

LiveCoding platform, executed on the client (browser tab)

Web page

- Provides a html element where the game can be rendered
- Display an editor for the user code

RessourcesProvider

- Manage the acces to the game sources
- Build the required urls to load the resources

Game

- Loaded and started by the GameManager
- Manage the game display and game logic

GameManager

- Responsible to load the game
- Start the game
- Proxy game commands sended by the processor to the game engine

ProcessorProvider

- Provides the access to the processor (abstracts the websockets managment)
- Proxy game commands between the game manager and the processor

External game source (host the game files)

GitHub repository

File server

The processor, server side, unix based system.

Processor (python server).

- Loads the supported languages
- Expose a websocket endpoint
- Launch corresponding subprocess to run the user code
- When a game is running, takes the role of proxy between subprocess and game engine

When the client launch his code, the processor will load the corresponding language files, and spawn a subprocess to execute it.

php 1

php 2

ruby 1

...