
                    break;
                case 'l':
                    if (lcdteller <= 16)
                    {
                        lcd.setCursor(lcdteller, 1);
                        lcd.write(255);
                        lcdteller++;
                    }
                    break;

                case 'm':
                    digitalWrite(Vodka, HIGH);
                    digitalWrite(Gin, HIGH);
                    digitalWrite(Sprite, HIGH);
                    digitalWrite(Cranberry, HIGH);
                    break;
            }
        }
        lcd.clear();
        drank = 0;
    }
}

```



Met deze modus kan je je eigen cocktail samenstellen door knoppen een bepaalde tijd in te houden.

Als een knop wordt ingedrukt dan verstuurd deze een bluetooth signaal die overeenkomt met een bepaalde drank.

De pomp blijft geactiveerd tot je de knop loslaat. Wanneer eender welke knop wordt losgelaten wordt de letter 'm' doorgestuurd waardoor alle pompen gedeactiveerd worden.

Om de zoveel tijd wordt ook de letter 'l' doorgestuurd. Wanneer dit gebeurt wordt er een blokje op de lcd gedisplayd. Zo kan je zien hoeveel drank je nog kan toevoegen. Dit elimineert de kans op het overlopen van het glas.

```
void einde()
{
  Serial1.write("klaar");
  lcd.clear();
  lcd.print("Drankje is klaar!");
  digitalWrite(22, HIGH);
  delay(1000);
  lcd.clear();
}
```

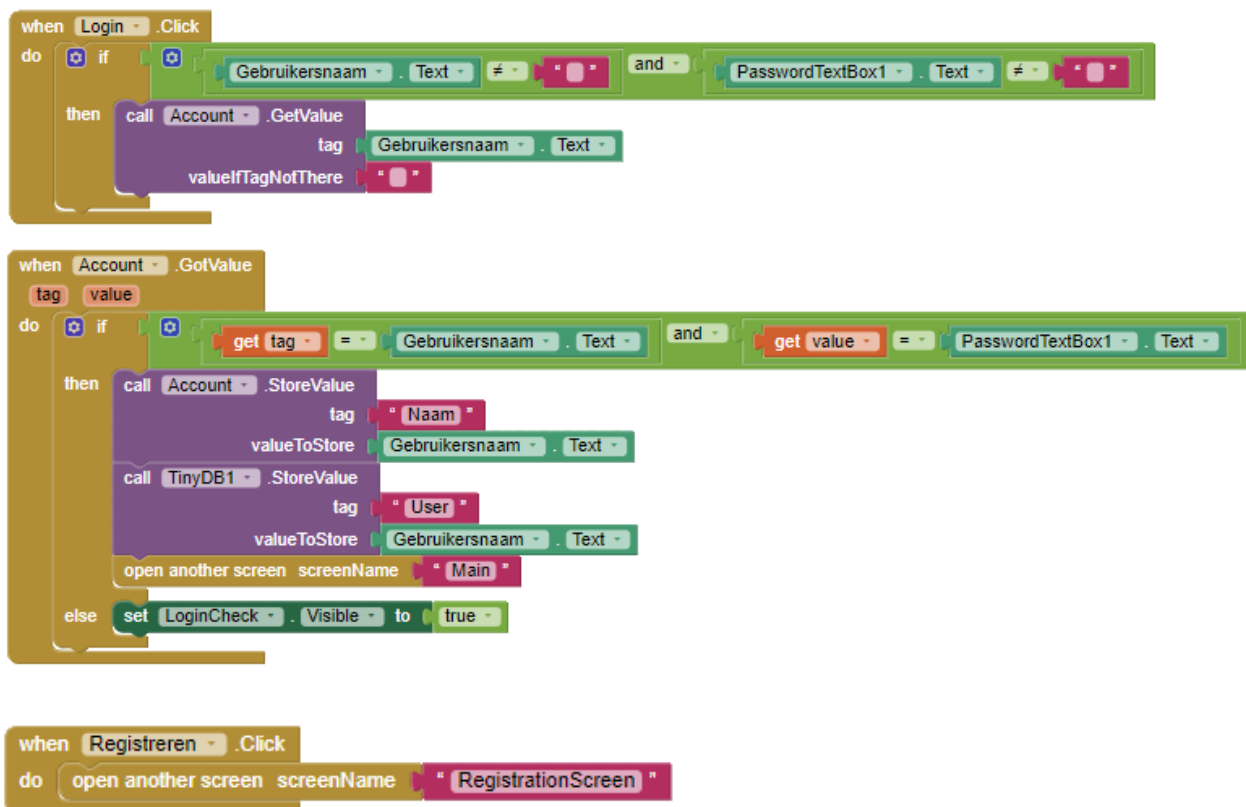
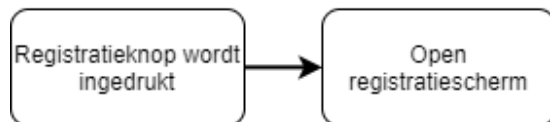
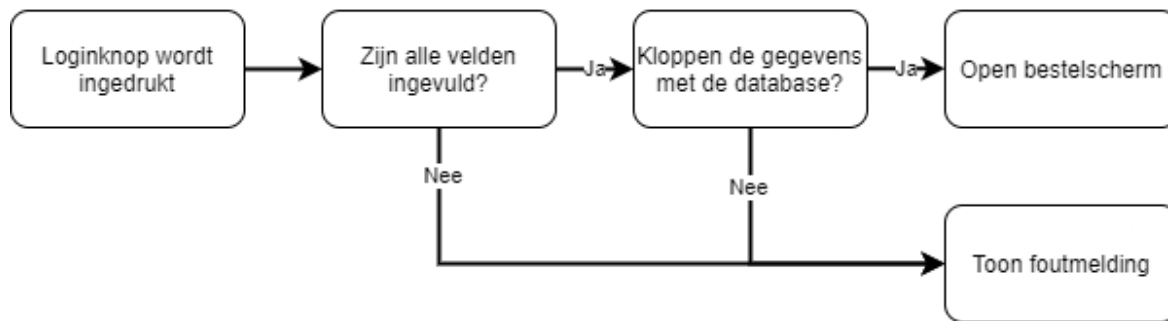
Als het volledige programma doorlopen is, wordt er aangegeven op de LCD dat de cocktail klaar is. Dit wordt ook doorgestuurd naar de app. Hierdoor kunnen de nieuwe bestellingen weer binnenkomen.

1.3 MIT app

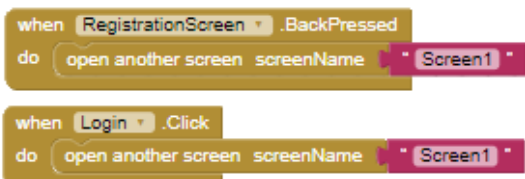
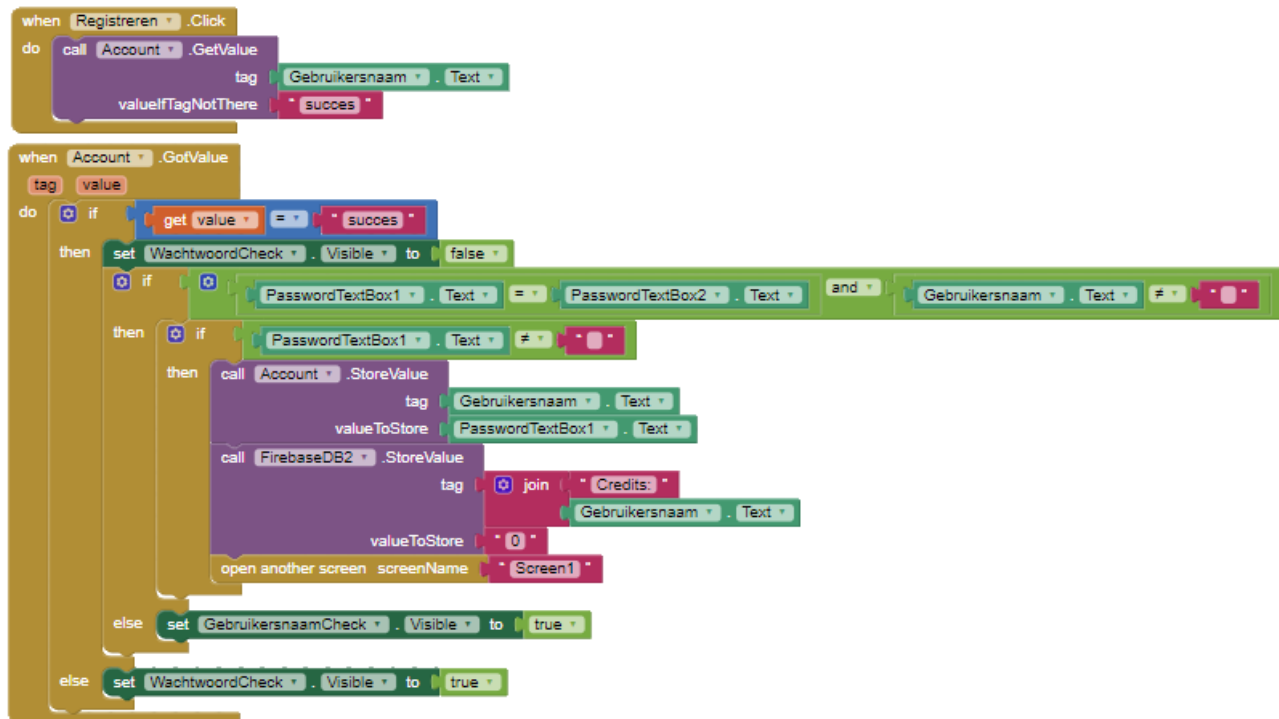
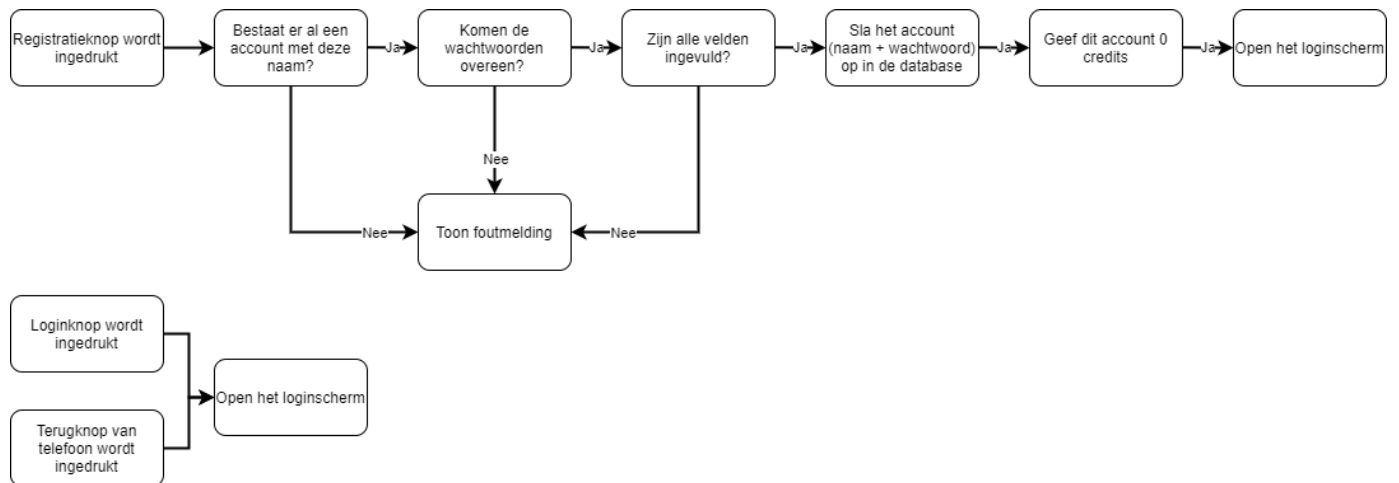
In de app kan je drank bestellen. Die bestellingen kosten “credits”. Die credits worden samen met een bijhorend account in een database bijgehouden. De app bestaat uit verschillende schermen.



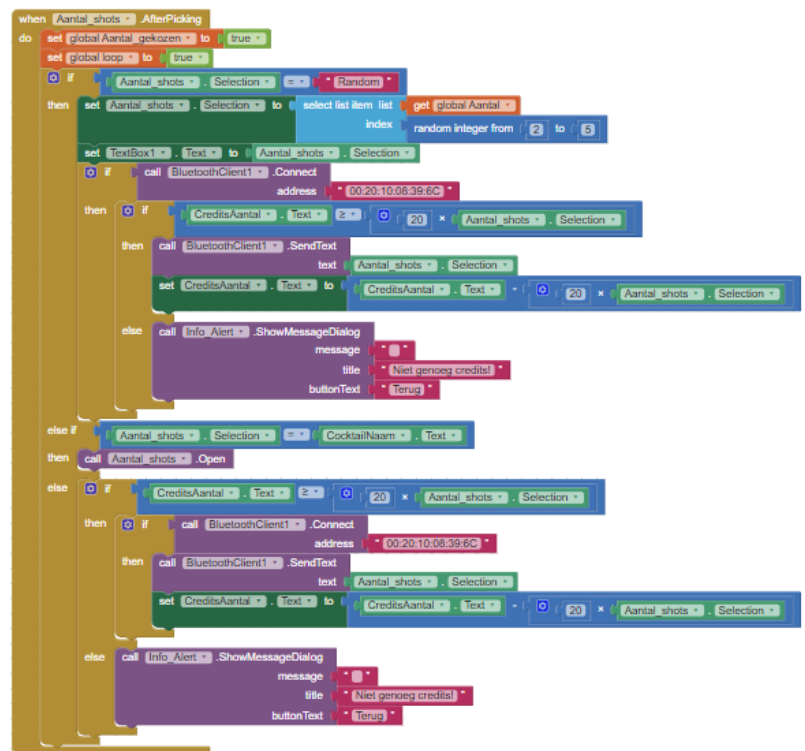
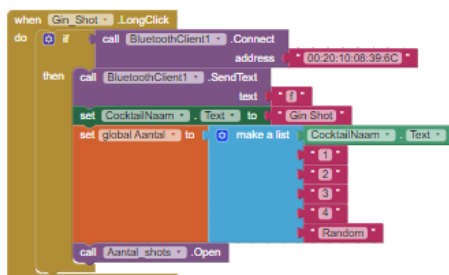
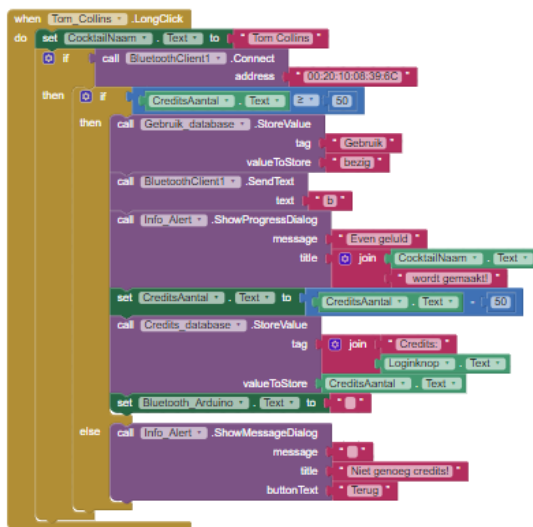
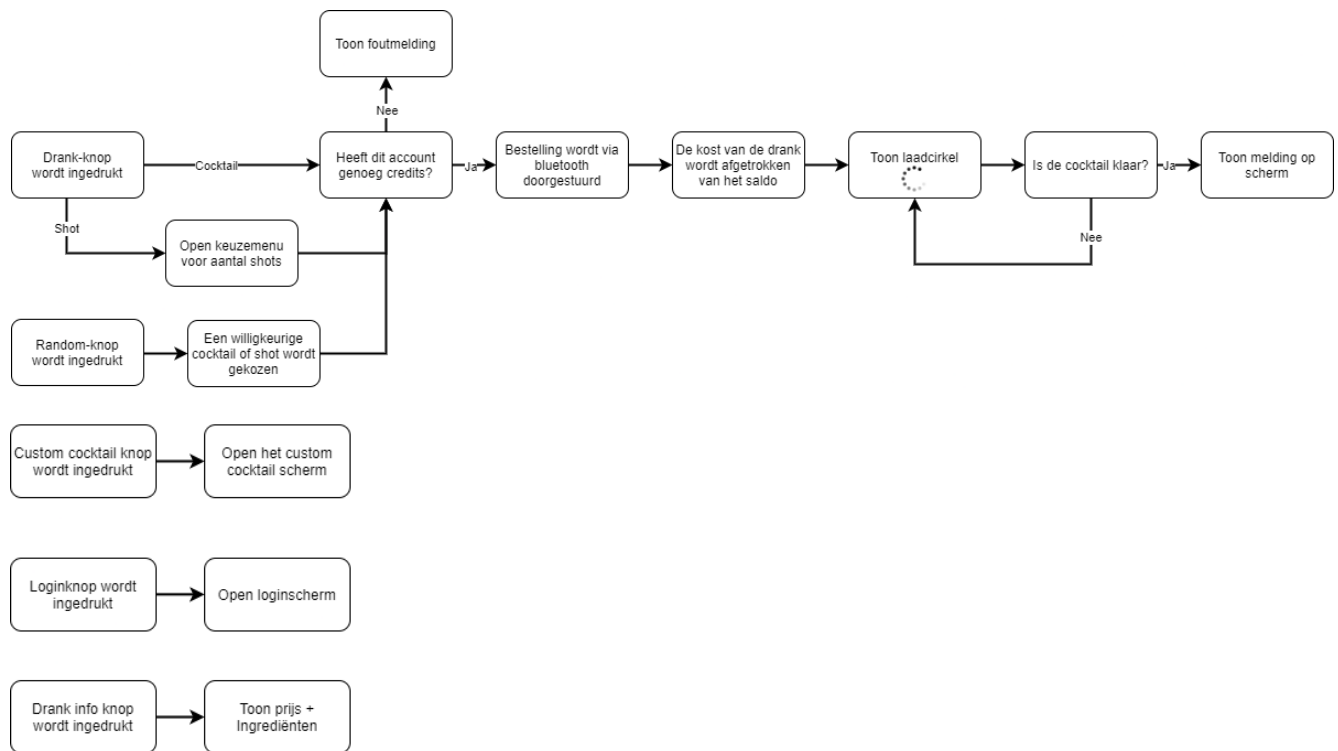
1.3.1 Login scherm



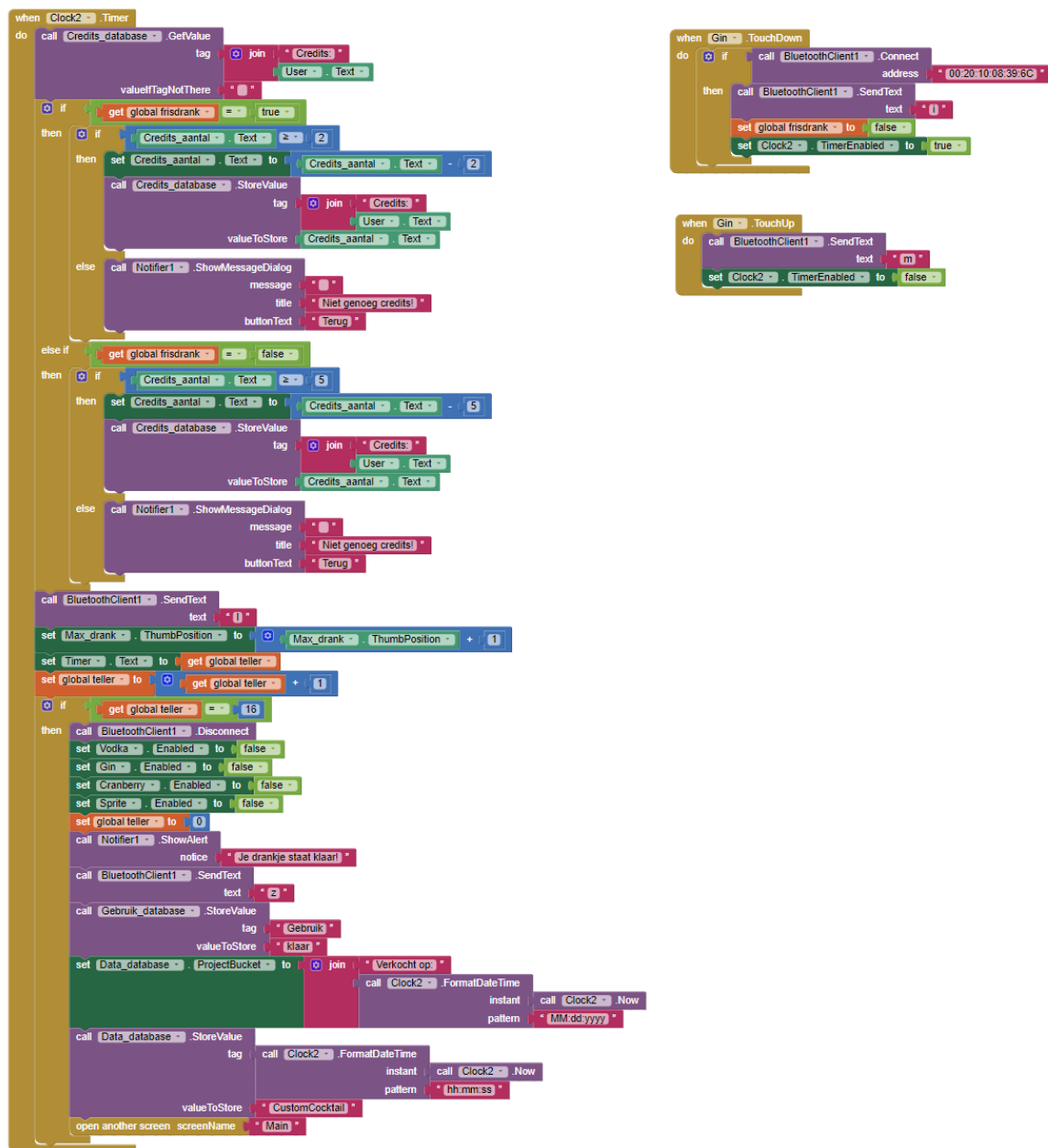
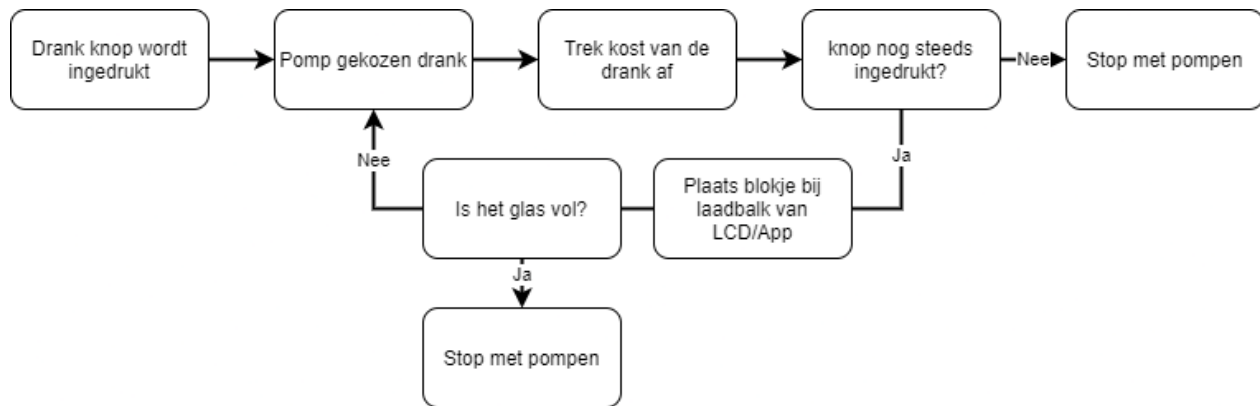
1.3.2 Registratie scherm



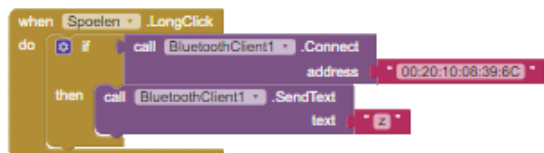
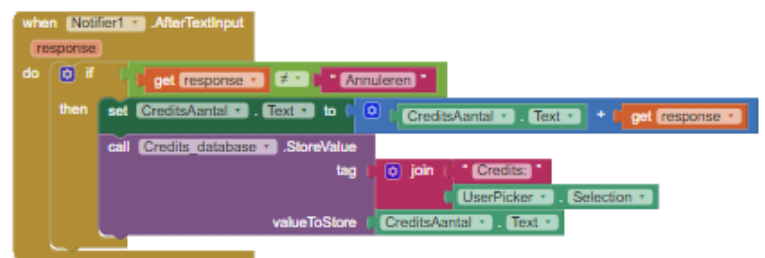
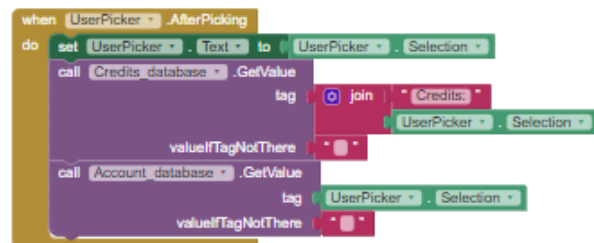
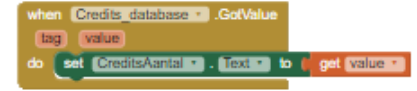
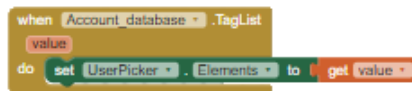
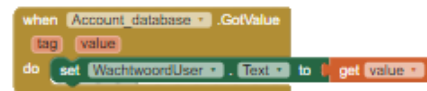
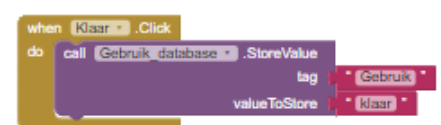
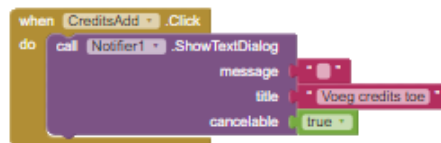
1.3.3 Bestelscherm



1.3.4 Customcocktailscherm



1.3.5 Adminscherm



2.0 Hardware:

2.1 Nema17 Stappenmotor

Inleiding:

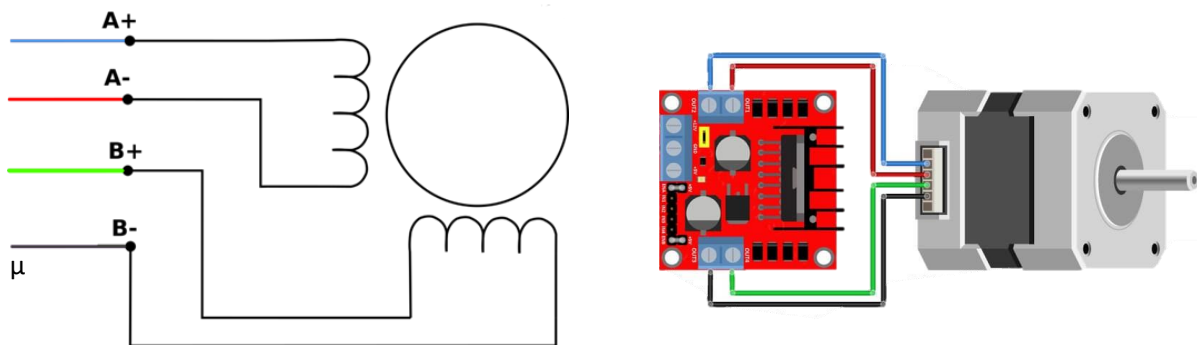
Om de draaiplaat te laten draaien maak ik gebruik van een stappenmotor. Aangezien er telkens een verdraaiing van 90° moet plaatsvinden om naar het volgende glas te gaan, is een stappenmotor hier zeer geschikt voor.

Omdat de draaiplaat is redelijk zwaar is, koos ik voor een Nema17. Een krachtige hybride stappenmotor, die zeer bekend is bij hobbyisten.



Aansluiting:

De Nema17 wordt aangesloten aan een (L298N) motordriver. Deze driver stuurt de spoelen van de stappenmotor aan. De driver moet wel nog gevoed worden en aangesloten worden aan een Microcontroller. Pas dan kunnen we de motor besturen.

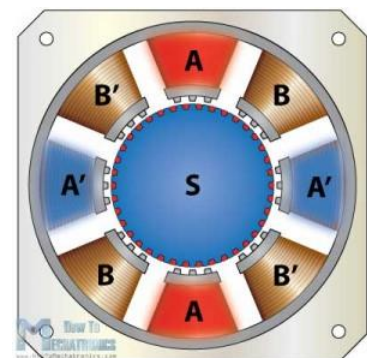


Werking:

Een hybride stappenmotor is een combinatie tussen de permanente magneet en variabele reluctance stappenmotoren. De rotor is magnetisch en getand, ook de polen van de stator zijn getand. De rotor zal zich bij het bekrachtigen van een fase verplaatsen naar de positie met de minste ruimte tussen de tanden van de rotor en de tanden van de stator.

Aantal voordelen van een hybride stappenmotor zijn:

- Groot koppel bij het starten en bij lage snelheden
- Weinig onderhoud
- Zeer kleine stapgroottes
- Weinig kans op slip



Programma:

```

#include <Stepper.h> // Stappenmotor bibliotheek

const int stepsPerRevolution = 200; // 360°/4,8° per stap
int Hbrug = 22;
Stepper myStepper(5, 6, 9, 10); // Stuur aan met pennen 5,6,9,10

void setup()
{
  pinMode(22, OUTPUT);
  digitalWrite(22, LOW);
  myStepper.setSpeed(30); // Zet draaisnelheid op 30 (rpm)
}

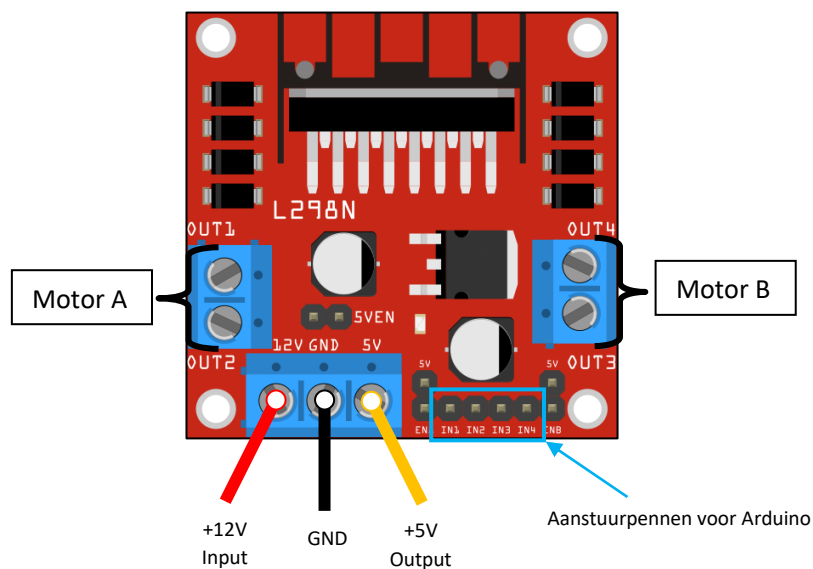
void loop()
{
  myStepper.step(50); // 4.8*50 = 90° wijzerzin
  delay(1000);
}

```

2.2 L298N DC Motor Driver

Inleiding:

Een van de gemakkelijkste en goedkoopste manieren om DC-motoren te besturen, is door de L298N Motor Driver te koppelen aan een microcontroller. Het kan zowel de snelheid als de draairichting van een DC- of stappenmotor regelen.

Aansluiting:

2.3 Ultrasoon Sensor

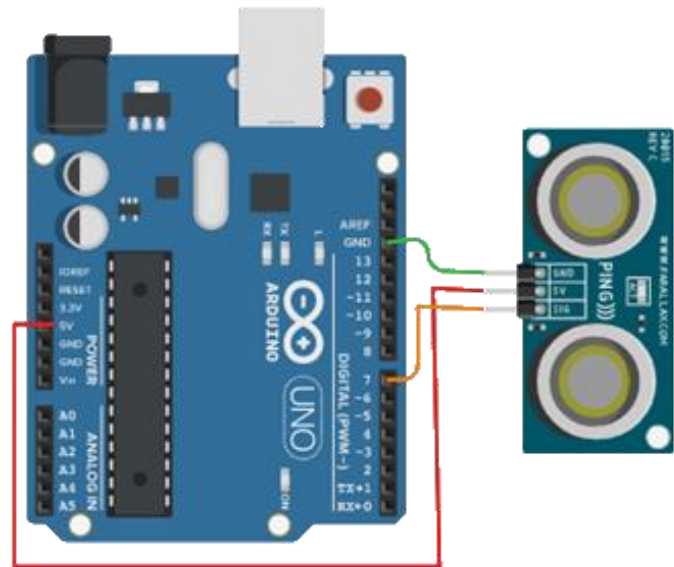
Inleiding:

Om te controleren of het glas precies onder de slangen staat, maak ik gebruik van een ultrasoon sensor. Het grote voordeel van deze sensor is dat de detectie gebeurt met geluidsgolven en niet met licht, wat met een glazen object niet goed zou werken.



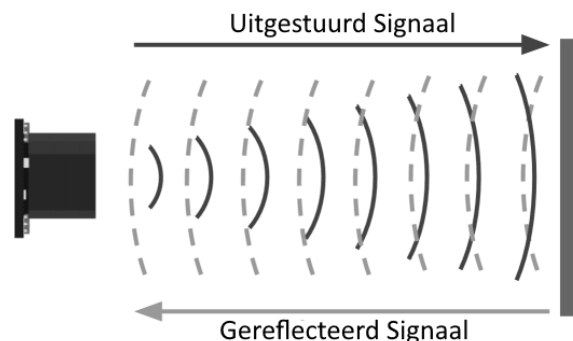
Aansluiting:

De sensor kan simpelweg worden aangesloten met een +5V, GND en een signaallijn.



Werking:

Ultrasone sensoren werken via het uitzenden van geluidsgolven van een frequentie die te hoog is voor mensen om te horen. Deze geluidsgolven reizen door de lucht met de geluidssnelheid van ongeveer 343 m / s. Als er zich een object voor de sensor bevindt, worden de geluidsgolven teruggekaatst en gedetecteerd door de ontvanger van de ultrasone sensor. Door te meten hoeveel tijd er verstreken is tussen het verzenden en ontvangen van de geluidsgolven, kan de afstand tussen de sensor en het object berekend worden.



Programma:

Dit klein voorbeeldprogramma geeft de gemeten afstand weer op de seriële monitor.

```
int pingPin = 7;
long duration, cm;
int afstand;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  pinMode(pingPin, OUTPUT);
  digitalWrite(pingPin, LOW);
  delayMicroseconds(2);
  digitalWrite(pingPin, HIGH);
  delayMicroseconds(5);
  digitalWrite(pingPin, LOW);
  pinMode(pingPin, INPUT);
  duration = pulseIn(pingPin, HIGH);
  cm = duration / 29 / 2;
  delay(100);
  Serial.println(cm);
}
```

2.4 Relaisbord

Inleiding:

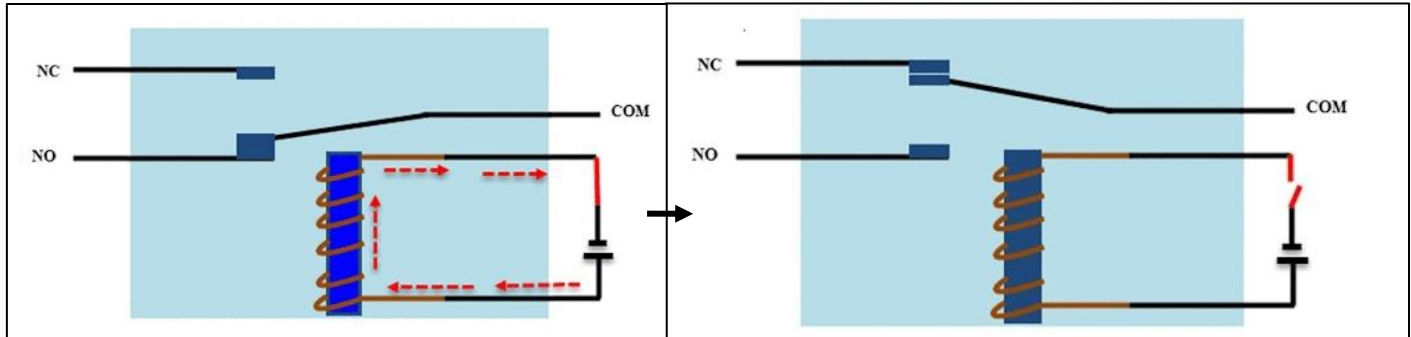
Om de pompen en de motor driver aan te sturen maak ik gebruik van een relaisbord. Als een relais geactiveerd wordt staat er spanning op de pomp, waardoor die de drank begint te pompen.

**Aansluiting:**

Het relaisbordje wordt gevoed met 5V. Elke relais op dit bordje heeft zijn eigen aanstuurpen. Die kan gekoppeld worden aan een digitale uitgang van een Arduino. Aan de bovenkant kan er bv. een pomp gekoppeld worden. Er is meestal een normaal open (NO) en een normaal gesloten (NG) contact. Mijn relaisbord ondersteunt spanningen tot 12V.

Werking:

Een relais werkt volgens het principe van elektromagnetische inductie. Wanneer er stroom door de elektromagneet vloeit, wekt hij een magnetisch veld eromheen op. Wanneer de spoel wordt bekrachtigd met een gelijkstroom, trekt deze het contact aan.



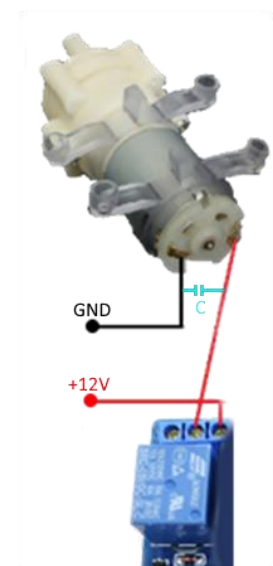
2.5 DC pomp

Inleiding:

Om de drank van de flessen naar het glas te transporteren maak ik gebruik van een DC-pomp die werkt op 12V. De pompen zijn 'food graded' dus ze kunnen gebruikt worden om drinkbare vloeistoffen te transporteren.

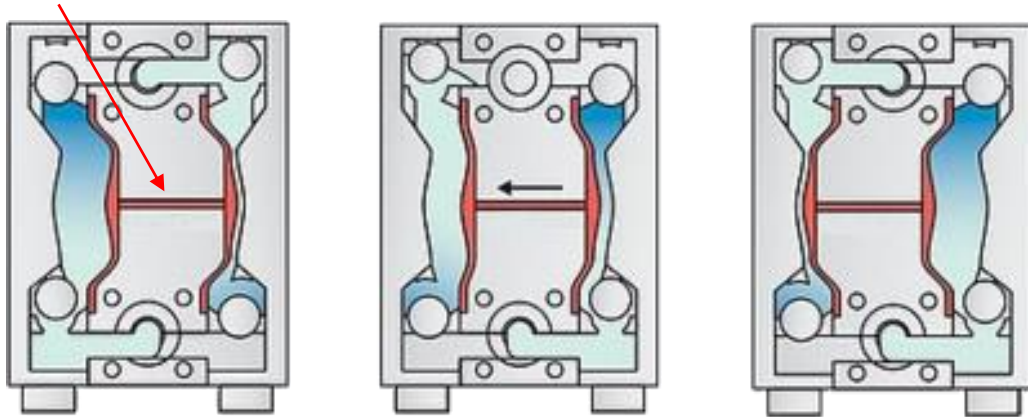
**Aansluiting:**

De pompen worden aangesloten aan het relaisbord die al eerder werd besproken. Tussen de pomp wordt er ook een condensator aangesloten van 0.1μF om de storing van de pompen tegen te werken. Dit is nodig omdat het bluetooth signaal van de bestellingen anders verstoord kan worden.



Werking:

De pompen die ik gebruik zijn dubbelmembraan pompen. Een dubbelmembraan pomp is een pomp die twee flexibele membranen gebruikt die heen en weer bewegen, waardoor een tijdelijke kamer ontstaat, die zowel vloeistof aanzuigt als afvoert door de pomp. De membranen werken als scheidingswand tussen lucht en vloeistof. In de afbeelding hieronder kun je de werking van dit membraan zien.



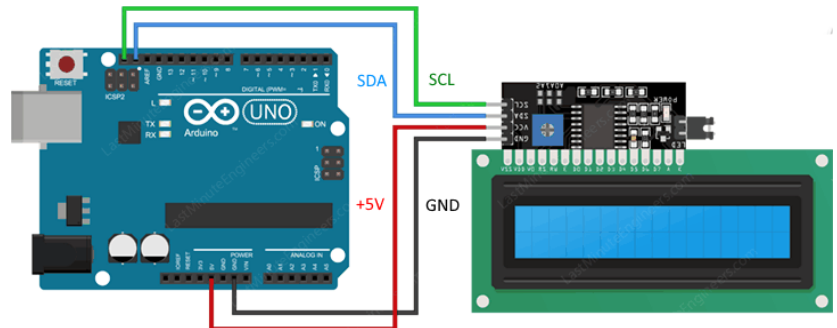
2.6 LCD-Display + I²C

Inleiding:

Alle informatie en instructies worden weergegeven op een LCD. Ik gebruik een 16x2 LCD, dat betekent dat er 2 rijen van 16 karakters zijn. Om het aantal aansluitingen sterk te verminderen maak ik gebruik van een I²C.

Aansluiting:

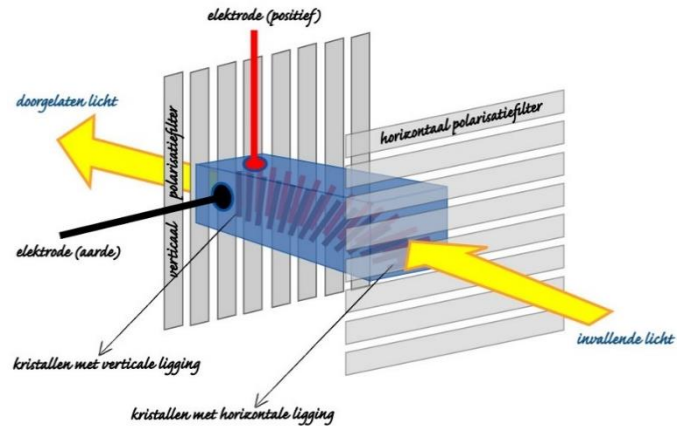
De aansluiting van een I²C is relatief eenvoudig. Je verbindt de SDA / SCL lijn met de overeenkomstige SDA / SCL lijn van je microcontroller. Daarnaast verbind je nog een +5V / GND. De helderheid van de LCD kan je instellen met de potentiometer op de I²C.



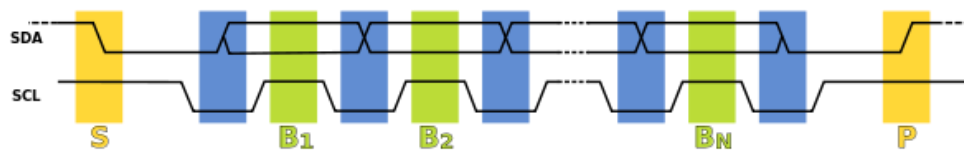
Werking LCD:

De werking van een LCD is redelijk complex, vandaar onderstaande info ter verduidelijking.

De afkorting LCD staat voor Liquid Crystal Display, oftewel vloeibare kristallen beeldscherm. De lcd-cel bestaat uit twee glasplaten die met een fotolithografieproces voorzien zijn van elektroden van ITO (indium-tinoxide). Daartussenin zit een laagje vloeibaar kristal (LC). Aan de buitenzijden van de cel zitten twee polarisatiefilters geplakt. Door op de onderste glasplaat verticale ITO-banen en op de bovenste plaat horizontale banen te etsen krijg je een raster van ITO-sporen waarmee je op elk 'kruispunt' een spanningsverschil kan opleggen. Door deze spanning kan de gedraaide structuur worden beïnvloed en zodoende de lichtdoorlaat worden geregeld. Nu kunnen pixels worden gecreëerd die elk afzonderlijk aan of uit kunnen worden gezet.

**Werking I²C:**

I²C werkt op basis van twee buslijnen, namelijk SDA (serial data) en SCL (serial clock). Over de SDA-lijn wordt de data verzonden en over de SCL-lijn wordt het kloksignaal verzonden. In het onderstaande timingdiagram wordt verduidelijkt hoe de SDA en SCL samenwerken:

**De werking van I²C dataoverdracht:**

Data verzenden wordt geïnitieerd met een STARTbit (S). De SDA wordt omlaag getrokken, terwijl de SCL hoog blijft.

De SDA verstuurt de eerste bit terwijl SCL laag gehouden wordt (gedurende de blauwe tijdsbalk.). De data wordt ontvangen als SCL naar omhoog gaat (groen).

Als de overdracht compleet is wordt een STOPbit (P) verzonden door SDA lijn omhoog te trekken, terwijl SCL continu hoog gehouden wordt.

Om storingen te voorkomen wordt het niveau van de SDA veranderd op de dalende flank (overgang van hoog naar laag) van SCL. Het uitlezen gebeurt op de stijgende flank (de overgang van laag naar hoog) van SCL.

Programma:

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x38,16,2); // Zet the LCD address op 0x38 voor
een 16x2 display

void setup()
{
  lcd.init(); // initialiseer de LCD
  lcd.backlight(); // Leg de verlichting aan

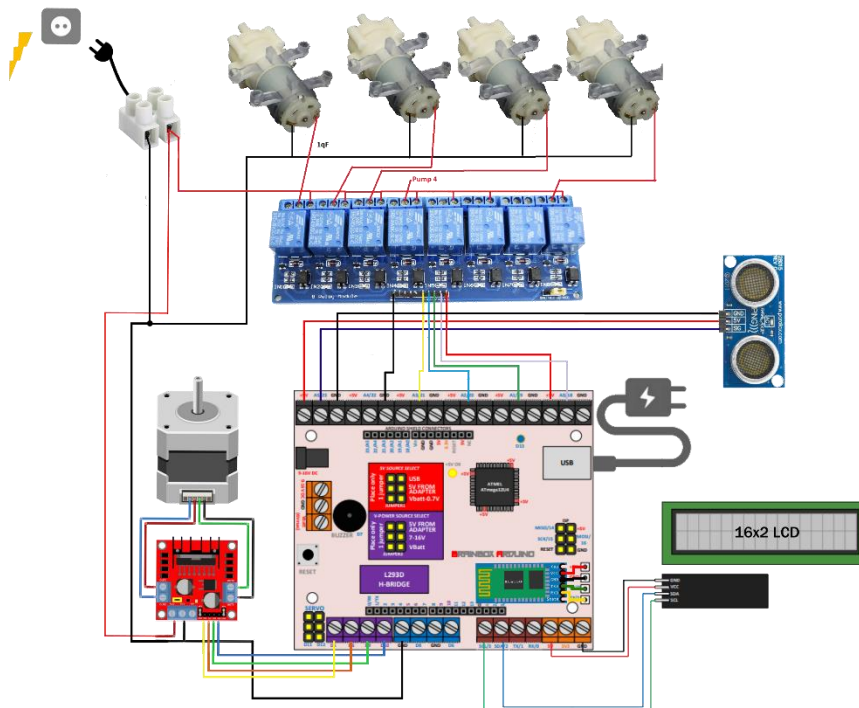
  lcd.setCursor(1,0);
  // Zet de cursor op het 1ste karakter van de 1ste rij

  lcd.print("COCKTAILMAKER");
  // Print "COCKTAILMAKER" op het scherm

  lcd.setCursor(1,1);
  // Zet de cursor op het 1ste karakter van de 2de rij

  lcd.print("GIP LUKAS");
  // Print "GIP LUKAS" op het scherm
}

```

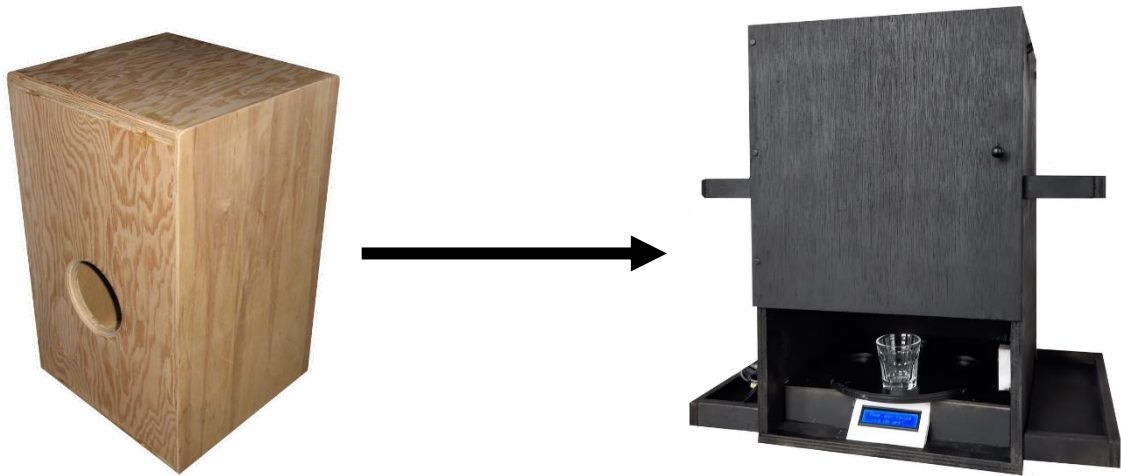
2.7 Bedradingschema

3.0 Externe onderdelen:

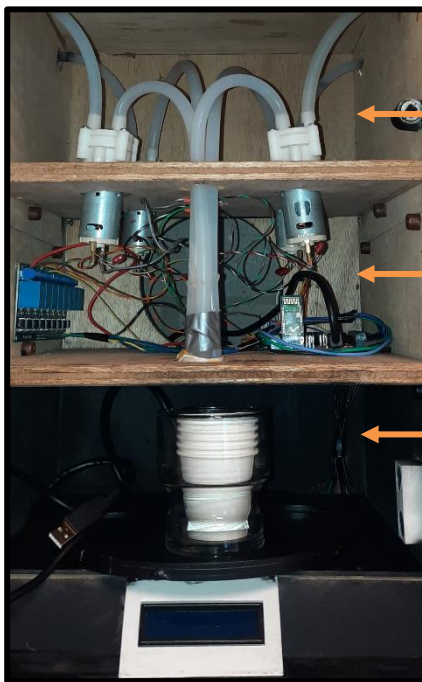
3.1 Houten kast

Als kastje gebruikte ik een oude cajon, een houten muziekinstrument.

Hierdoor had ik al een basis waarop ik kon verder bouwen, waardoor ook de kostprijs gedrukt kon worden.



Langs de zijkant zijn er 2 plankjes gemonteerd waar de flessen op komen te staan. Er is ook een deurtje voorzien aan de voorkant voor een makkelijke toegang aan de elektronica.



Ik verdeelde de binnenkant in 3 delen:

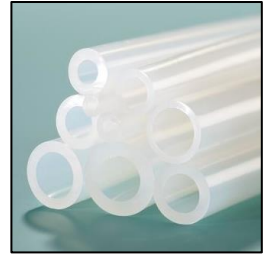
een deel waar alle pompjes geplaatst zijn

een deel voor alle elektronica

een plaats om de glazen te plaatsen

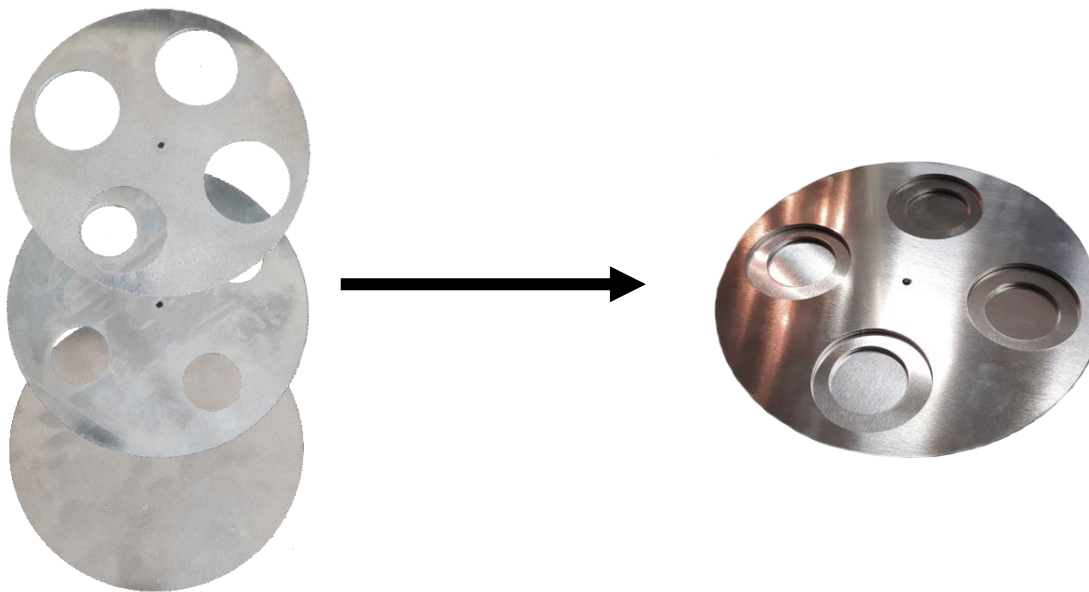
3.2 Slangen

Om de drank te transporteren van de fles naar het glas gebruik ik silicone buizen. Ze zijn smaakloos en geschikt voor voeding.



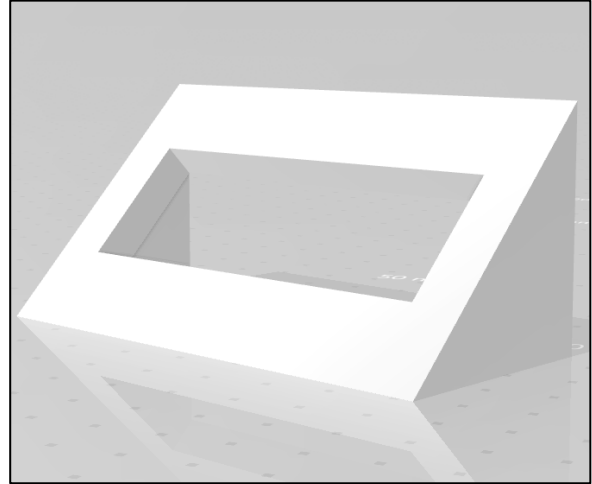
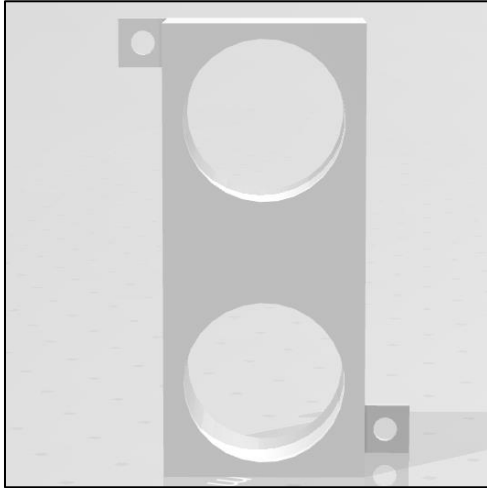
3.3 Draaitafel

De draaitafel is vervaardigd uit 3 gefreesde aluminium plaatjes die samen werden gelast tot 1 geheel. Het werd vervaardigd in de metaalafdeling van WAAK in Kuurne.



3.4 3D-Shields

Om wat draden te verbergen en om het geheel visueel aantrekkelijker te maken heb ik shields laten 3D-printen voor de ultrasoon sensor en het LCD-display. Om de stukjes te tekenen maakte ik gebruik van Solid-Edge.



Besluit:

Ik ben zeer tevreden over het eindresultaat, alsook de weg er naar toe. Ik was zeer enthousiast over mijn idee en had dus veel motivatie om er het hele jaar door hard aan te werken. Het eindresultaat was ruim op tijd klaar. Hierdoor moest ik mij niet haasten om de documentatie van dit project samen te stellen. Dit was een groot pluspunt aangezien ik nu veel meer tijd had om afbeeldingen & video's uit te werken.

Er zijn enkele dingen die ik zou veranderen moest ik dit project opnieuw starten:

- Ik zou de communicatie volledig baseren op wifi en niet op bluetooth. Hierdoor kan alle communicatie gebeuren via 1 netwerk. Ook de afstand tussen de zender en ontvanger is geen knelpunt meer.
- Ook de vorm en indeling van de kast kan beter. Voor de elektronica had ik weinig plaats terwijl ik er voor de pompen teveel had. Dit zorgde ervoor dat het werken aan de elektronica zeer moeilijk verliep nadat alles al bevestigd was in de kast.
- In het begin van het jaar werd er gevraagd om met een bedrijf samen te werken. Dit viel echter in het water door het COVID-19 virus. Toch heb ik mijn draaitafel laten frezen & lassen in WAAK. Wanneer er niet moest gewerkt worden met een bedrijf kan je de draaitafel beter vervaardigen uit kunststof. Hierdoor zou je een veel minder krachtige motor nodig hebben.
- Mijn project werd veel uitgebreider dan verwacht, waardoor er veel meer componenten aanwezig zijn dan origineel gepland. Als ik dit op voorhand had geweten, dan maakte ik een printplaat. In dit geval zouden veel minder draden aanwezig zijn waardoor de algemene uitstraling van de elektronica er veel netter & professioneler zou uitzien

Dankwoord:

Ik wil graag dhr. Hertens bedanken voor het begeleiden van dit project. Ook mijn klasgenoten verdienen een plaats in mijn dankwoord. Zij hielpen me als ik ergens vastzat. Ze lieten me mijn eindwerk ook zien vanuit hun perspectief. Hierdoor kreeg ik veel inspiratie. Het 3D-printen gebeurde door Gauthier Vanhove (leerling in het 6^{de} Industriële Wetenschappen), waarvoor dank.