

Intime

Intime est un dispositif qui met en relation, via leur pouls, des individus évoluant dans un même espace physique. À chacune des ses extrémités, des capteurs de pouls relèvent et traduisent

 Difficulty **Medium**

 Duration **1 month(s)**

 Categories **Art, Decoration, Electronics, Machines & Tools, Play & Hobbies, Science & Biology**

 Cost **30 EUR (€)**

Contents

Introduction

Step 1 - Evolution d'Intime

Step 2 - Test des effets lumineux

Step 3 - Electronique

Step 4 - Code

Step 5 - Design de l'objet

Step 6 -

Comments

Introduction

Intime est un dispositif qui met en relation, via leur pouls, des individus évoluant dans un même espace physique. À chacune des ses extrémités, des capteurs relèvent et traduisent en flash lumineux le pouls de deux personnes saisissant l'objet. Lorsque les pouls se synchronisent, la lumière diffusée dans la barre change. À deux, ou trois, ou tout un groupe, les utilisateurs partagent une expérience commune, articulée autour du partage d'un mécanisme invisible et pourtant indispensable à la vie.

Materials

hardware électronique

neon

batteries

 Résumé Articles.pdf

Tools

Step 1 - Evolution d'Intime

Après avoir réalisé « Pulsum » (ancien version disponible sur le site du Fablab de Sorbonne Université), nous avons d'abord pensé à continuer ce projet en le reproduisant presque à l'identique, de façon plus « professionnelle ».

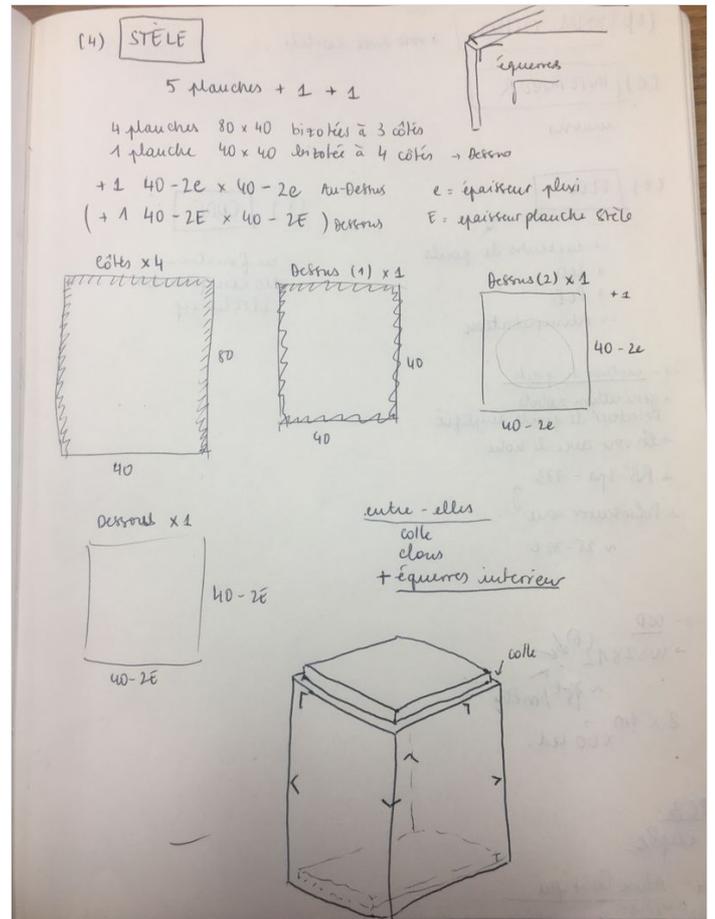
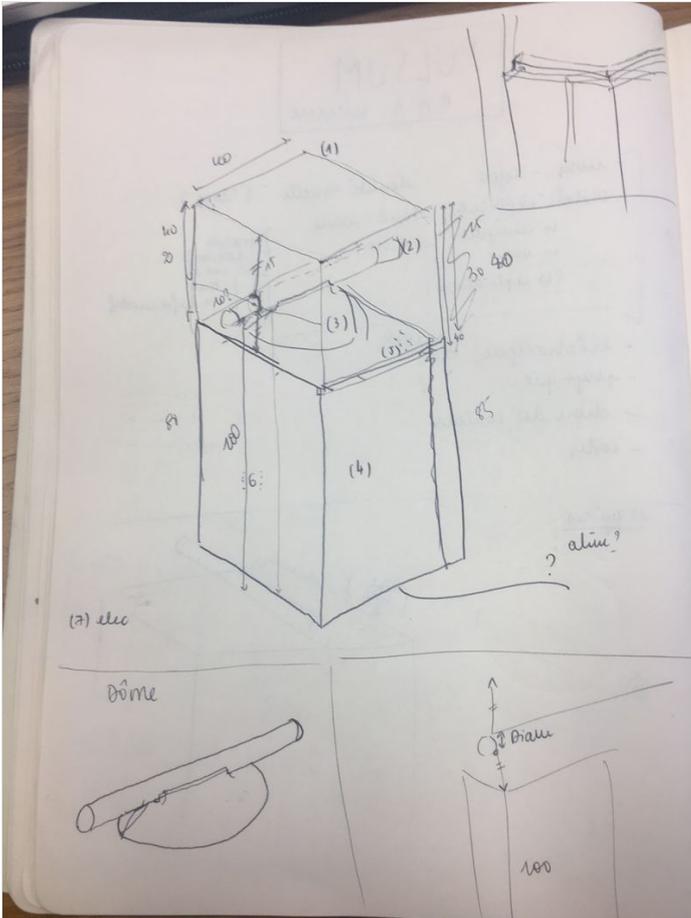
Dans un premier temps, nous avons imaginé ce dispositif comme une installation fixe avec notamment une stèle lourde structurant notre installation ainsi qu'un dôme, socle de la barre lumineuse. Voici les premières esquisses de l'évolution de ce dispositif en installation en photos.

Après réflexion nous nous sommes rendus compte que nous désirions permettre à des personnes de se lier entre elles. Après avoir lu l'article de Charles Lenay autour du croisement perceptif nous avons compris que ce qui comptait réellement était de créer un univers commun, où ces personnes pourraient agir entre elles. Par exemple, dans le monde visuel commun, le fait de voir autrui n'est pas le plus important; ce qui compte c'est de pouvoir interagir.

La synchronisation ne doit donc pas être vue comme une fin en soi mais comme la création d'un univers commun. C'est dans cette idée qu'est né In.Time : Pourquoi ne pas redéfinir la synchronisation comme phénomène dynamique qui évolue ? Pourquoi lier seulement un couple de personne ?

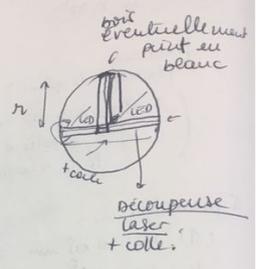
À la différence de Pulsum, In.Time a pour but de lier plusieurs personnes ensemble. Le design de Pulsum nous a paru alors bien trop complexe. Nous avons simplifié celui-ci en se limitant à la barre. L'idée est que deux personnes puissent se lier par le pouls grâce à cette simple barre puis en saisir une autre et s'associer à une personne supplémentaire, et ainsi de suite. Une chaîne de personnes liées est alors créée. L'objectif n'est plus de montrer que deux personnes sont liées mais d'agrandir ce cercle de personnes « connectées ».

De plus, nous nous sommes rendus compte lors des utilisations de Pulsum que la visualisation du moment de synchronisation des utilisateurs n'était pas claire : on pouvait voir que les deux pouls concordait mais ce phénomène n'était pas mis en avant par notre design. Nous avons alors décidé de palier à ce problème en modifiant l'effet lumineux lorsque les deux rythmes cardiaques sont synchrones : les deux bandeaux lumineux n'agissent alors plus de façon indépendante mais effectuent le même mouvement. Ce mouvement n'est plus un mouvement lumineux partant d'un utilisateur vers un autre utilisateur mais un mouvement unique partant du centre vers les deux utilisateurs. De plus la couleur change également lors de cette synchronisation.

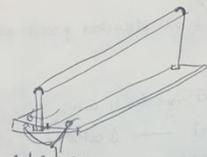


(2) BARRE

- nouvelle barre.
60-70 de long. (lorry merlin)
rayon comme l'ancien.
en forme (semi-opaque)

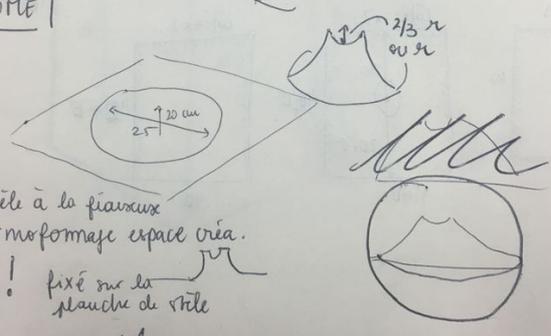


trou intérieur



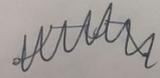
+ cales pour fixer dans la barre

(3) DÔME



- modèle à la fraiseuse
- thermoformage espace créa.

Blanc ! fixé sur la planche de stèle



(1) cube en plexi

taille: 40 x 40 x 40

en quoi? plexi total transparent

comment en bizot

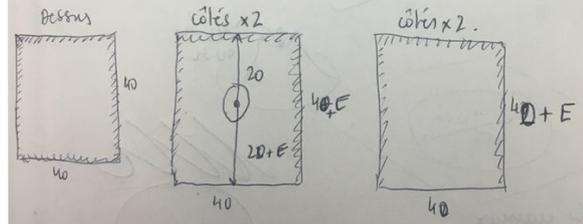
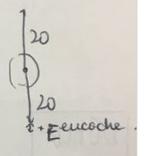
avec - fraiseuse
série new (à tester sur dents)

fixer les planches de plexi?

- 5 planches 40 sur 40
- 1 horizontale à 4 côtés
- 2 à 3 côtés → 2 des planches percées pour la barre

5 planches

- (1) 40 x 40 x 5 mm horizontale sur des 4 côtés
- 4 : 40 x 40 x 5 mm horizontale — 3 côtés
- ↳ 2 percés pour la barre.



Step 2 - Test des effets lumineux

Afin de s'assurer de l'utilisation des bandeaux de led adressables WS2812B nous avons effectué plusieurs tests sur l'IDE Arduino.

Nous avons utilisé la bibliothèque Adafruit_NeoPixel. Cette dernière est très facile à utiliser et voici plusieurs liens expliquant clairement son utilisation :

<http://www.fablabredon.org/wordpress/2017/12/17/lumiere-sur-larduino-avec-de-la-couleur-et-des-led/>

<https://www.tweaking4all.com/hardware/arduino/arduino-ws2812-led/>

Les tests lumineux les plus basiques ont d'abord été réalisés : allumage des leds, variations des couleurs, du nombre des leds, allumage de led précises.

Par la suite, nous avons travaillé sur différents effets. Les effets fournis par la bibliothèque ne répondant pas complètement totalement à nos besoins (utilisation de delay provoquant des pauses lors du programme, pas de contrôle des paramètres des effets tels que le temps de parcours etc ...), nous avons décidé de commencer à coder nos propres effets.

Les effets créés ont été les suivants : parcours des leds sur la longueur du bandeaux pendant une durée correspondant à un pouls donné, parcours symétrique de deux bandeaux à partir du centre (pouls des utilisateurs synchronisés).

Les différents problèmes rencontrés ont été les suivants : dernières led restant allumées, effet saccadé ...

Les différentes solutions trouvées ont été d'utiliser plusieurs leds pour les parcours à intensité variées afin de donner un effet plus fondu, de trouver les bonnes bornes des boucles de parcours (arrêt avant le nombre total de led).

Les vidéos montrées issues ont été effectuées après l'achat d'un néon (cf section design de l'objet).

https://wikifab.org/wiki/Fichier:Intime_Test-Lumie_re2.mov

https://wikifab.org/wiki/Fichier:Intime_Test-Lumie_re3.mov

https://wikifab.org/wiki/Fichier:Intime_Test_Lumie_re9.mov

https://wikifab.org/wiki/Fichier:Intime_Test-Lumie_re5.mov

Step 3 - Electronique

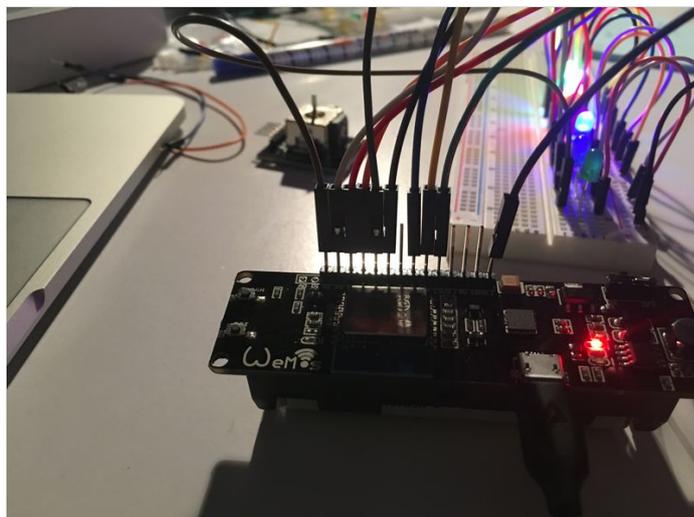
Afin de pouvoir faire rentrer toute la partie électronique dans le néon nous nous sommes tout d'abord portés sur l'ESP8266 avec un module fourni de chargement de batterie. Nous avons donc effectué de nombreux tests pour tester ce module (encodage en wifi avec MicroPython). Cependant, l'ESP8266 ayant seulement un seul port analogue nous avons acheté les mêmes modules sur Aliexpress avec des ESP32. Après 6 semaines d'attente, les ESP32 sont finalement arrivés. Nous avons alors tenté de les encoder avec des simples blinks en code arduino en passant par l'IDE Arduino via un câble FTDI.

La difficulté rencontrée ici était de découvrir que l'ESP32 doit subir une manipulation afin de le passer en mode "programmable" pour téléverser le code. Cette dernière en appuyant au moment du téléversement sur le bouton Reset pendant quelques secondes et appuyer sur Boot par la suite en même temps puis relâcher ces boutons. Si la procédure s'est bien déroulée l'IDE Arduino commence à afficher un processus de téléchargement avec des points et des pourcentages de téléversement.

Par la suite, nous avons trouvé le moyen de téléverser via le port micro-usb de la board achetée : il suffit d'installer un driver afin de pouvoir voir le port de l'ESP32. Ce driver est le CP2102 :

<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>.

Cependant après avoir fait marcher notre ESP32 sur secteur avec le code souhaité de InTime (quelques problèmes visuel de led qui restent légèrement allumées), nous nous sommes rendus compte que celui-ci ne s'exécutait pas sur batterie. Après plusieurs recherches, nous nous sommes en réalité rendu compte que les



boards achetées sont défectueuses. En effet la sortie 5V sur batterie est en réalité en 1.1 Volts ce qui est bien trop faible pour faire fonctionner notre système.

Nous nous sommes alors tournés vers une Arduino Nano pour réaliser ce projet. Le circuit est donc constitué d'une Arduino Nano, de deux bandeaux de LED WS2812B, deux capteurs de pouls pulse sensor, d'une batterie externe et de quelques câbles. Ce circuit est plutôt simple :

- Les +/- des bandeaux sont reliés entre eux et à l'Arduino en 5Volts. Les Pins sont respectivement reliés aux Pins 12 & 13 de l'Arduino.
- Les +/- des capteurs sont reliés entre eux et à l'Arduino en 3.3 Volts. Les Pins sont respectivement reliés aux Pins A0 & A1 de l'Arduino.
- L'Arduino est alimentée par les pins Vin et GND par une batterie externe via un câble USB dénudé.
- La batterie externe est rechargée par le biais d'un câble micro-usb : un câble micro-usb à l'intérieur du dispositif est relié à la batterie et à un port micro-usb femelle soudé et fixé dans le bouchon.
- Un interrupteur permet d'éteindre l'alimentation de l'Arduino par la batterie externe : un fil est connecté au + du câble usb et au Vin de l'Arduino

Pour les capteurs de pouls nous avons eu de nombreux problèmes. Tout d'abord les délais de livraison étant très longs nous avons décidé d'effectuer des tests sur ce type de capteurs en achetant un au magasin LetMeKnow de Paris en attendant la livraison des autres capteurs.

Les valeurs obtenues à partir de ces derniers n'étant pas fiables du tout. Nous avons notamment utilisé de nombreux codes Arduino de différents projets renseignés sur divers sites internet (Instructables et autres). Après avoir passé des heures sur ces capteurs nous nous sommes retournés voir le magasin LetMeKnow afin d'échanger sur ces problèmes et ces problèmes nous ont bien été confirmé par eux également.

Ainsi après avoir passé des semaines sur ces problèmes de capteurs nous avons décidé de pas prendre en compte leurs valeurs et nous nous sommes tournés vers un mode démo du dispositif dans le but d'une présentation de projet.

Step 4 - Code

Le code entier est disponible dans le fichier et est assez simple à comprendre.

Les capteurs de pouls ne donnant pas des valeurs fiables nous avons décidé de programmer un mode démo qui permet de montrer le processus imaginé de l'utilisation.

Step 5 - Design de l'objet

Step 6 -
