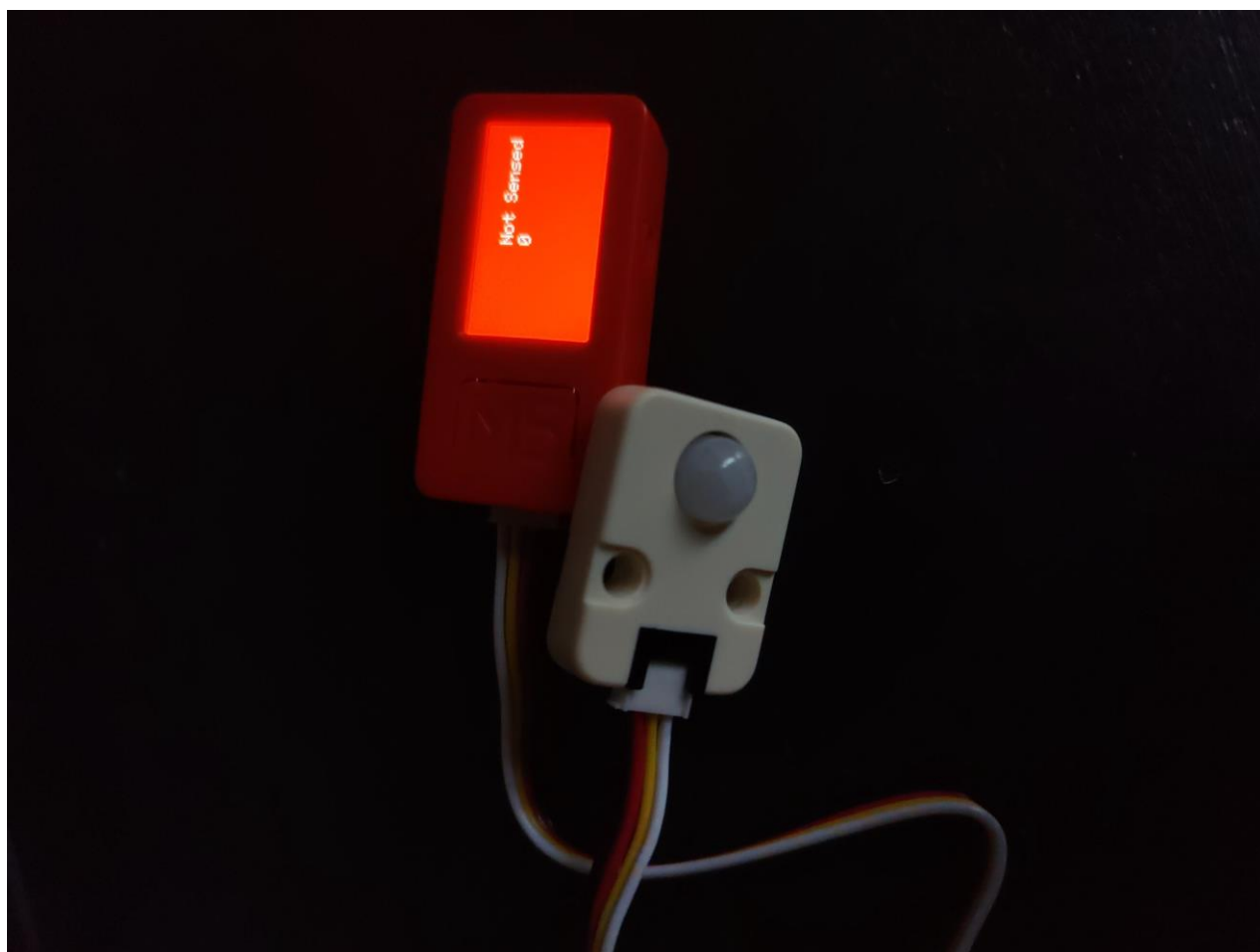


# Begivenhed detektor



Dato: 03/12/2023

Fag: Informatik B

Skole & Klasse: NEXT Københavns Mediegymsium, M3vf If

Elever: Zhentao Wei & Carl Benjamin S. Dreyer

Lærer: Daniel Peter Withenstein (dapw)

## Kravsspecifikation

1. I skal lave et HMI-interface der enten udvider eller erstatter en/flere af de menneskelige sanser.
2. vælg en af de fem HMI metoder/områder til at løse jeres problem.
3. Brug design tænkning til at udarbejde en eller flere prototyper
4. Test ved at lave en komparativ analyse på baggrund af Ergonomics kriterierne på jeres prototype og et eller flere eksisterende interfaces.
5. Alle faserne i design tænkning dokumenteres i en rapport.
6. Desuden skal i bruge Scrum til planlægningen af jeres arbejde.

## Designovervejelser

Hvorfor:

- Mennesker prøver altid at udvide sig selv
- Mennesker prøver altid at forbedre sig selv
- Man kan tjene penge af produkterne man laver
- Det kan afstatte de ting handicappede har mistet
- Det kan hjælpe med at gøre livet nemmere

Vi vil fokusere på at tilføje en ny sans:

- Ved ikke hvornår ens mor kommer ind i værelset (En lyd som kun unge kan høre)
- Ved ikke hvornår ens 3d printer er færdig
- Læreren ved ikke hvornår eleverne er forvirret eller irriteret (2 ideer)
- Man ved ikke om at man har pisset sig selv
- Ved ikke hvordan vejret er udenfor

## Define

Vi har et problem med at mennesker ikke har en ekstra sans som signalerer om forholdende uden for personens naturlige sanse rækkevidde.

## Problemformulering

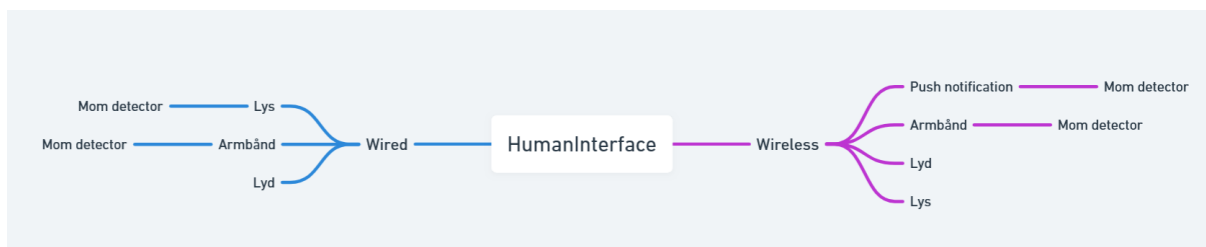
Hvordan kan vi tilføje en sans, som advarer brugeren om begivenheden, som er uden for den brugerens naturlige sanse rækkevidde?

## Krav

1. Produktet skal kunne observere, når begivenheden sker.
2. Det skal have evnen til at advare brugeren på en diskret måde.
3. Produktet skal kunne advare så tidligt som muligt.
4. Produktet skal være simpelt at producere.

## Brainstorm

Vi har brugte en mindmap til at brainstorme vores ideer. Her kan vi kategorisere løsningerne som vi finder på.



## Ideate (Løsning)

Ud fra brainstormen har vi udvalgt 3 forskellige løsninger til vores problemformulering:

- Løsning 1: En armbånd som bliver aktiveret trådløst af en sensor.
- Løsning 2: En push notifikation på telefonen som bliver aktiveret af en sensor
- Løsning 3: En ændring i LED lys som bliver aktiveret af en sensor

## Kravsmatrix

Baseret på de tre løsninger vi har udvalgt, skal vi nu foretage en endelig udvælgelse og vælge én af dem til at udvikle som en prototype. For at træffe det bedst mulige valg blandt de tre muligheder vil vi benytte os af en kravmatrix. Denne matrix vil blive sammensat ved hjælp af de fire tidligere fastlagte krav og vi vil evaluere hvilken af de tre løsninger der bedst opfylder disse krav. På denne måde sikrer vi at den valgte prototype vil være i stand til at imødekomme de definerede krav på den mest optimale måde.

Krav

Krav 1: Produktet skal kunne observere, når begivenheden sker. (vægt: 5)

Krav 2: Det skal have evnen til at advare brugeren på en diskret måde. (vægt: 2)

Krav 3: Produktet skal kunne advare så tidligt som muligt. (vægt: 3)

Krav 4: Produktet skal være simpelt at producere. (vægt: 4)

Krav	1	2	3	4	Total score
Vægt	5	3	3	4	
Løsning 1 (armbånd)	$5 \cdot 5 = 25$	$5 \cdot 3 = 15$	$5 \cdot 3 = 15$	$3 \cdot 4 = 12$	67
Løsning 2 (notifikation)	$5 \cdot 5 = 25$	$5 \cdot 3 = 15$	$5 \cdot 3 = 15$	$1 \cdot 4 = 4$	59
Løsning 3 (lys)	$5 \cdot 5 = 25$	$3 \cdot 3 = 9$	$5 \cdot 3 = 15$	$4 \cdot 4 = 16$	65
Løsning 4 (lyd)	$5 \cdot 5 = 25$	$1 \cdot 3 = 3$	$5 \cdot 3 = 15$	$4 \cdot 4 = 16$	59

## Valg af løsning

Baseret på vores kravvurdering har vi valgt den mest velegnede løsning, da den opnåede den højeste score i vores kravvurdering. Vores valgte løsning er et armbånd, der giver brugeren besked, når en trådløs sensor aktiveres. Armbåndet vil informere brugeren ved at vibrere når sensoren registrerer en aktivering. Sensoren sender et signal til armbåndet så snart den registrerer en bevægelse foran den. Denne sensor kaldes en passiv infrarød sensor eller PIR-sensor.

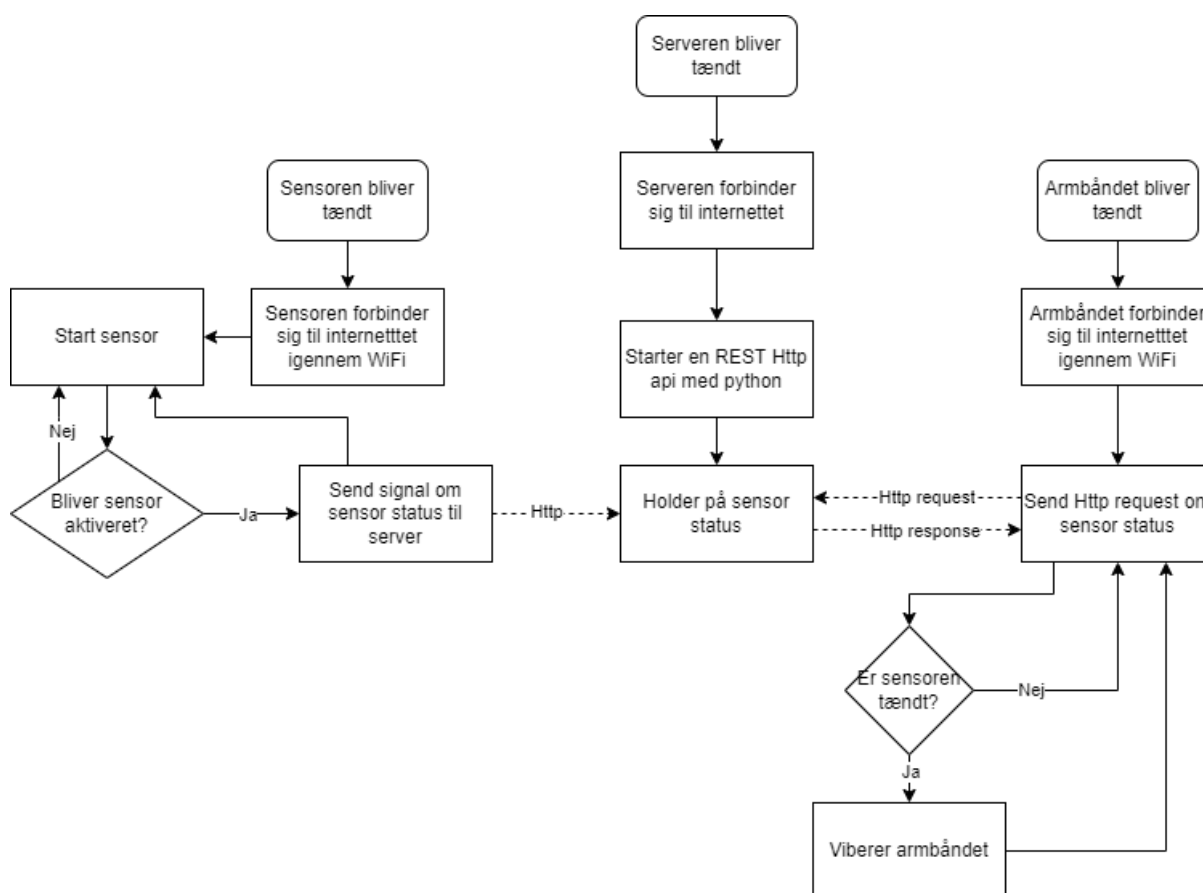
Som beskrevet i undersøgelsen omkring forskellige HMI-teknologier, vil løsningen være en udvidelse til burgerens sanser, og fungere som en 6. sans <sup>1</sup>.

---

<sup>1</sup> Cannan, J., & Hu, H. (s. 3)

## Implementering af kerne dele af jeres product

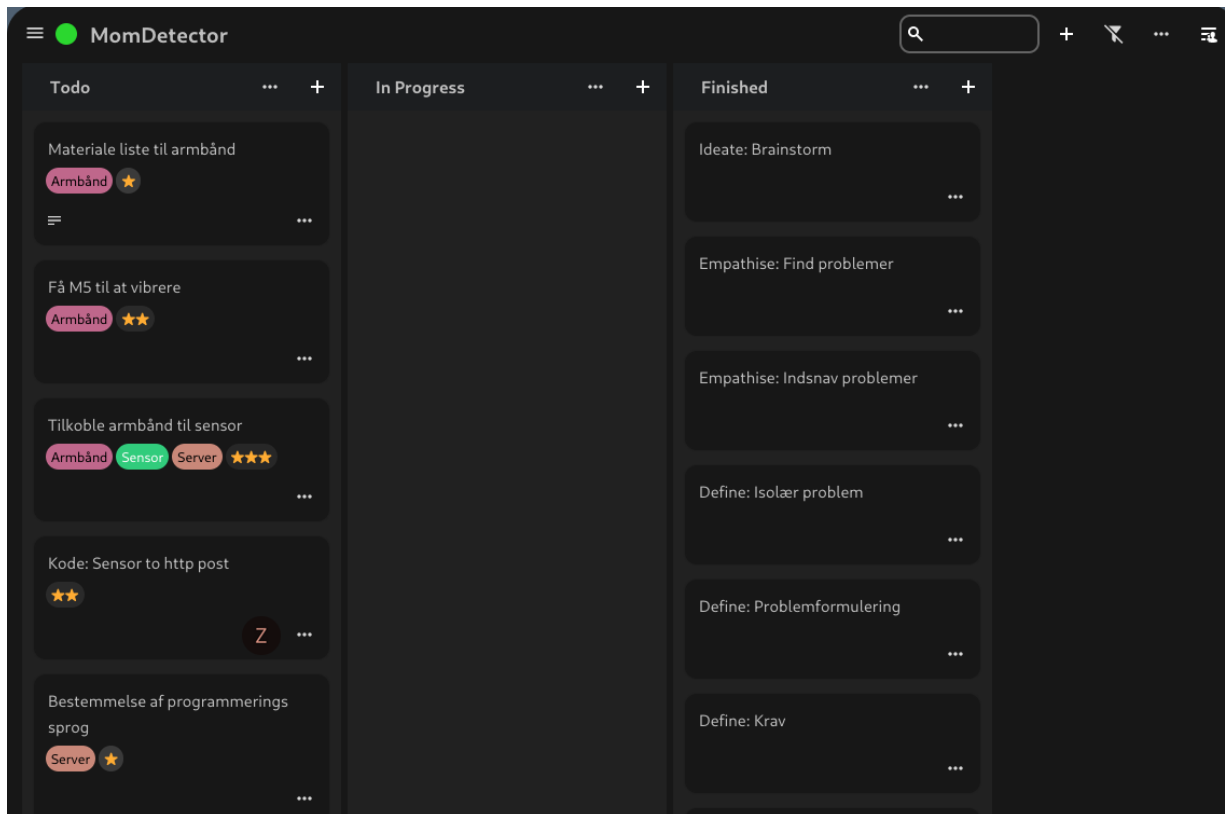
Produktet består primært af tre systemer; en sensor, et vibrerende armbånd og en server. Sensoren står for at detektere hvis en person er på vej ind på værelset. Armbåndet får information omkring status fra sensoren gennem en tredje part server. Det er ikke teknisk nødvendigt at have en separat server - armbåndet kunne bare direkte forbinde sig til sensoren, men det giver en række fordele, som f.eks. at det vil kunne gøre det muligt at få besked fra hele verdenen, derudover vil det være muligt at forbedre detektionsmetoden ved at tilføje flere sensorer, som forbinder sig til samme server. Det er allerede ret indviklet at holde styr på hvad og hvem der sender data fra og til hindanen, så derfor er det oplagt at udarbejde et flowdiagram, som beskriver hvordan hele produktet virker, og hvilke beskeder der vil blive sendt og hentet over nettet:



Som set foroven, starter det hele med at systemerne starter op – sensoren, serveren og armbåndet starter. Serveren forbinder sig til internettet, hvor den opretter en REST HTTP API, som både sensoren og armbåndet vil kommunikere med. Serverens job her, er at holde status på sensoren i en boolean. Efter sensoren er startet, forbinder den til internettet, og sender en besked til serveren hvis sensoren bliver aktiveret. Armbåndet starter med at forbinde til internettet, og "poller" derefter serveren omkring sensorens status fra serveren i et fast interval. Hvis status er true, indikere det at der er en person detekteret og armbåndet vil derfor vibrere i en kort periode for at informere brugeren. Hvis status er false, vil den gå tilbage til at poll serveren.

Problemet med at gå i gang med at implementere produktet nu, er at der er ingen fornemmelse i forhold til hvornår produktet er færdigudviklet og om det kan nås inden deadline. Ved at udnytte scrum, kan vi øge kvaliteten og effektiviteten af arbejdet<sup>2</sup>.

Flowdiagrammet giver et godt overblik over hvilke nogle opgaver som skal løses og det er derfor et oplagt sted at starte med at dele opgaverne op i små bider og samle dem i et trello board:



Her giver vi hver opgave en mængde stjerner, som indikerer hvor svært det er at løse opgaven. Det giver et godt overblik over hvilke opgaver man i dette sprint kan nå at færdiggøre, og medfører at arbejdsbyrden er konstant gennem hele forløbet, ligesom manifesteret for agile development's 8. princip siger: "*Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.*"<sup>3</sup>.

<sup>2</sup> Ken Schwaber & Jeff Sutherland, 2020

<sup>3</sup> *Manifesto for Agile Software Development*

Server

```
from threading import Timer

SensorActivated = False
sensor_timer = None

@app.route("/Sensor", methods=["GET", "POST", "OPTIONS"])
def SensorActivate():
    global SensorActivated, sensor_timer
    SensorActivated = True

    if sensor_timer is not None:
        sensor_timer.cancel() # Cancel the previous timer

    sensor_timer = Timer(1, DelayDeactivate)
    sensor_timer.start()

    return "Okay"

@app.route("/SensorActivation", methods=["GET"])
def SensorState():
    global SensorActivated
    return str(SensorActivated)

def DelayDeactivate():
    global SensorActivated
    SensorActivated = False
```

Serveren kører en Python REST API med Flask biblioteket. Den sætter 2 HTTP endpoints op; "/Sensor" og "/SensorActivation". Når sensoren detektere en person skal den rapportere det til "/Sensor" endpoint. Armbåndet kan derefter så læse værdien af sensoren ved "/SensorActivation".

## Sensor

```
sensor.ino

1 #define PIN 36
2 #define PORT
3 #define FCDM "188.181.83.156"
4 #define URL "http://188.181.83.156:4567/Sensor"
5
6 #include <M5StickCPlus.h>
7 #include <WiFi.h>
8 #include <WiFiMulti.h>
9 #include <HTTPClient.h>
10
11 WiFiMulti wifMulti;
12 HTTPClient http;
13
14 const char* ssid = "Next-Guest";
15 int state = 0;
16
17 void setup() {
18     M5.begin();
19     pinMode(PIN, INPUT);
20
21     wifMulti.addAP(ssid, "");
22 }
23
24 void loop() {
25     int newState = digitalRead(PIN);
26     Serial.printf("sensing %d\n", newState);
27
28     if (state != newState)
29         sendHTTP(newState);
30
31     state = newState;
32
33     delay(100);
34     M5.update();
35 }
36
37 void sendHTTP(int newState) {
38     if (!newState) return;
39     if (wifMulti.run() != WL_CONNECTED) return;
40
41     http.begin(URL);
42     int httpCode = http.GET();
43     http.end();
44     Serial.println(httpCode);
45
46 }
```

Sensoren virker ved at den først forbinder sig til internettet (l. 21), som vist på flowdiagrammet. Derefter læser den sensorens status og hvis den er ændret sig, sender den en HTTP besked til serveren omkring dens status (l. 28 –29).



## Armbånd

Implementationen af armbåndet virker meget som beskrevet på flowdiagrammet:

```
armbaand.ino

1 #include <Arduino.h>
2 #include <WiFi.h>
3 #include <WiFiMulti.h>
4 #include <HTTPClient.h>
5 #include <M5StickCPlus.h>
6
7 #define URL "http://188.181.83.156:4567/SensorActivation"
8 WiFiMulti wifMulti;
9 const char* ssid = "Next-Guest";
10
11 void setup() {
12   M5.begin();
13   wifMulti.addAP(ssid, "");
14   M5.Lcd.print("Connecting...");
15   while (wifMulti.run() != WL_CONNECTED) {
16     M5.Lcd.println(".");
17     delay(100);
18   }
19 }
20
21 void loop() {
22   if (wifMulti.run() != WL_CONNECTED) return;
23   HTTPClient http;
24
25   http.begin(URL);
26   int httpCode = http.GET();
27   if (httpCode < 0) { Serial.printf("HTTP Error: %s\n", http.errorToString(httpCode).c_str()); return; }
28   if (httpCode == HTTP_CODE_OK) {
29     String response = http.getString();
30     Serial.printf("Res: %s\n", response.c_str());
31     const char* res = response.c_str();
32     if (strcmp(res, "True") == 0)
33       vibrateMotor();
34     else if (strcmp(res, "False")
35             )
36       stopVibrate();
37     else
38       M5.Lcd.fillScreen(RED);
39   }
40   http.end();
41   delay(100);
42 }
```

Den starter med at forbinde til internettet (l. 13-18). Derefter i et loop requester den sensorens status fra serveren igennem dens REST API (l. 23-26). Hvis status er true – sensoren er tændt, så vibrere den motoren, ellers stopper den med at vibrere (l. 32-35).

Ligesom artiklen “What are Human Machine Interfaces and Why Are They Becoming More Important?” beskriver, så giver armbåndet information til operatøren (brugeren) at der er bevægelse, ved at vibrere en motor <sup>4</sup>.

---

<sup>4</sup> machinedesign.com, 2015

## Test af jeres prototyper

Der bliver brugt test-case metoden til at teste dette produkt. Test-case metoden bliver udført ved at opstille nogle scenarier for at se hvordan et produkt vil reagere. Baseret på reaktionen af produktet, så kan scenariet/casen enten være en "pass" eller en "fail". "pass" betyder at produktet er bestået og modsat er "fail" selvfølgelig ikke-bestået. Hvis produktet er bestået, så kræver produktet ingen ændringer i scenariet. Men hvis den ikke består casen, så betyder det at produktet kræver nogle forbedringer for at bestå casen. Alle cases skal være bestået før man forsætter med et produkt.

### Test cases

Her bliver der opstillet 5 test-cases:

Nr.	Beskrivelse	Pass/Fail
1	Kan sensoren aktivere armbåndet?	Pass
2	Kan sensoren aktivere armbåndet indenfor 3 sekunder?	Pass
3	Kan sensoren aktivere armbåndet fra lang afstand?	Pass
4	Sensoren bliver ikke aktiveret uden forstyrrelse?	Pass
5	Kan armbåndet notificere brugeren diskret?	Pass

Det ser ud til at produktet har bestået alle cases, hvilket betyder at der kan forsættes.

## Refleksion af jeres arbejdsmetode på baggrund af Scrum

Scrum har hjulpet udviklingsprocessen ved at indele arbejdsbyrden op i overkommelige sprints. Baseret på erfaring med tidligere projekter, hvor der ikke blev anvendt scrum, har scrum medført mange positive ting, som:

- Overkommelige sprints – konstant arbejdsbyrde
- Ingen overarbejde
- Godt overblik
- Kan approksimere hvornår produktet er færdig baseret på ens backlog

På baggrund af foroven nævnte ting, har scrum været en fornøjelse at bruge.

## Konklusion

I dette projekt har vi udviklet en innovativ HMI-løsning, der udvider menneskelige sanser gennem et trådløst armbånd, som vibrerer i respons til en tilknyttet sensor. Gennem en grundig og metodisk proces, der inkorporerede design tænkning og Scrum metodologi, ledte vi projektet fra idegenerering til succesfuld prototypetesting. Vores strategiske tilgang, fremhævet ved brugen af en kravsmatrix, førte til valget af armbåndet som den mest hensigtsmæssige løsning. Testningen bekræftede prototypens evne til effektivt at advare brugere om begivenheder uden for deres naturlige sanseområde.

## References

- Schwaber, K., & Sutherland, J. (u.d.). *The Scrum Guide*. Hentet 27. 11 2023 fra <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- Cannan, J., & Hu, H. (u.d.). *Human-Machine Interaction (HMI)*. Hentet 27. 11 2023 fra <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.186.1644&rep=rep1&type=pdf>
- Manifesto for Agile Software Development*. (u.d.). Hentet 27. 11 2023 fra <https://agilemanifesto.org/iso/en/manifesto.html>
- What are Human Machine Interfaces and Why Are They Becoming More Important?* (16. Sept 2015). Hentet 27. 11 2023 fra <https://www.machinedesign.com/automation-iiot/article/21834387/what-are-human-machine-interfaces-and-why-are-they-becoming-more-important>