

Fiche descriptive de projet Hub

Wendling Johan

Contexte et but du projet

Ce projet a pour but de créer un système permettant de piloter un ordinateur via une télécommande infrarouge standard. Le signal reçu est interprété pour déclencher des actions automatiques (simuler des touches clavier, mouvements de souris, clics, ou ouverture de sites web). Ce projet peut s'inscrire dans une logique d'accessibilité, d'automatisation, ou de démonstration technique. Ce projet a aussi pour but d'être totalement modulaire selon la télécommande utilisée.

Résumé du projet

- Recherche et sélection des composants matériels (Arduino, récepteur IR)
- Installation et configuration de l'environnement Arduino sous Linux
- Lecture des signaux infrarouges via un récepteur IR connecté à l'Arduino
- Transmission des signaux IR au PC via le port série USB
- Écriture d'un script Python (ou C++) sur le PC pour lire le port série (Python est plus simple pour plus de modularité et peut comprendre windows + linux grâce à certaines lib python)
- Traduction des signaux IR en actions utilisateur (ex. simuler Ctrl+W, espace, clic souris)
- Tests et calibrages avec différentes télécommandes
- Intégration et démonstration multi-plateformes (Linux & Windows)
- impression 3d du boîtier

Porteur(s) du projet

- **Wendling Johan** – Epitech Strasbourg – 3e année – Développement Arduino, script Python, tests matériels, Gestion des signaux IR, configuration Linux, intégration logicielle, Interface graphique + config multiplateforme et impression 3d

Environnement technique / technologique

Matériel :

- Carte Arduino UNO (ATmega328P)

- Récepteur infrarouge 3 broches (G, R, Y)
- Télécommande infrarouge
- Câbles Dupont, breadboard

Langages :

- C++ (code Arduino)
- Python 3 (lecture série + actions clavier/souris avec PyAutoGUI)

Environnement d'exécution :

- PC sous Fedora Linux (testé également sous Windows)
- Arduino IDE 2.x
- Terminal série
- Python avec bibliothèques : `pyserial`, `pyautogui`, `mouseinfo`

Ressources :

- Librairie IRremote Arduino
- Documentation officielle Arduino
- Stack Overflow, forums Arduino et PyAutoGUI
- Google / youtube et chat (pour apprendre à utiliser arduino sur les boards spécifiques et pour voir comment souder sans se tromper)

Description du livrable

- **Code Arduino** pour lire les signaux IR et les transmettre via USB
- **Script Python** interprétant les codes IR et déclenchant des actions
- **Interface graphique** permettant à l'utilisateur de configurer les actions associées aux codes IR
- **Boîtier imprimé en 3D** pour accueillir proprement le matériel
- **Documentation technique et utilisateur** (README, tutoriel, schémas de câblage, STL du boîtier)
- **Niveau de finition** : Fonctionnel, en cours de stabilisation, documentation en cours, interface

Organisation et temporalité

1 – Lecture des signaux IR sur Arduino

Explication : Connexion du récepteur, test avec librairie IRremote, affichage dans le moniteur série

Temps estimé : 6h

2 – Développement du script Python de traitement

Explication : Lire le port série, analyser les codes reçus, déclencher actions via PyAutoGUI

Temps estimé : 10h

3 – Interface graphique de configuration

Explication : GUI pour associer des codes IR à des actions, sauvegarde dans un fichier de config

Temps estimé : 8h

4 – Tests & Intégration multi-plateformes

Explication : Tester sous Linux et Windows, corriger les erreurs spécifiques à chaque OS

Temps estimé : 5h

5 – Conception et impression 3D d'un boîtier

Explication : Modéliser le boîtier dans un logiciel , ajuster les dimensions, imprimer en PLA

Temps estimé : 6h

6 – Documentation & mise au propre du projet

Explication : Rédaction de README, capture d'écran, schémas de câblage, instructions utilisateur

Temps estimé : 5h

Estimation du projet

Total estimé : environ 7 jours

