



# LiveCoding

Julien, Davide, Bastien

# Sommaire



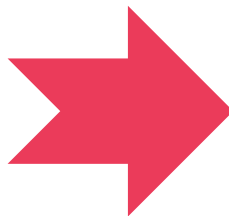
- Structure du projet
- Client
- Serveur
- Jeux

# Le projet



## État à la reprise

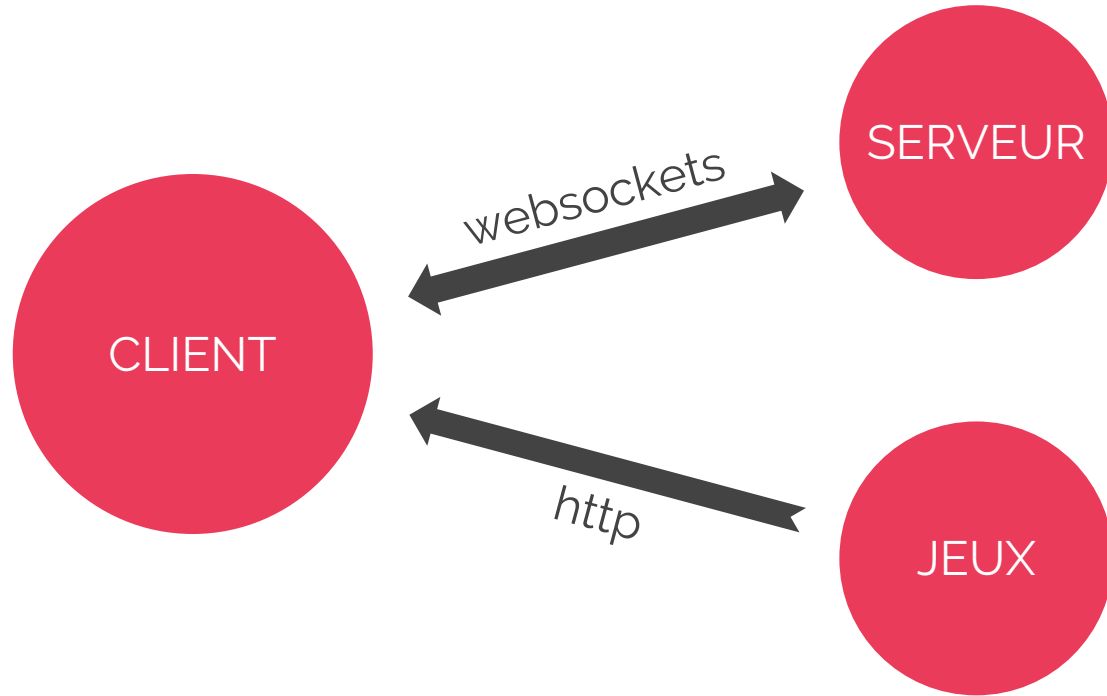
- Documentation technique
- Un POC



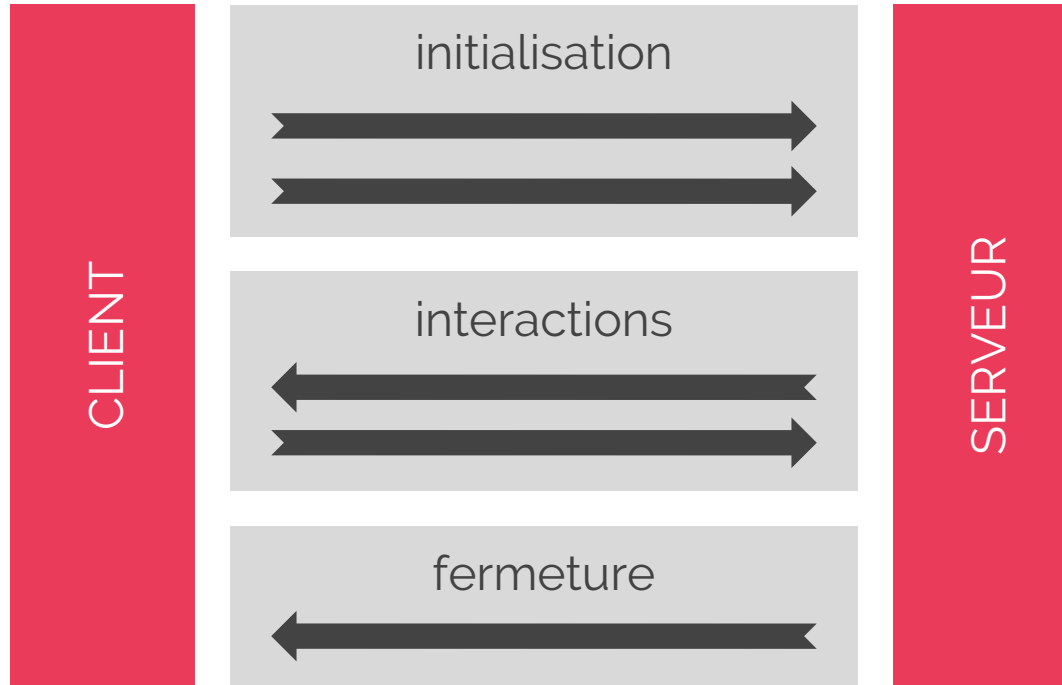
## Objectif

- Application complète
- Proposer un jeux de démonstration

# Structure



# Protocole



# Client



>\_ ruby

Run

Instructions


LiveCoding - Pokedash

C

Jeux

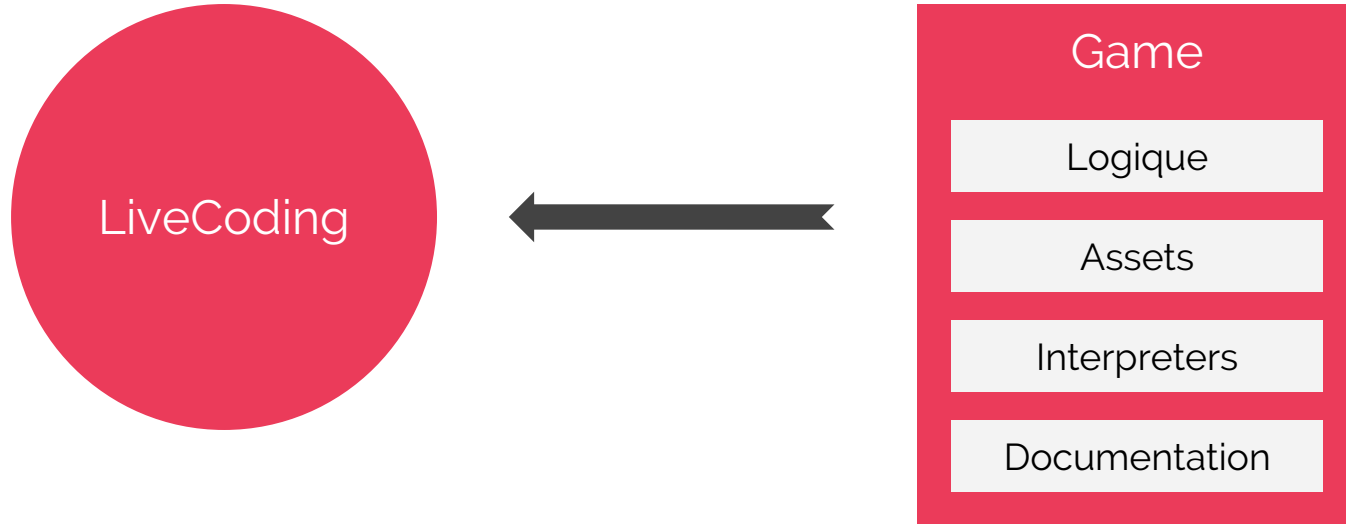
Paramètres

```
1 # Infinite loop to handle array key event
2 while true do
3
4   # Check if the action we are going to perform will happen in
5   if is_in_map(x, y, LIM_X, LIM_Y, dir, DIST) == 'true' # The b
6
7     # Checking the element next to the protagonist to do a sp
8     element = get_element(dir, DIST)
9
10    case element
11    # Moving Pikachu 1 case
12    when 'Road'
13      swap_sprite(dir, 0, DIST)
14
15    #If we can push boulder, move boulder and pikachu one cas
16    when 'Boulder'
17      if is_in_map(x, y, LIM_X, LIM_Y, dir, DIST+1) == 'tru
18        if get_element(dir, DIST + 1) == 'Road'
19          swap_sprite(dir, DIST, DIST + 1) # Swapping t
20          swap_sprite(dir, 0, DIST) # Swapping the prot
21        else
22          end
23        end
24
25    # Take the objective
26    when 'Objective'
27      take_objective
28      swap_sprite(dir, 0, DIST)
29    # Once the door is an objective left
```



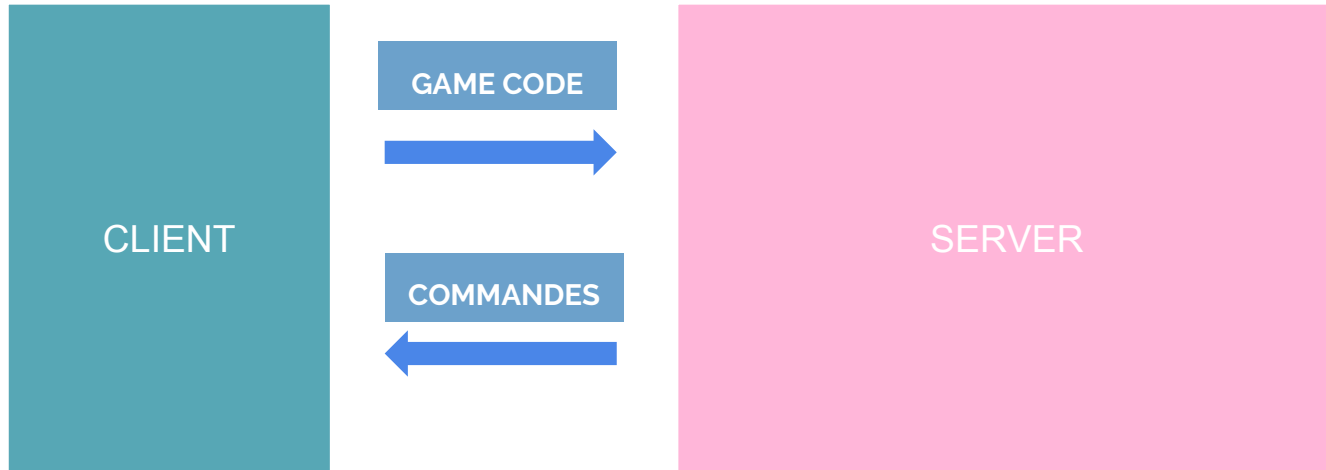
```
[14:44:52] Game box mounted, ready to load game !
[14:44:52] OK, Game loaded !
[14:44:52] OK, Game started, ready to go !
```

# Gestion des jeux



# Serveur

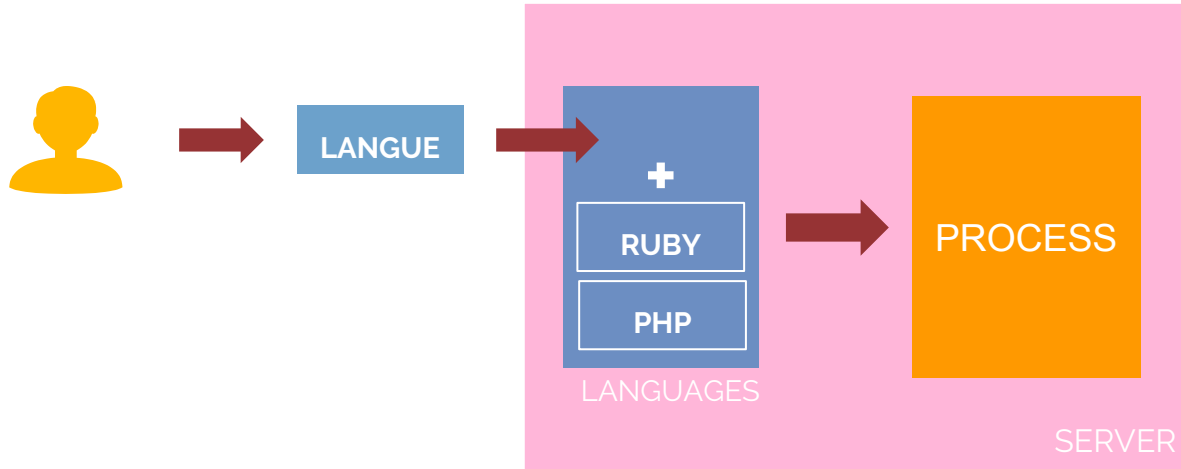
ROLE





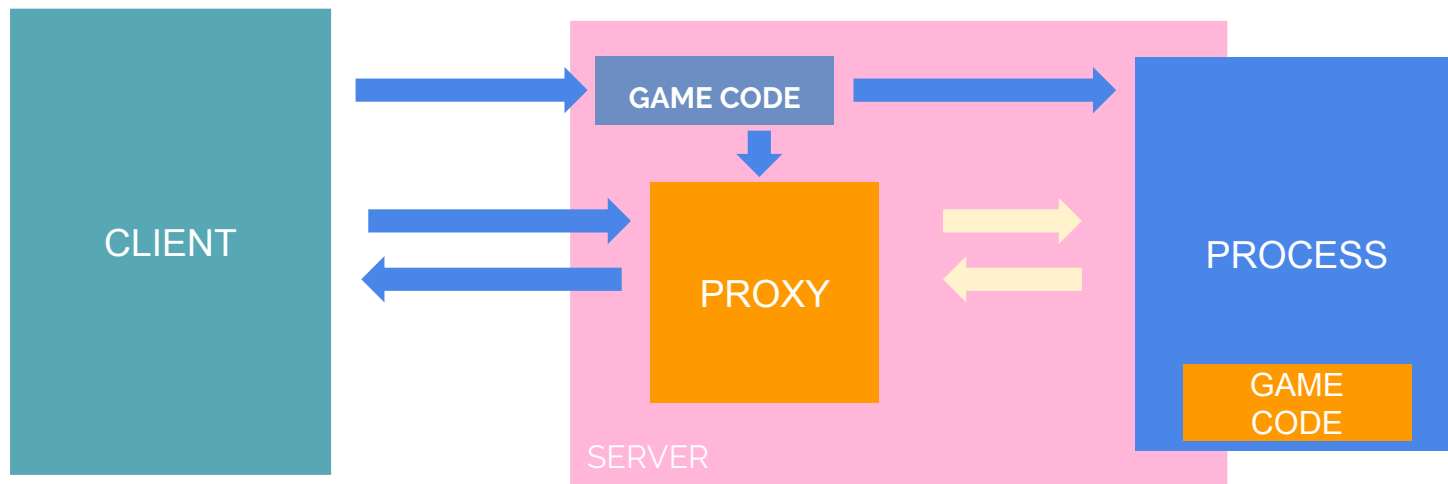
# Serveur

MODULE LANGUAGES



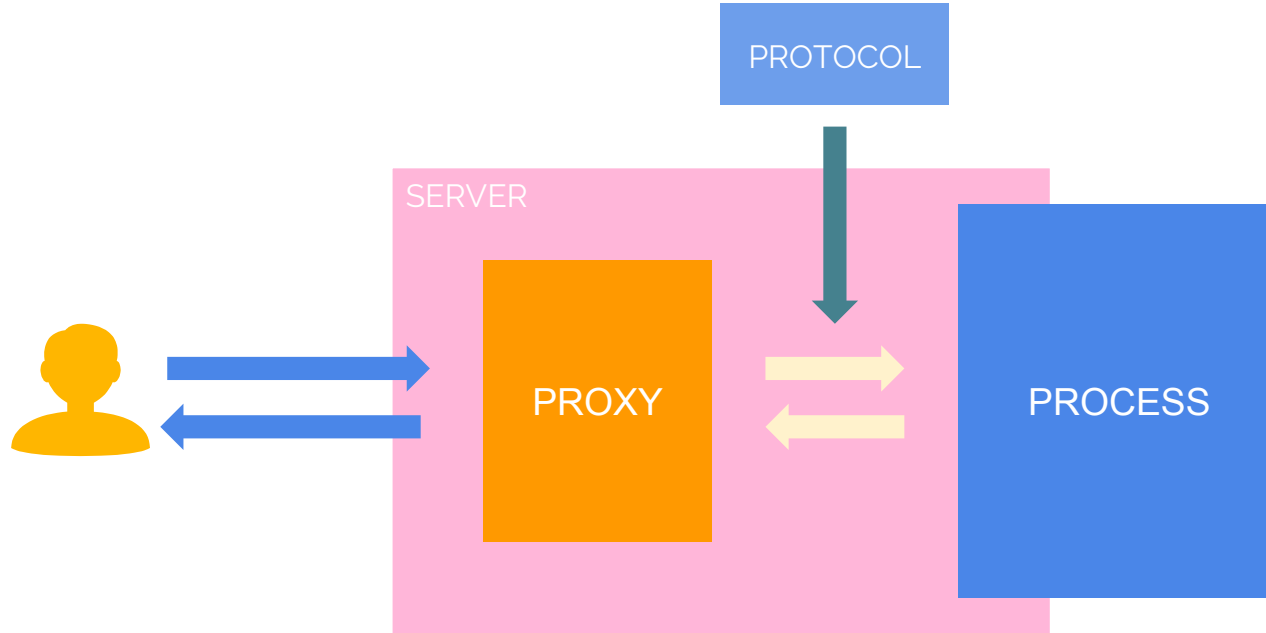
# Serveur

PROXY



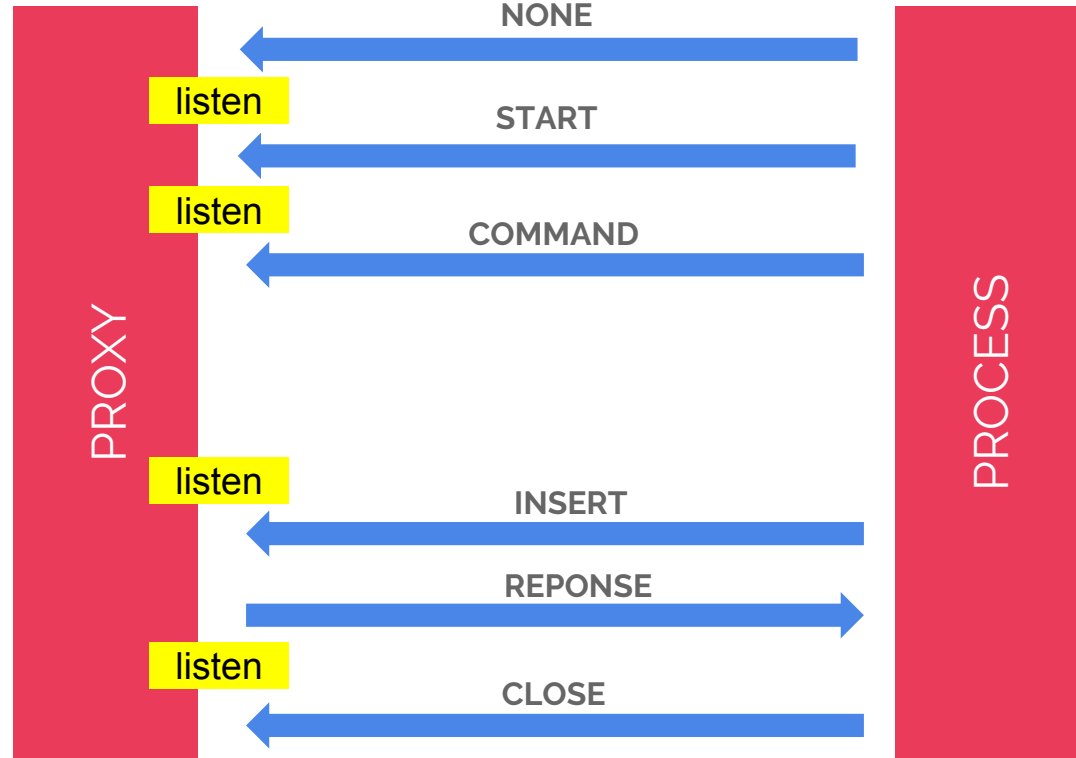
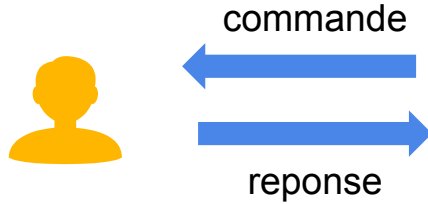
# Serveur

PROTOCOL



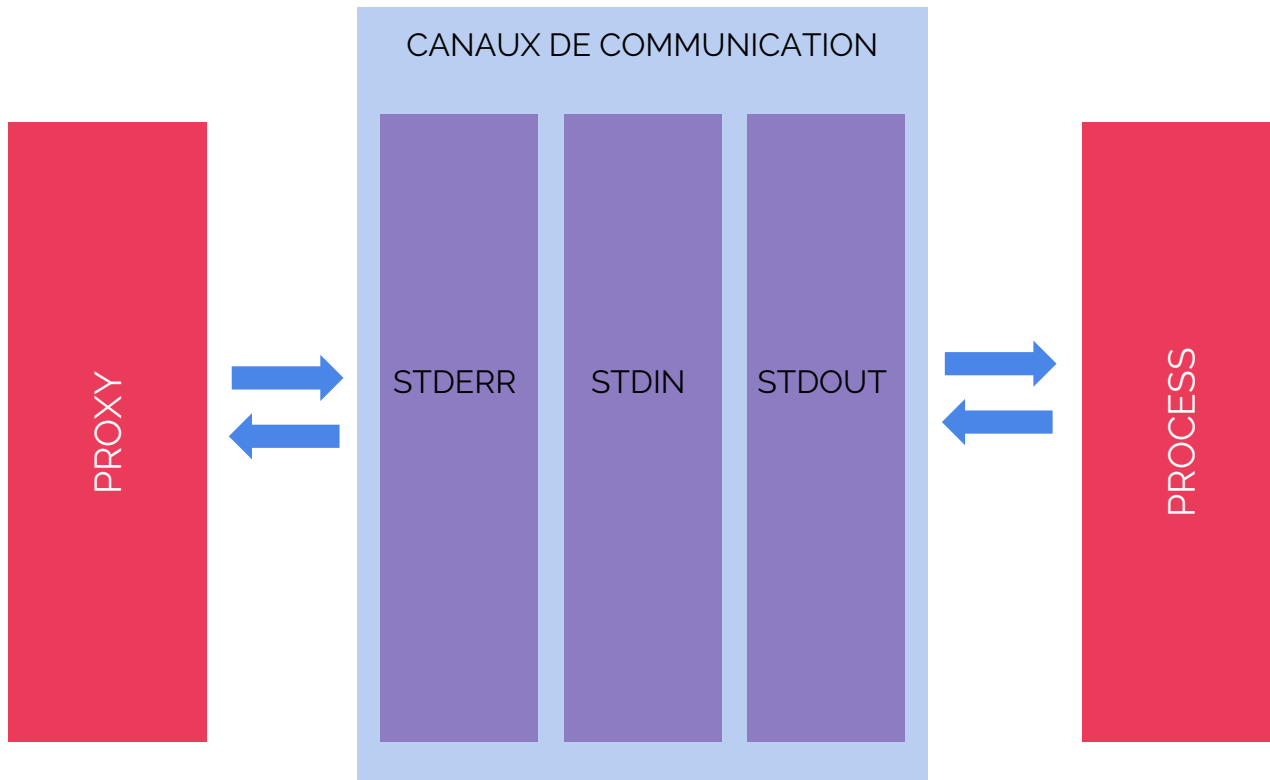
# Serveur

PROTOCOL

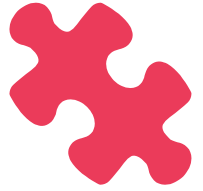


# Serveur

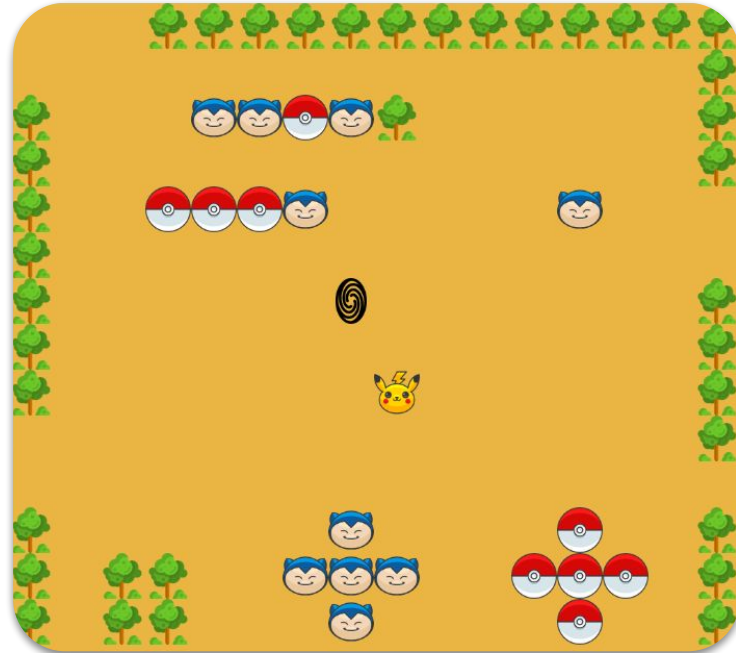
PROTOCOL



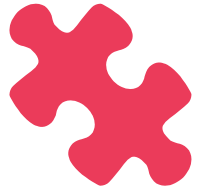
# Jeux



- Apprendre à coder de manière ludique
- Utiliser les contrôles de base
- Coder avec plusieurs langages de programmation



# Jeux



p5\*Js



# Améliorations



- Sécurité
  - Isoler le jeux
  - Isoler les processus
- Plus de langages supportés
- Autocomplétion
- Gestion des erreurs de syntaxe



# Conclusion



- Jeux indépendants
- Ajout simplifié de nouveaux langages
- Plateforme didactique pluridisciplinaire



# Questions ?