

# Application du C++ au domaine des objets connectés

Année 2022-2023

## 1. Objectifs

Ce Bureau d'étude a plusieurs objectifs :

- Réfléchir à une conception orientée objet
- Utiliser le langage C++
- Développer une bibliothèque modulaire, extensible pour s'interfacer avec des capteurs et des actionneurs
- Imaginer un problème à résoudre ou un nouveau service que vous pouvez créer dans le monde des objets connectés et de l'Internet des Objets
- Développer une application attractive pour utiliser votre bibliothèque

Vous allez utiliser un microcontrôleur ESP8266 programmable en arduino. Vous pourrez ajouter des capteurs et des actionneurs. La carte ESP8266 embarque aussi en natif une carte WIFI. Néanmoins comme ces cartes ne sont pas déclarées sur le réseau IOT de l'INSA, il vous faudra créer un réseau WIFI avec votre smartphone ou votre ordinateur. Si toutefois la connexion sur le réseau de l'INSA est nécessaire, nous pourrions faire une demande pour déclarer l'adresse mac de votre carte sur le réseau IoT.

Vous allez travailler à 2 sur le projet, vous mettrez vos sources dans un dépôt git accessible afin de pouvoir travailler en parallèle aisément. Cela permettra aussi à votre enseignant de BE de pouvoir accéder à votre code.

## 2. Déroulement

Ce BE se déroule sur 5 séances de 2h45 en binôme (1h est laissé pour faire le mooc java si nécessaire). La dernière séance de BE est dédiée à des QCM d'évaluation (C++ et java) et la démonstration et le rendu de votre BE aux enseignants de TP. Vous devrez fournir :

- L'ensemble de vos fichiers source dans une archive
- Un rapport décrivant votre conception (cas d'utilisation, scénarios, diagramme de classe, diagramme de séquence, etc), le déroulement du projet, comment faire marcher votre système, toute analyse intéressante sur un maximum de 3 pages.
- Un fichier README expliquant comment compiler votre programme, comment l'exécuter et comment interagir avec lui pour le faire réagir

Cette année, le déroulé des séances est particulièrement serré. Il vous faudra donc en priorité prendre des capteurs et actionneurs déjà disponibles sur l'INSA.

Projet fonctionnant	4
Respect des principes de l'objet	2
Respect du cahier des charges en terme d'utilisation du C++ : <ul style="list-style-type: none"><li>• Création de plusieurs classes</li><li>• Utilisation du mécanisme d'héritage</li><li>• Redéfinition d'opérateur</li><li>• Utilisation de la STL</li><li>• Utilisation des exceptions</li></ul>	1 1 1 1 1
Programme propre et commenté	2

Qualité et quantité du projet	2
Idée intéressante en terme d'application	2
Rapport clair et intéressant	3
Non respect des échéances	-5

### 3. Contenu des séances

Ce découpage est indicatif vous pouvez prendre de l'avance.

#### - Première séance

Cette séance sera dédiée à la prise en main du simulateur.

##### 1. Création d'un espace de partage

- Si vous n'avez pas déjà un compte sur github, créez en un par binôme (github.com)
- Définissez un nouveau projet (ce sera plus simple pour les enseignants ensuite pour suivre vos projets). Cette espace sera le point central de votre projet.
- Partager le lien sur github de votre projet avec votre enseignant de BE
- Suivez un tutoriel rapide sur internet si vous n'êtes pas déjà familier avec git, par exemple : <https://www.hostinger.fr/tutoriels/tuto-git/>
- Cloner le projet sur votre machine et commencer à travailler, en parallèle avec votre binôme, chacun sur sa machine
- Pensez à faire des push pour synchroniser le travail du binôme et notamment pour donner une vue de votre projet à vos enseignants.

##### 2. Description de la carte ESP8266 et installation de l'environnement de développement

- Recherchez des informations sur les capacités de votre carte en terme de puissance, mémoire, entrées, sorties, etc.
- Installer l'IDE d'arduino
- Configurer l'IDE pour la carte ESP8266 avec les téléchargements de modules supplémentaires  
(Utilisez les tutoriels disponibles sur le WEB)

##### 3. Utilisez la led interne de la carte

Cherchez dans les exemples de programme celui qui permet d'allumer la led qui se trouve sur la carte (build'in LED). Testez ainsi que votre configuration et votre carte fonctionnent correctement.

##### 4. Ajoutez le shield grove et connectez un capteur ou un actionneur grove

Utilisez un capteur/actionneur simple pour vérifier l'utilisation du shield grove. Vous pouvez commencer à tester d'autres capteurs/actionneurs qui pourraient vous servir pour votre BE. Attention au problème d'alimentation électrique de la carte (3,3V) et la compatibilité avec les capteurs/actionneurs.

#### 5. Tester le WIFI (éventuellement à faire en séance 2)

Créer un réseau WIFI avec votre smartphone ou ordinateur portable. Tester la connexion de l'ESP à ce réseau.

#### - Deuxième séance

L'objectif de cette séance est de mettre en place une bibliothèque en C++ pour manipuler l'ensemble des senseurs et actionneurs de manière aisée et standard. Réfléchissez aux attributs communs, aux informations complémentaires intéressantes, aux méthodes communes et différentes. Ceci doit vous amener à proposer une architecture de classes avec des classes héritées.

Dans le rapport, vous devrez justifier ce choix et commentez l'arborescence de classe que vous avez créé. Écrivez et testez votre code.

#### - Troisième séance

Dans cette séance, vous allez finir de coder votre bibliothèque et commencez à développer à une application à même de tirer parti des senseurs et actionneurs.

1. Vous définirez les uses case montrant comment votre application va fonctionner, vous pouvez fournir quelques scénarios.
2. Réfléchissez ensuite aux principales classes, attributs et méthodes nécessaires pour construire votre application. Ces 2 points alimenteront aussi votre rapport.
3. Validez votre approche avec l'encadrant de TP
4. Commencez à coder votre application

#### - Quatrième séance

Finalisez votre application et votre rapport. Ce rapport de 3 pages maximum doit contenir notamment une petite introduction positionnant la problématique, un diagramme de classe, le schéma de fonctionnement matériel et logiciel, une conclusion (But atteint, problème rencontrés, perspective d'évolution, etc). Partager l'ensemble sur github et prévenez votre enseignant que le projet est dans sa version finale.

**Soyez inventif , mais pas trop ambitieux vu la configuration du BE cette année. Travaillez en parallèle dans le binôme et de manière incrémentale sur vos prototypes pour être sûr d'avoir quelque chose à montrer à la fin.**

#### 4. Quelques ressources :

[arduino.cc](https://arduino.cc)

[github.com/esp8266/Arduino](https://github.com/esp8266/Arduino)