



PROJEKT: MAKEY MAKEY

Af Carl-Benjamin, Morgan & Sebastian



Indholdsfortegnelse

Indholdsfortegnelse	1
Kravspecifikation.....	2
En installation som gør brug af Makey Makey	2
Den skal være specifikt tilpasset et sted på skolen.	2
Forklaring af jeres system. Dvs. de ting som I har brugt for at få jeres installation til at fungere og hvordan de spiller sammen.	2
Forklaring af jeres basis ide, og hvordan denne skulle "ændre" det sted som I har opsat det.....	2
Begrundelse for valg af kodesprog	2
Design.....	2
Forklaring af fysisk udformning af produkt.....	2
Flowchart	3
Forklaring af Affordance, og hvorfor man skulle have lyst til at benytte sig af jeres installation. Hvorfor går man ikke bare forbi Implementering.....	4
Funktions-duelig prototype	4
Udvalgte dele af koden gennemgås i detaljer.	4
Ændringer i projektet og Design Thinking.....	5
Forbedringer af produkt	7
Konklusion.....	7

Kravspecifikation

En installation som gør brug af Makey Makey

Vores installation gør brug af en Makey Makey og en Raspberry Pi.

Den skal være specifikt tilpasset et sted på skolen.

Vores installation finder sted i gangen mellem fredagscaféen og musiklokalet, hvor der er helt mørkt, da det giver mening at gøre dette sted "spooky".

Forklaring af jeres system. Dvs. de ting som I har brugt for at få jeres installation til at fungere og hvordan de spiller sammen.

Vi brugte Makey Makey'en til at tage imod input når døren åbnede og så brugte vi Raspberry Pi'en til at køre et script derefter. Vi har heldigvis muligheden for at bruge en Raspberry Pi 4 til vores projekt. Vi har valgt at programmere vores program til afspilning af lyd i Python.

Forklaring af jeres basis ide, og hvordan denne skulle "ændre" det sted som I har opsat det.

Vores idé er, at der i gangen mellem fredagscaféen og musiklokalet bliver spillet en spooky lyd, f.eks. "Minecraft ambient cave sounds". Formålet med at placere vores produkt dette sted, er at gøre gennemgangsoplevelsen mere interessant, for skolens elever og lærere. Vores målgruppe er elever på Københavns Mediegymsnasium, hvilket betyder at vi skal målrette vores idé mod vores målgruppe - nemlig HTX-elever. Mange HTX-elever har spillet videospil, som det kendte sandbox spil, Minecraft. Derfor er det aktuelt at indføre en idé hvor Minecraft indblandes og på den måde målretter vi os specifikt mod vores målgruppe, da de kan relatere til spillet.

Begrundelse for valg af kodesprog

Vi har valgt Python som vores kodesprog, da Python har mange forskellige biblioteker man kan implementere og bruge i vores kode. Derudover er det nemt og hurtigt at implementere features, hvilket gør det hurtigere, nemmere og smartere at udvikle produktet.

Design

Forklaring af fysisk udformning af produkt.

Nede i kælderens er der en gang der fører fra fredagscaféen og ind til musik. Her er en automatisk dørlukker på én af dørene, som er fastgjort på væggen og døren. Vi har placeret et krokodillenæb på hver af de stænger som får dørlukkeren til at hænge sammen. Når døren er lukket, har krokodillenæbene konstant kontakt. Når krokodillenæbene ikke længere har

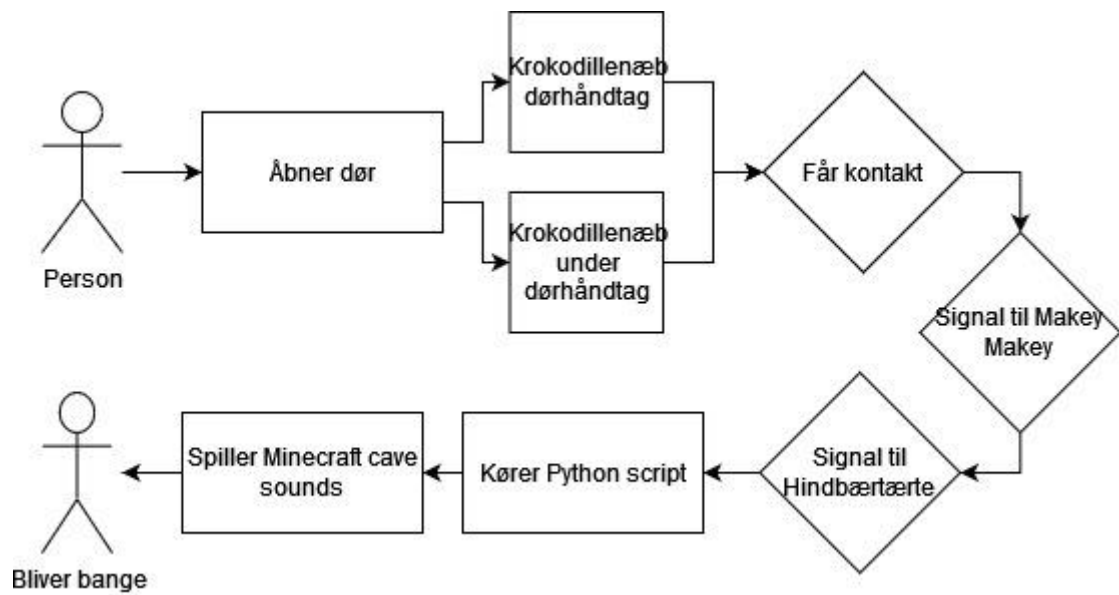
kontakt, får Raspberry PI'en signal om at den skal afspille en lydfil. Foruden ses billeder af installationen:

Figur 1 Døren er lukket - kontakt

Figur 2 Døren er åben - igen kontakt



Flowchart



Forklaring af Affordance, og hvorfor man skulle have lyst til at benytte sig af jeres installation. Hvorfor går man ikke bare forbi Implementering Affordance forklare om hvor intuitivt noget er designet. Det handler om at man ved hvordan man skal bruge en ting, som f.eks. en hammer, uden at fået noget at vide. Hvis vi tager udgangspunkt i en hammer, så kan man nemt finde ud af hvordan man skal holde den på grund af håndtaget. Man ved hvordan man skal hamre med den da den har et metal hoved hvor den ene side er større end den anden og er flad.



Vores installation er lavet sådan at man ikke vil opdage den. Den fungerer sådan at når man åbner en dør så vil den spille en uhyggelig lyd, af den grund vil man ikke kunne se den. Den er sat op i en lang mørk gang af denne af denne grund vil man ikke kunne gå forbi den uden af den aktiveres.

Funktions-duelig prototype

<https://www.youtube.com/shorts/7osznxwecG8>

Udvalgte dele af koden gennemgås i detaljer.

Koden gør brug af pygame biblioteket. Pygame tilbyder en API, som gør at vi kan afspille lydfiler nemt. Dette gør det lettere for os programmører, da vi ikke behøver at have et ekstensivt kendskab til hvordan koden bag lydafspilning virker. Dette betyder altså at vi kan fodre en lydfil til pygame, også er det op til pygame for at få lydfilen afspillet. Koden virker ved at den går ind i et uendeligt loop, og tjekker om "a"-knappen bliver sluppet. Kontakten, som sidder på døren, er altså forbundet til Makey Makey'ens a knap, hvilket informerer programmet. Man kan se på linje 14, at den går ind i et uendeligt loop, hvor den tjekker når a-knappen bliver løsladt.

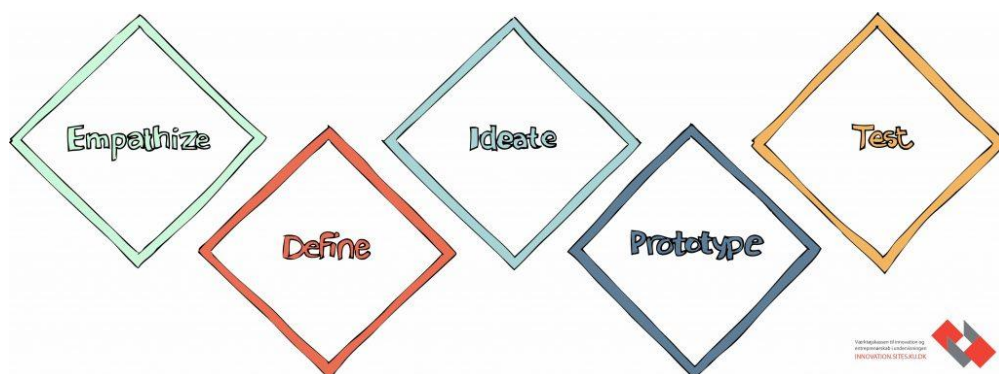
```
playsound.py

1 // Code heimport os
2 import vlc
3 import pygame
4 from pygame.locals import *
5 import time
6 import random
7
8 pygame.init()
9 pygame.mixer.init()
10 pygame.mixer.music.set_volume(1)
11
12 screen = pygame.display.set_mode((500, 500))
13 running = True
14 while running:
15     for event in pygame.event.get():
16         if event.type == pygame.KEYUP:
17             if event.key == K_a:
18                 time.sleep(1.5)
19
20                 randomFile = random.choice(os.listdir("Minecraft_cave_sounds"))
21                 pygame.mixer.music.load("Minecraft_cave_sounds/" + randomFile)
22                 pygame.mixer.music.play()
23                 break
24
25             elif event.key == K_ESCAPE:
26                 running = False
27 pygame.quit()
```

Når programmet får information fra Makey Makey'en om at knappen "a" er blevet sluppet, venter programmet 1,5 sekunder (Se linje 18). Derefter vælger programmet en tilfældig lydfil fra en folder (Se linje 20), og afspiller den.

Ændringer i projektet og Design Thinking

Gennem projektet har vi lavet en række ændringer på baggrund af empiri fra de test vi har lavet. Design processen er en iterativ proces, hvilket indebærer at man tester produktet og laver ændringer baseret på den indsamlede empiri flere gange, indtil man er tilfreds. Design processen kan opdeles i fem forskellige faser, som set på figuren forneden:



Den første fase er kaldt "Emphasize", hvor det handler om at dykke ned i brugerens behov. I denne fase holdt vi fokus på at vores brugere var gymnasieelever, som går på et HTX-gymnasie. Under analysen af vores målgruppe fandt vi bl.a. ud af at der er stor chance for at elever kender til forskellige computerspil. Næste fase, er "Define" fasen, hvor det handler om at skabe mening ud fra empiri man har fået indsamlet fra de tidligere målgruppeanalyser. I denne fase fandt vi bl.a. ud af at vores løsning skulle inddrage computerspil, da vores målgruppe har en speciel relation til computerspil og vil dermed forbedre gennemslagskraften af løsningen.

Efter den fase, går udviklingsprocessen ind i en fase kaldt "Ideate", som handler om at finde på løsninger, som afdækker brugerens behov ud fra tidligere analyser. Her fandt vi på idéen om at afspille uhyggelige Minecraft lyde når man går nede i den kedelige kældergang. Dette stemmer overens med at der skal være computerspil involveret, og styrker relationen mellem løsningen og brugeren.

Den næstsidste fase er kaldt "Prototype", og handler om at udarbejde en prototype. Vores første prototype bestod af at der var en plade med aluminium folie under håndtaget på døren. Dette gjorde at når man træk langt nok ned i håndtaget, så ville håndtaget og aluminiumfolien have kontakt, som medførte at Makey Makey'en fik besked om at der var kontakt og sendte informationen videre til et Python program der kørte på en Raspberry PI. Efter at have testet det, fandt vi ud af at der var nogle problemer med kontakten mellem håndtaget og aluminiumsfolien, hvilket gjorde at lyden kun blev afspillet en gang imellem. Derudover blev aluminiumfolien bøjet efter hver gang man træk i håndtaget, hvilket medfører at denne løsning ikke vil holde ret længe inden den går ville gå i stykker. Efter at have indsamlet empiri fra vores test af vores prototype, gik vi hen til "Ideate" fasen igen. Vi fandt ud af at man kan sætte kontakten et andet sted, hvor der er bedre kontakt, og som ikke bøje aluminiumfolien på samme måde.



Som man kan se på billedet, så sidder kontakten i toppen af døren, hvilket er en bedre løsning end sidst, da den er sværere at se, har bedre kontakt med aluminiumfolien og bliver ikke slidt lige så meget efter hvert brug.

Forbedringer af produkt

Hvis vi skulle gøre det endnu bedre, så skulle det være på begge døre og ikke kun på den ene. En anden forbedring vil være at sørge for at den ikke spiller den samme lyd to gange i træk. En anden ting kunne være at den bliver ved med at spille indtil man forlader rummet.

Konklusion

I løbet af projektet har vi lavet en række ændringer på løsningen. Disse ændringer er lavet på baggrund af empiri efter at have testet de forskellige løsningsidéer. Til sidst endte vi med en løsning, som er ens med hvad vores løsningsforslag var. Løsningen formår altså at starte en Minecraft cavesound når man åbner døren.

Der er også blevet reflekteret over hvilke nogle ting, som kunne forbedres i fremtiden, som f.eks. at der ville blive afspillet lyde fra begge døre, frem for at begrænse os til den ene.