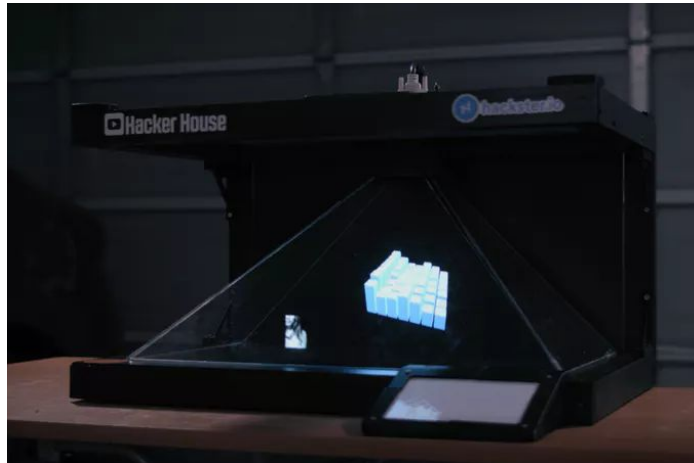


Documentation



Pour la réalisation de ce projet nous nous sommes entièrement basés sur un tutoriel disponible à l'adresse suivante:

<https://www.hackster.io/hackerhouse/holographic-audio-visualizer-with-motion-control-e72fee>

L'ensemble des documents, fichiers d'impression 3d ou code sont disponibles sur le github suivant:

<https://github.com/squaregolab/holoseius>

Le projet a été réalisé par Valentin Breton et Pierre Brosa, si vous reprenez ce projet, vous pouvez nous contacter par mail: valentin.breton@imerir.com pierre.brosa@imerir.com

Liste de matériel :

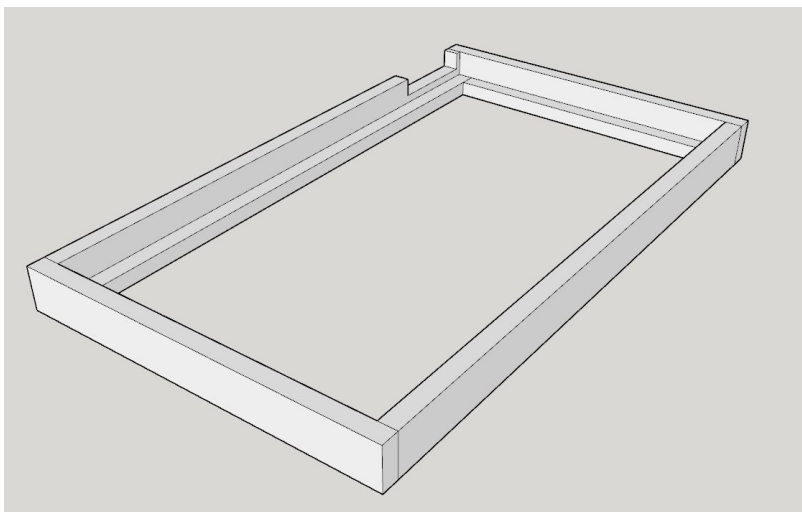
- *Raspberry pi 3 (x 1)*
- *Flick Large (x 1)*
- *Ecran 24" (x 1)*
- *Plexiglas 2mm épaisseur (dépend de la taille de l'écran)*
- *Equerre imprimées en 3D (x 4)*
- *Support des plaques plexiglass imprimées en 3D (x 1)*
- *Support du Flick Large imprimé en 3D (x 1)*
- *Planches de bois pour faire la base de la structure dépendant de la taille de l'écran (x 4)*
- *Grande planche permettant de faire le fond de la structure dépendant de la taille de l'écran (x 1)*
- *Planches de bois pour créer le support de l'écran en haut de la structure dépendant de la taille de l'écran (x 8)*
- *PC sous linux ou windows*
- *Vis*
- *Bombe en spray noir*
- *Stickers noir*

Construction de la structure:

Commencez par mesurer les dimensions exactes de l'écran que vous allez utiliser.

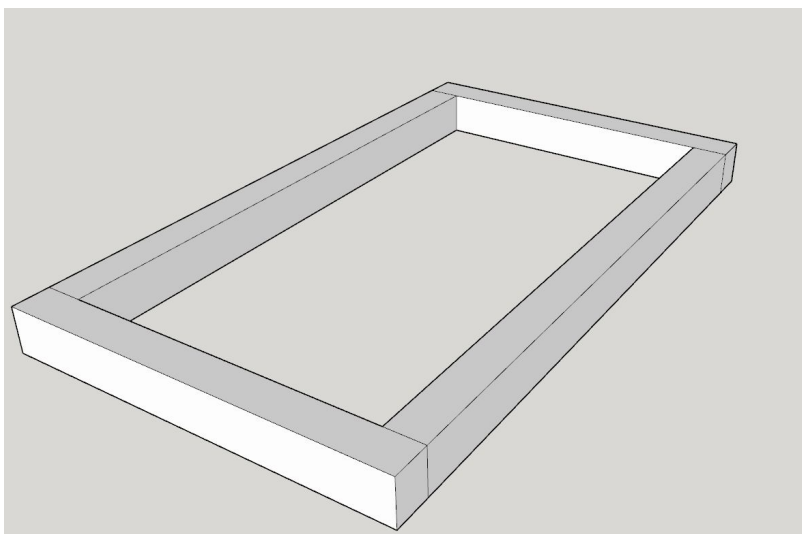
Ici nous utilisons un écran 24 Pouces, ATTENTION les mesures seront déterminantes !

Faites en premier le support pour l'écran:



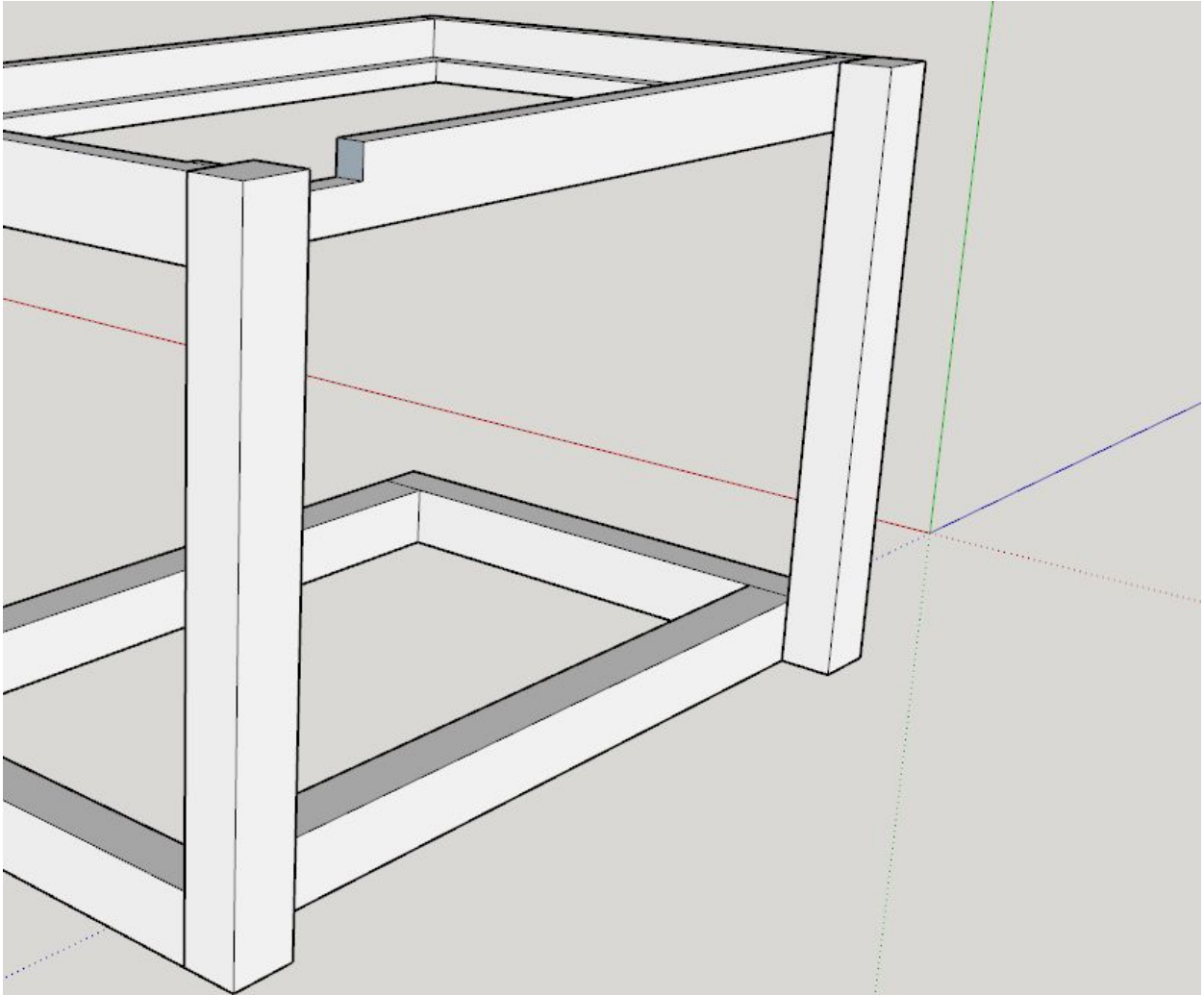
Votre écran doit s'insérer parfaitement dans le cadre, l'idéal étant qu'il n'y est pas de jeu. L'encoche sert uniquement à accéder facilement aux boutons de l'écran.

Ensuite la base:

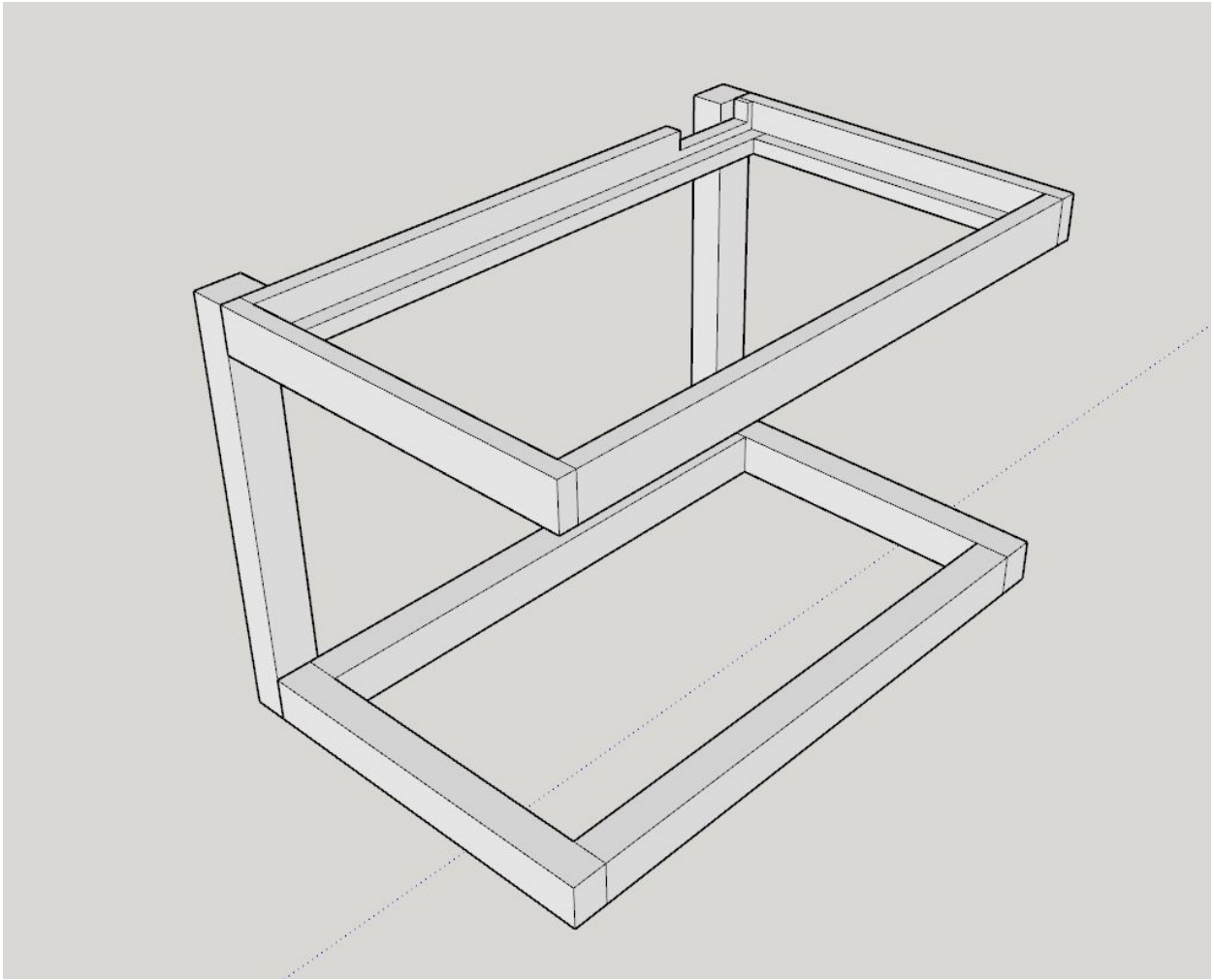


Prévoyez une base plus ou moins solide suivant la masse de votre écran. La base est de même dimension que le cadre entourant l'écran.

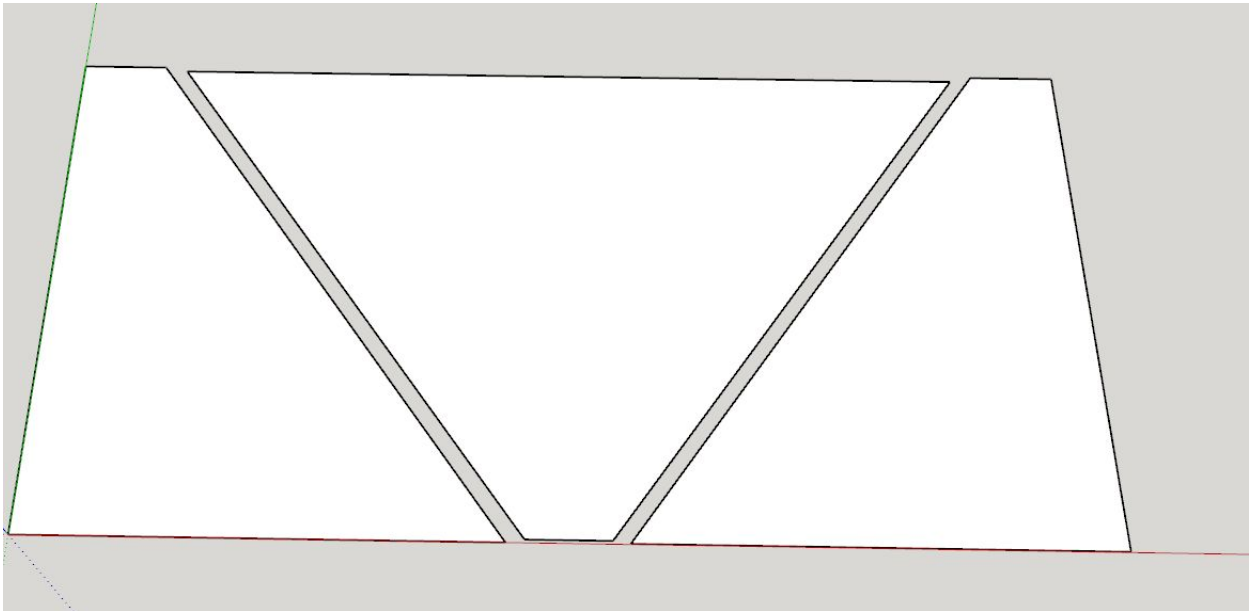
La hauteur:



Rendu final sur sketch up



Découpe des plaques de plexiglass :



Elles peuvent être coupé à la main (avec un cutter et de la patience) cependant, pour un rendu propre et rapide, nous l'avons fait avec une découpeuse laser (environ 2 min pour les 3 plaques).

Le schéma fait sur sketchup a été converti en fichier SVG puis à l'aide d'Inkscape et son plugin converti en .dkg.

Exécuter le code

Ce projet fonctionne sur deux appareils différents. Un Raspberry Pi contrôle la carte de reconnaissance de gestes Flick qui envoie des commandes au visualiseur fonctionnant sur un ordinateur Mac / Windows / Linux plus puissant prenant en charge WebGL pour exécuter les graphiques.

Pour commencer, assurez-vous de télécharger ce référentiel sur l'ordinateur sur lequel vous souhaitez exécuter le visualiseur et le Raspberry Pi.

Visualiseur

Le visualiseur doit fonctionner sur un ordinateur normal, car les animations du visualiseur 3D sont décalées sur le Raspberry Pi. Les animations du visualiseur sont exécutées dans un environnement Electron et utilisent Three.js (une bibliothèque WebGL) pour les graphiques 3D. Un serveur fonctionnant avec Node.js permet la communication avec le tableau gestuel de la main.

Commencez par installer la dernière version stable de Node.js sur votre ordinateur. J'ai utilisé la version 6.x. Ensuite, accédez au répertoire du référentiel et exécutez `sudo npm install` pour installer les dépendances pour le projet.

Démarrer le visualiseur avec: `npm start`

La liste de lecture Soundcloud peut être modifiée en modifiant la liste de lecture `const = 'chemin / vers / liste de lecture'`; `const` dans le fichier `render.js`. Le chemin peut être trouvé en extrayant la fin de l'URL pour une playlist Soundcloud. Par exemple, si l'URL d'une playlist est:

`https://soundcloud.com/someartist/sets/aplaylist`

Le `const` serait:

```
const playlist = 'someartist / sets / aplaylist';
```

Contrôles

Tout d'abord, assurez-vous que les dépendances nécessaires sont installées sur votre Raspberry Pi pour le tableau de mouvements Flick. Commencez par exécuter la commande suivante dans votre terminal:

```
curl -sSL https://pisupp.ly/flickcode | sudo bash
```

Assurez-vous de redémarrer votre Pi, puis exécutez la commande suivante dans le terminal pour tester votre tableau Flick. Vous devez vous assurer qu'il est correctement branché avant de procéder au test.

flick-demo

La partie contrôles de ce projet s'exécute sur un Raspberry Pi avec un programme python. Vous devez vous assurer que votre Raspberry Pi utilise python 2.7 pour que le programme fonctionne correctement. Vous pouvez vérifier la version de python en cours d'exécution sur votre système en exécutant:

```
python --version
```

Accédez au dossier des contrôles dans le répertoire du référentiel et exécutez la commande suivante pour installer toutes les dépendances:

```
sudo pip install -r requirements.txt
```

Avant d'exécuter le programme de contrôle, modifiez le fichier `swipe-controller.py` en remplaçant la partie `<HOST_IP>` de la variable `host = 'http:// <HOST_IP>: 3000'` par l'adresse IP locale de l'ordinateur exécutant le visualiseur.

Remarque: les deux ordinateurs doivent être sur le même réseau. Vous pouvez trouver l'adresse IP de votre ordinateur en tapant `ifconfig` (`ipconfig` / `all on windows`) dans le terminal et en recherchant l'adresse `inet`.

Exécutez le programme avec:

```
python swipe-controller.py
```

Une fois que les deux programmes sont en cours d'exécution, vous pouvez le tester!