



SCENOSCOPE

Gitlab – Guide et explications

Table des matières

Introduction	3
Boitier	4
Data Logging Arduino Shield.....	4
IMU10DOF	4
Journée AGDE 1.....	5
PCB.....	5
PROTO_FINAL.....	5
PYTHON_V2	6
Prototypes + Démo initiale + sans IMU	6
Python Directory	6
Rapports PFE Aline et Hugo	6
Shield Sparkfun – Lecteur RFID.....	7
Test_IES	7
Contacts.....	7

Introduction

Nous avons créé le gitlab « scenoscope » dans le cadre d'un stage à l'Institut d'Electronique (IES), réalisé lors de notre première année de cycle ingénieur, spécialité Microélectronique et Automatique. Ce stage s'inscrit dans le programme de recherche « Scenoscope » dirigé par le laboratoire l'Humain, en collaboration directe avec l'IES.

Le projet Scenoscope vise à étudier l'expérience culturelle des visiteurs dans les musées ; et d'en déduire plusieurs choses quant à la scénographie ou encore les différents comportements des visiteurs devant certaines œuvres.

Notre rôle à nous était de réaliser, à partir de premiers prototypes déjà réalisés par Arnaud Vena, notre maître de stage ainsi que Aline Lefebvre et Hugo Marquette, deux élèves ayant travaillé sur le projet lors de leur PFE, un dispositif de localisation indoor dans les musées. Ce dispositif fonctionne avec la technologie RFID : le but est de coller des tags à des endroits stratégiques dans un musée. Les visiteurs se promènent avec une boîte dans laquelle nous avons intégré le circuit électronique permettant de lire les tags, de déduire la position du visiteur dans le musée, sa vitesse, son inclinaison et le temps passé devant chaque œuvre.

Une fois ces données récupérées, nous avons à les traiter et à retracer le trajet du visiteur dans une application.

Nous avons eu l'occasion de tester notre dispositif lors d'une journée dans le Musée de l'Ephèbe à Agde, et cela nous a donné des premiers résultats très encourageants pour la suite du projet.

Ce Gitlab que nous avons créé regroupe tous les documents et programmes que nous avons utilisés/créés/modifiés et nous allons dans ce guide vous détailler son contenu.

Ce document traitera chaque dossier dans l'ordre alphabétique, ordre dans lequel ils apparaissent sur le Gitlab.

Boitier

Ce dossier regroupe tous les fichiers créés sous FreeCad pour réaliser le boîtier qui contient le circuit électronique.

- Dossier « FINAL » : il regroupe tous les documents relatifs à la modélisation 3D de la boîte.
- « Plans_Boitier.pdf » est la photo des esquisses préliminaires réalisées par Victor avant de se lancer dans la modélisation sur logiciel.

Data Logging Arduino Shield

Ce dossier contient deux fichiers :

- « Lien Adafruit overview » qui contient seulement le lien du site sur lequel nous avons trouvé toute la documentation du shield. Ce shield permet de sauvegarder les données dans la carte SD et d'horodater les données grâce au RTC intégré.
- « Test_sd_RTC.ino » qui est le programme arduino qui nous a permis de tester le shield avant de l'incorporer dans le programme principal.

IMU10DOF

L'IMU10DOF est un composant nous permettant de récupérer l'accélération, la magnitude et l'inclinaison de l'utilisateur du boîtier. Dans ce dossier on trouve :

- « Sseed_Arduino_IMU10DOF-master.zip » : Ce dossier .zip est la librairie utilisée dans le programme arduino pour faire fonctionner l'IMU.
- « Ido_imu_10 » : Ce document est une image qui montre les branchements de l'IMU sur la carte Arduino
- « testIMU.ino » : ce fichier est le programme arduino qui permet de tester l'IMU avant de l'incorporer dans le programme principal.

Journée AGDE 1

Ce dossier regroupe tous les documents qui résultent de notre journée de test à Agde.

- « PlacementTagsMusée.png » : Cette image présente la position des tags avec des points rouges sur le plan de l'exposition. Ceci est un screen de l'application créée sur Python.
- « PosedeTags-MANUSCRIT/pdf » est la photo de la feuille sur laquelle nous avons travaillé lors de la pose des tags. On y voit la position des tags et leur identifiant en décimal.
- « PosedeTagsMuseeAgde.pdf » et PosedeTagesMuseeAgde.xls » (fichier excel) sont le tableau contenant l'identifiant du tag en décimal, la position de chaque tag en x et en y, son identifiant en hexadécimal et le placement dans le musée (nom de la salle). Ce fichier doit être dans le même dossier que le programme python.
- « Plan_expo.png » est la photo du plan de l'exposition que l'on utilise dans le programme python.
- « Test_musee_Ephebe.xls » est le fichier excel résultant de l'expérience au musée : on y retrouve toutes les informations : date et heure de la détection du tag, numérotation du tag, données de l'IMU, EPC, RSSI. Ce fichier doit également être dans le même dossier que le programme python lors de son exécution.
- « withscaler.py » est la version de l'application python avec les bons documents du premier test au musée d'Agde. Cette version a comme option supplémentaire de tracer le trajet grâce à un scaler.

PCB

Ici on retrouve seulement le dossier compressé « tic_tac.rar » qui regroupe tous les documents permettant d'imprimer le PCB reliant batterie, chargeur, interrupteur. Le schéma électrique et le PCB ont été réalisés sur KiCad.

PROTO_FINAL

On a, dans ce dossier :

- « Proto5-RIOS-MULLER.ino » : Avant dernière version du programme Arduino qui récupère l'EPC entier en hexadécimal.
- « ProtoFINAL-RIOS-MULLER.ino » : Dernière version en date du programme Arduino qui récupère le nombre de tags et leur EPC en décimal (facilite le traitement des données avec le programme Python).
- « SparkFun_UHF_RFID_Reader.cpp » et « SparkFun_UHF_RFID_Reader.h » : bibliothèque modifiée du lecteur RFID pour régler le problème de détection de plusieurs tags simultanément : il faut ABSOLUMENT copier ces deux fichiers dans le même dossier que le programme Arduino pour assurer le bon fonctionnement du dispositif.

PYTHON_V2

Dans ce dossier se trouve :

- « Proto_V2.py » et « Test_croix.py » qui sont les programmes permettant de tracer le trajet du visiteur. La dernière version du code et celle à utiliser est « withscaler » figurant dans le même dossier, elle a comme option supplémentaire de tracer le trajet à la vitesse que veut l'utilisateur, car celui-ci peut « tirer » une barre de défilement qui au fur et à mesure que l'on « avance » dans la barre, les positions s'affichent.
- « sceno_home.ico » est l'icône de l'application, il faut qu'elle soit dans le même dossier que le .py pour que tout fonctionne comme voulu.

Prototypes + Démo initiale + sans IMU

Dans ce dossier, on a :

- « Prototypes.zip » qui contient toutes les versions du programme arduino avant d'arriver au prototype final.
- « demo2v1.ino » est le programme que Mr.Vena nous avait donné comme base, que nous avons ensuite modifié pour l'adapter à nos Shields.
- « essaisansIMU.ino » est le programme arduino qui fonctionnait sans l'IMU.

Python Directory

Dans ce dossier nous avons les premières versions du code Python, pour la version plus récente et fonctionnelle il faut se référer au dossier « PYTHON_V2 » :

- « IHM_Python_1.zip » est le dossier initial que Mr. Vena nous a fourni et que nous avons modifié par la suite
- « Test_croix2.py » est la version de l'application python qui permet seulement de placer les tags dans le plan et de rentrer leur EPC à chaque placement de tag.
- « plan_expo.png » est à nouveau la photo du plan de l'expo.

Rapports PFE Aline et Hugo

« RapportPFE_AlineLefevre.pdf » et « RapportPFE_AlineLefevre.pdf » sont les rapports respectifs de Aline et Hugo, étudiants ayant fait leur projet de fin d'étude sur le projet scenoscope. Eux avaient travaillé sur STM32 mais on a quand même pu bien s'aider de leur rapport pour comprendre les enjeux du projet.

Shield Sparkfun – Lecteur RFID

Dans ce dossier se trouve tout ce qui se rapporte aux tests du lecteur RFID seul :

- « Guide Shield Sparkfun - Lecteur RFID » est le lien du site qui nous a servi à prendre en main et à se documenter sur le lecteur Sparkfun.
- « SEN-14066_datasheet.pdf » : ce pdf est la datasheet du Shield Sparkfun.
- « Simultaneous_RFID_Tag_Reader.pdf » : ce document est le schéma électrique
- « Test_EPC.ino » est le programme Arduino qui permet de tester le Shield : ce programme active le buzzer dès que le lecteur détecte un tag.
- « singleReadingRSSI.zip » est la bibliothèque créée par Mr.Vena.

Test_IES

Dans ce dossier nous avons tous les documents se référant au test effectué dans le couloir de l'IES. En effet, nous avons fait un second test (après celui du musée) de notre dispositif dans l'IES, où nous avons posé des tags. Nous avons donc :

- « Proto_V3.py » qui est le programme python utilisé pour visualiser le trajet.
- « TAG_DATA.xls » est le fichier qui recense les identifiants des tags et leur position en x et y associée. Ce fichier doit être dans le même dossier que le programme python sinon ce dernier ne fonctionnera pas.
- « Test_IES.xls » est le fichier excel dans lesquelles on retrouve les données récupérées de la « visite » du couloir de l'IES. Ce fichier doit aussi se trouver dans le même dossier que le programme python.
- « plan.png » est le plan du couloir, qu'il faut aussi placer dans le même dossier que le programme lors de son exécution.

Contacts

Pour plus d'informations concernant ce GitLab ou ce projet, contactez-nous à une de ces deux adresses mail :

- maelynn.muller@etu.umontpellier.fr
- victor.rios@etu.umontpellier.fr