

# Projet 8 - IMA4 : Simulateur de Processus Physiques

Par Corto Callerisa, Sébastien Dardenne, Alex Lagneau, Quentin Normand



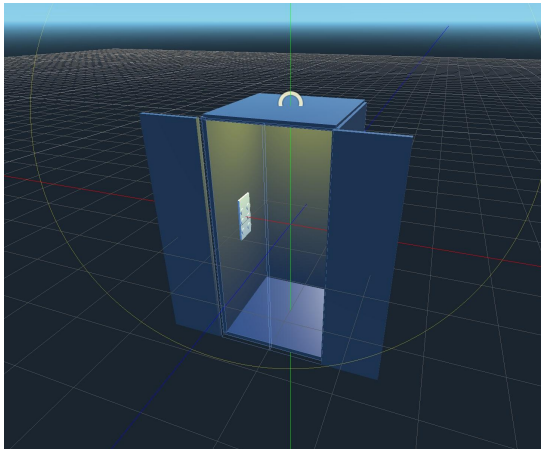
Tuteurs : Blaise Conrard / Xavier Redon

# Sommaire

1.	Projet S6 .....	3
2.		
3.	Projet S7 .....	4
4.		
5.	Objectif S8 .....	5
6.		
7.	I. Simulation : Équipe 1 .....	6
	• <i>Quentin Normand, Alex Lagneau</i>	
5.	II. Communication : Équipe 2 .....	18
	• <i>Corto Callerisa, Sébastien Dardenne</i>	
6.	Conclusion .....	24

# Projet S6

## Simulation de l'ascenseur



- Actionneurs : appel de fonction
- Capteurs : objet Raycast

## Communication UDP

```
(Ascenseur) record script.cmd  
(Ascenseur) open  
(Ascenseur) close  
(Ascenseur) playback script.cmd  
(Ascenseur) get vitesse  
(Ascenseur) etage 1  
(Ascenseur) set vitesse 20  
(Ascenseur) etage 0  
(Ascenseur) close  
(Ascenseur) quitter
```

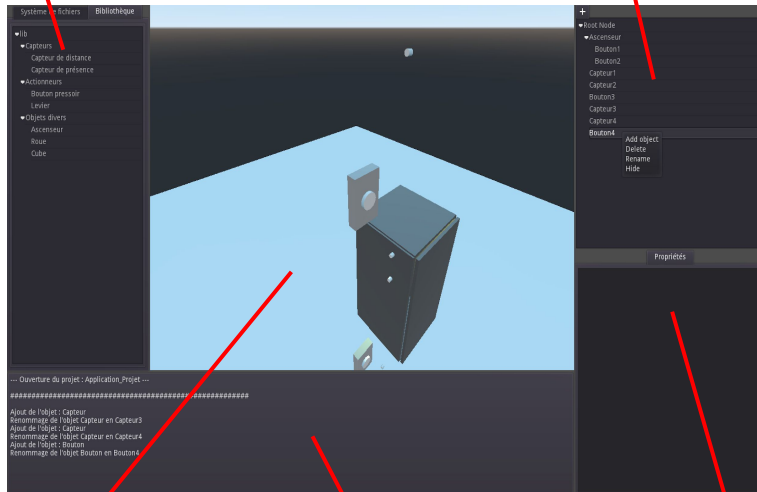
- Modifier / Récupérer une variable
- Effectuer une action
- Enregistrer / Rejouer une série de commande

# Projet S7

## Simulation

Bibliothèque d'objets

Arbre des objets



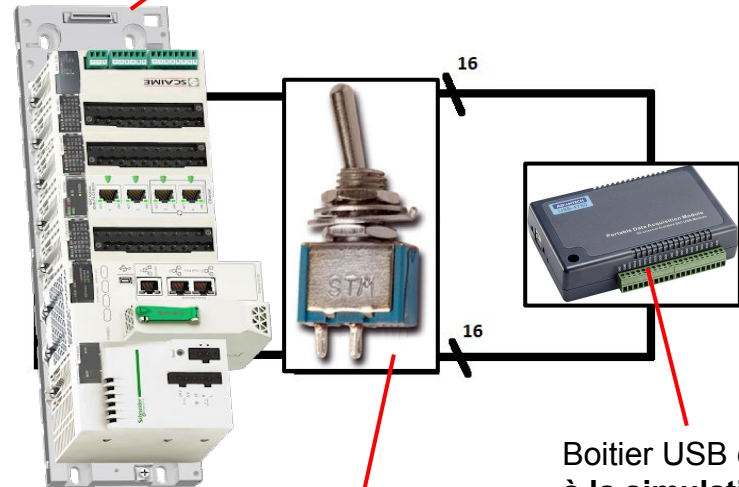
Scène en 3D

Affichage des modifications

Propriétés de l'objet sélectionné

## Communication

Automate connecté à UnityPro



Interrupteurs 3 états (x32)

Boîtier USB connecté à la simulation

# Objectifs S8

- Finaliser la communication entre l'automate et la simulation
- Créer des fichiers de sauvegarde pour les systèmes créés
- Améliorer le déplacement des objets à la souris
- Personnalisation des capteurs
- Développement d'un automate virtuel
- Fonctionnement de l'automate avec une simulation de l'ascenseur

# Partie I : Simulation

Equipe 1 : Alex Lagneau, Quentin Normand

# Simulation

Scène en 3D

Bibliothèque d'objets

Arbre des objets

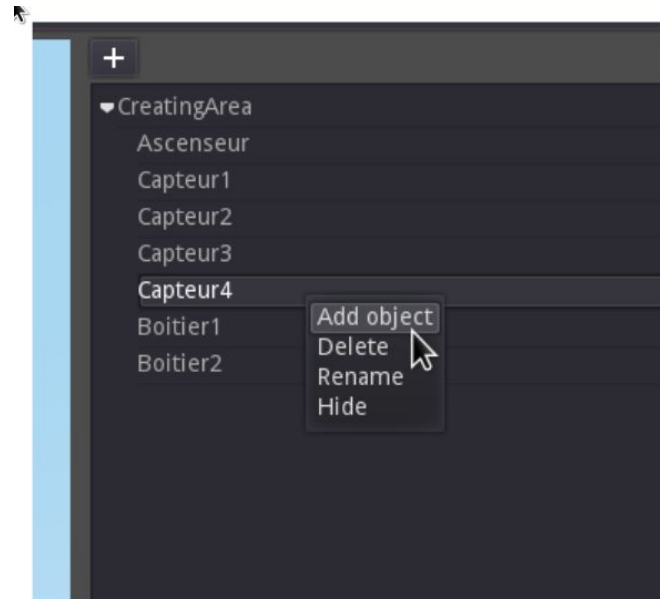
Affichage des modifications

Propriétés de l'objet sélectionné

The screenshot displays a 3D simulation environment. On the left is a 'Bibliothèque' (Library) panel with a tree view containing categories like 'Capteurs' (Sensors), 'Actionneurs' (Actuators), and 'Objets divers' (Miscellaneous objects). The central 'Scène en 3D' (3D Scene) shows a dark grey cabinet with a small square sensor on top. On the right is an 'Arbre des objets' (Object Tree) panel listing 'Ascenseur', 'Bouton1', 'Bouton2', 'Capteur1', 'Capteur2', 'Capteur3', 'Capteur4', and 'Bouton4'. Below the tree is a 'Propriétés' (Properties) panel with buttons for 'Add object', 'Delete', 'Rename', and 'Hide'. At the bottom, a console window shows a log of actions such as 'Ajout de l'objet - Capteur' and 'Renommage de l'objet Capteur en Capteur3'.

# Simulation

~~Drag & Drop~~ → Arbre des objets

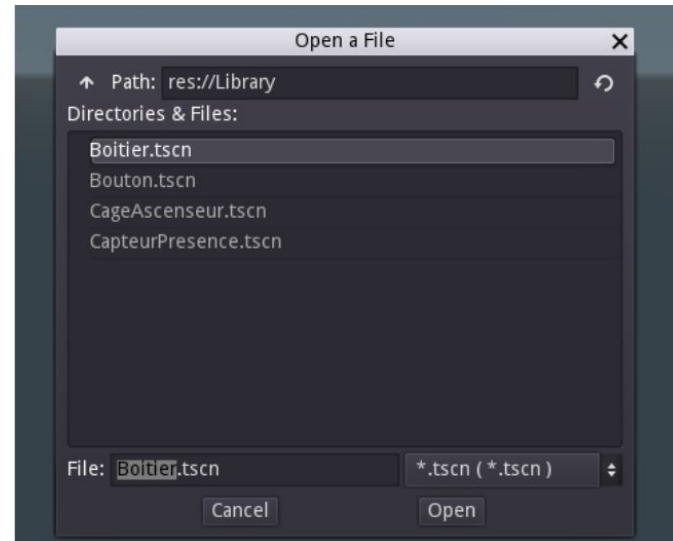
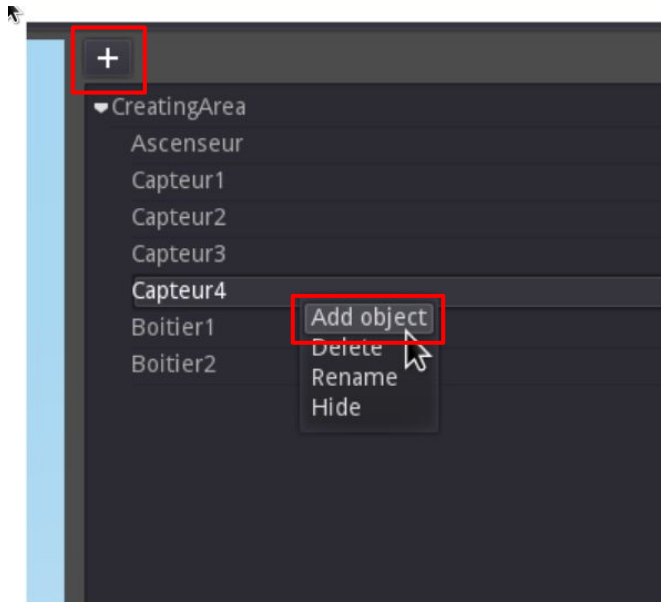




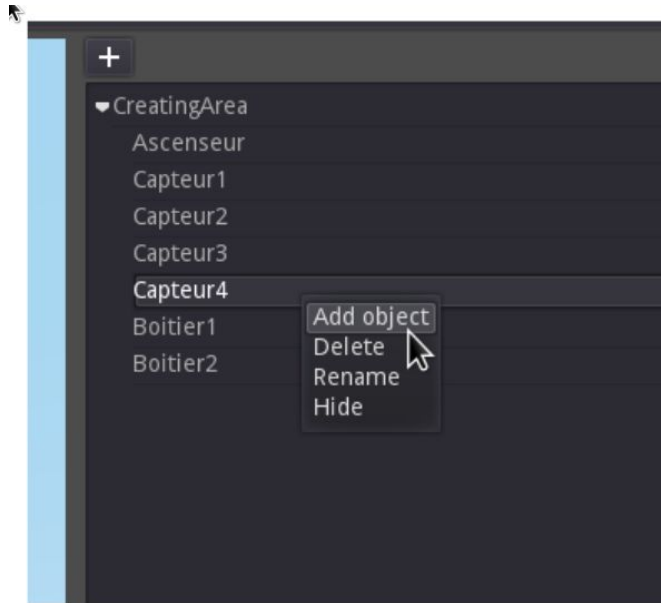
# Simulation

~~Drag & Drop~~

Arbre des objets



# Simulation

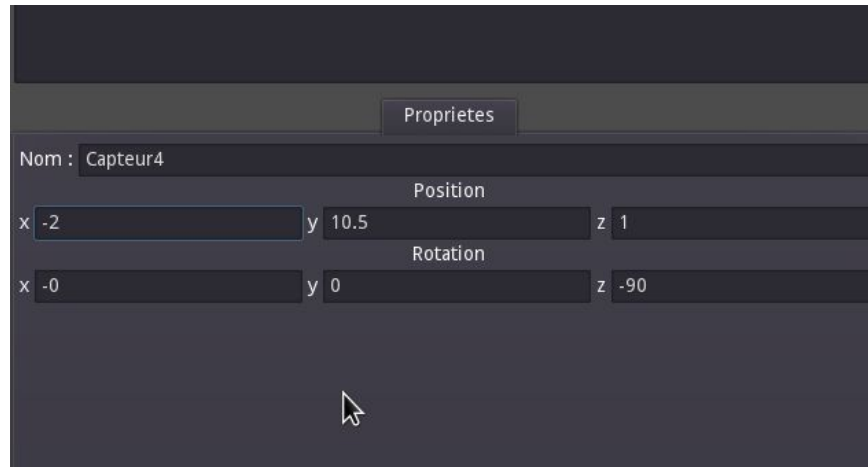


- Ajouter un objet ✓
- Supprimer un objet ✓
- Renommer un objet ✓
- Cacher un objet ✗

# Simulation

## Déplacement des objets :

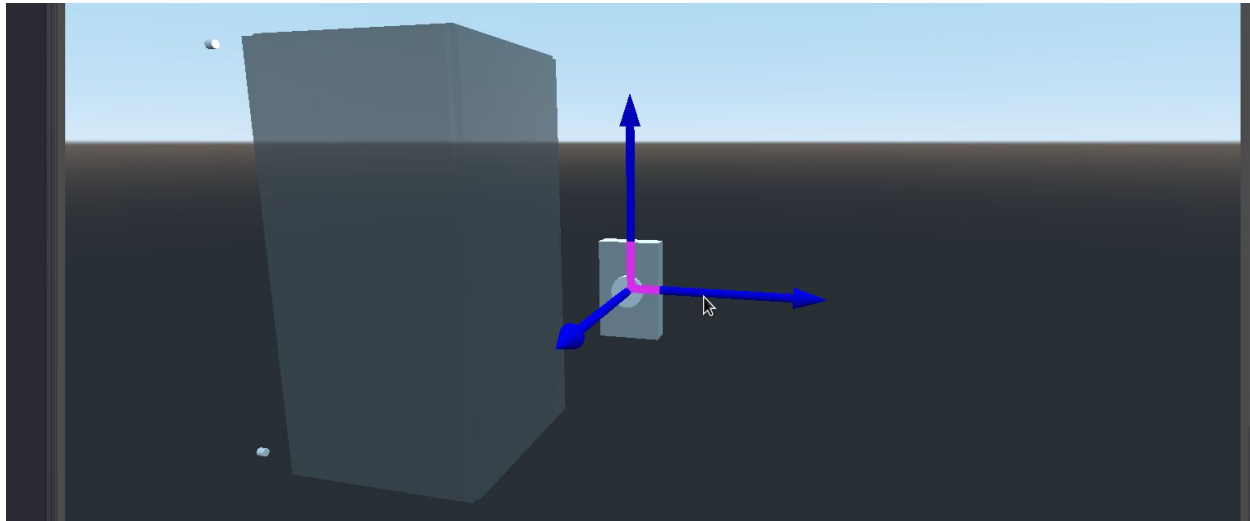
#1 : Utilisation de la fenêtre des propriétés



# Simulation

Déplacement des objets :

#2 : Déplacement grâce à la souris dans la scène 3D



# Simulation

## Déplacement des objets :

#1 : Utilisation de la fenêtre des propriétés

- Plus précis
- Rotation possible
- Moins visuel

#2 : Déplacement grâce à la souris dans la scène 3D

- Moins précis
- Pas de rotation
- Plus visuel
- Certains bugs

# Simulation

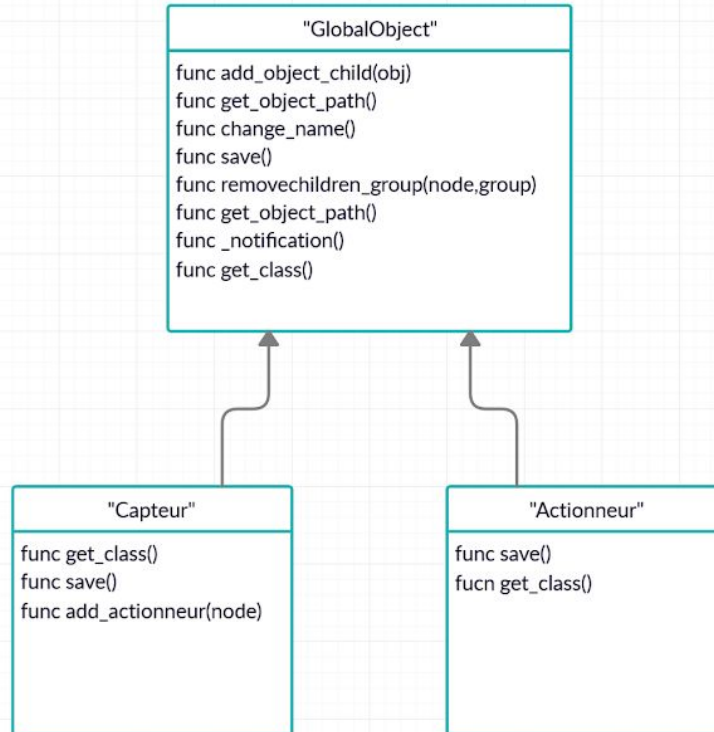
## Sauvegarde du projet :

func save\_project\_tree()

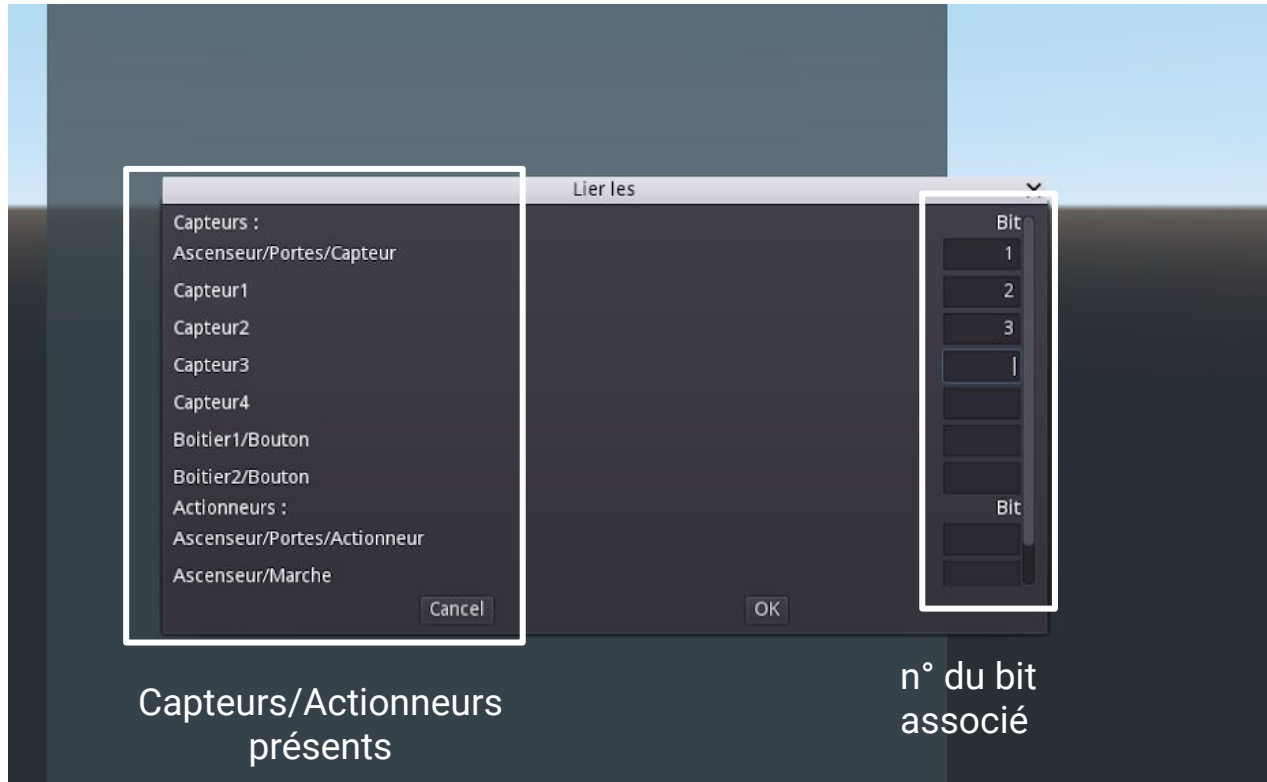
## Chargement du projet:

func open\_project\_file → func load\_project\_tree → func add\_node

# Simulation

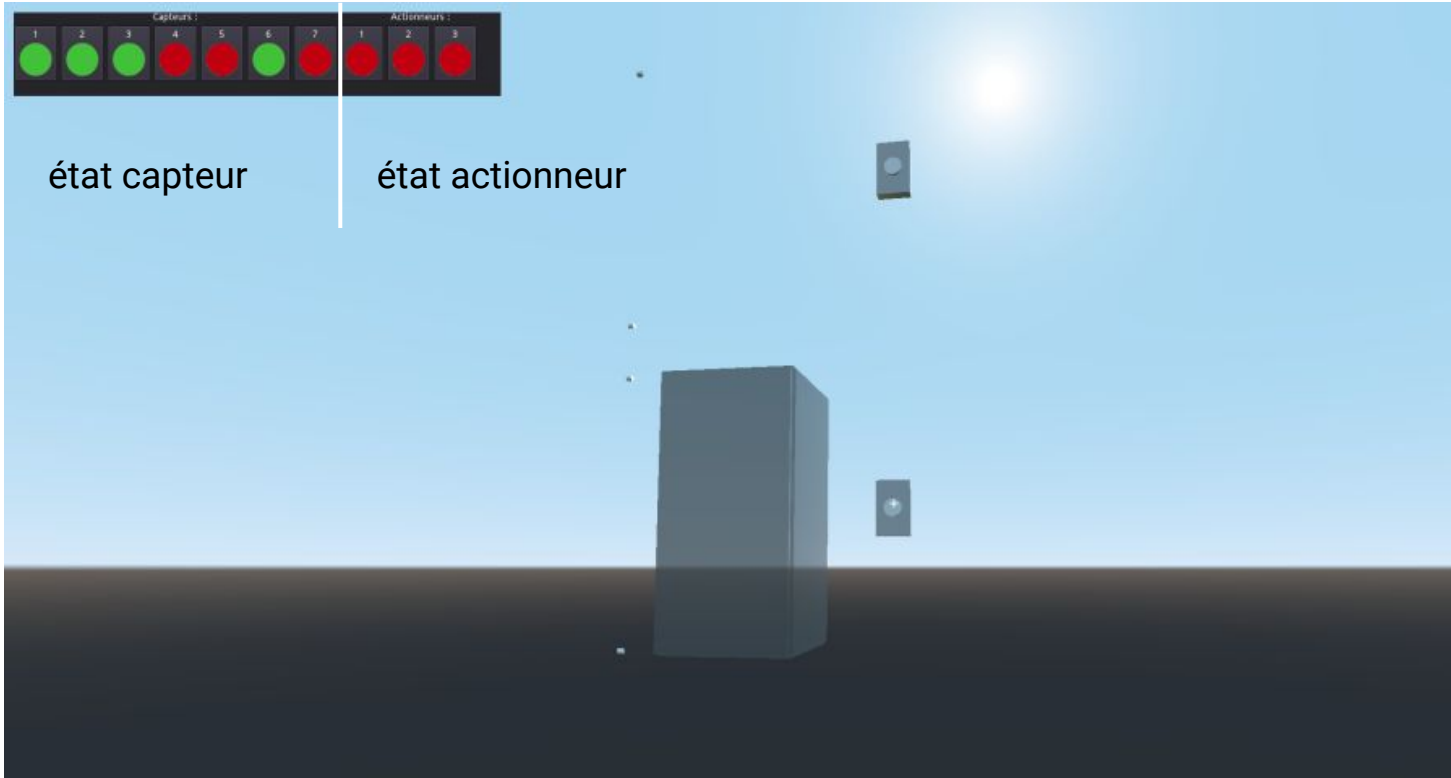


# Simulation





# Simulation



# Partie II : Communication

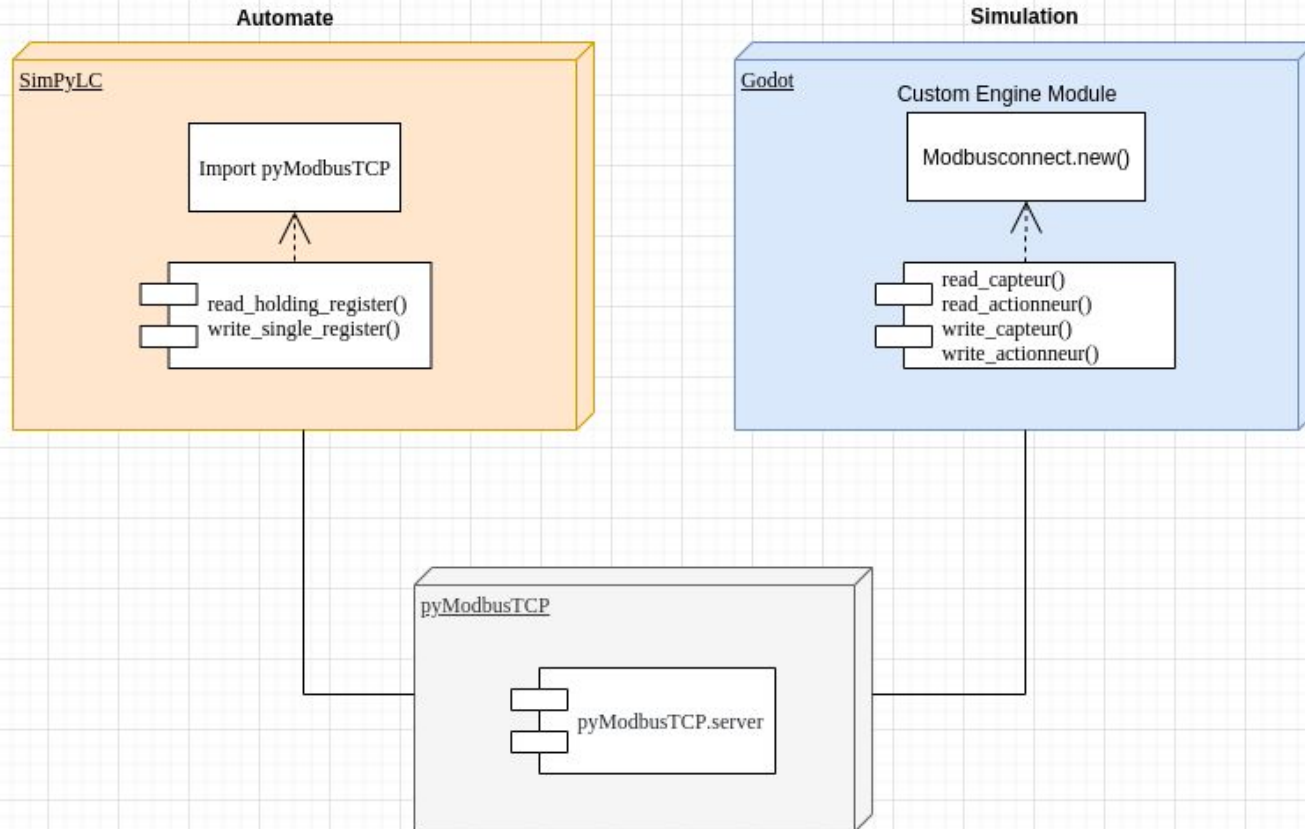
Equipe 2 : Corto Callerisa, Sébastien Dardenne

# Communication

1. Rappel de la problématique
2. Statut pré-covid et post-covid
3. Démonstrations

# Vue Top-level de la Communication

## Architecture de la Communication



# Communication

Démonstration : <https://youtu.be/tlm8bXnCuTc>

The screenshot shows a YouTube video player displaying a simulation. The video content is divided into three main sections:

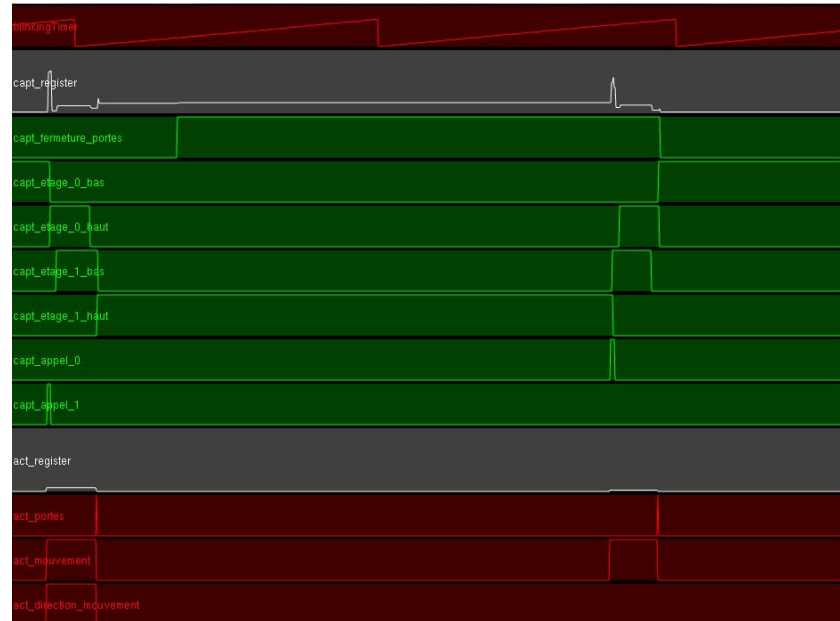
- Diagram:** A block diagram titled "Simulation" showing a "Custom Engine Module" containing "Modbusconnect.new()". Below it, a separate block contains functions: "read\_capteur()", "read\_actionneur()", "write\_capteur()", and "write\_actionneur()".
- Code:** A terminal window showing a series of "c.write\_single\_register(1, X)" commands, where X ranges from 0 to 11. Each command is followed by "True".
- Output:** A terminal window showing the output of the simulation, including "Output 00. mp4" and "Metadata" for a video stream. The output includes details like "encoder: Lavf58.29.100", "Stream #0:0: Video: H264 (libx264) (avc1 / 0x31537861), yuv444p, 960x1080, q=1-1, 30 fps, 1536 tbn, 30 tbc", and "Side data: cbp: bitrate max/min/avg: 0/0/0 buffer size: 0 vbv\_delay: -1".

# Affichage de l'automate

## Tableau de bord de l'automate

blinkingTimer	0.6634130000000198
pulse	0
counter	21
run	1
capt_register	2
act_register	0
capt_fermetureportes	0
capt_etage_0_bas	1
capt_etage_0_haut	0
capt_etage_1_bas	0
capt_etage_1_haut	0
capt_appel_0	0
capt_appel_1	0
actportes	0
actmouvement	0
actdirectionmouvement	0
appel_etage_0	0
appel_etage_1	0
porte_charge	0
porte_charge_bas	0
latch_intermediaire	0

## Suivi d'évolution des registres



# Conclusion

- 1. Réponse à l'objectif ?**
- 2. Limites de la solution**
- 3. Apports de ce projet**

# Merci de votre attention !

N'hésitez pas à nous poser des questions sur les obstacles rencontrés lors du développement :  
Il y a de la matière !