

Disseny del joc SickieBlowOne

Adrià Garriga Alonso

11 de juliol de 2013

Llegenda: **Classe abstracta**

- *Classe*: hereda de l'anterior

1 Explicació de totes les classes

Level: tot el que tregui dades aptes per a **LevelRenderer**

- **PLevel**: tot nivell que realitza simulacions físiques a partir de moviment i accions dels seus jugadors
 - *PLevelDeProva*
 - *PLevelCampDeBatalla*
 - *etc.* tots els nivells específics. Cadascun en una llibreria compartida.
- **NLevel**: es basa en dades rebudes per un *NetworkManager*
 - *NLevelDeProva*
 - *NLevelCampDeBatalla*
 - *etc.* tots els nivells específics. Cadascun en una llibreria compartida.

Player: tot el que treu dades renderitzables per un **PlayerRenderer**

- **PPlayer**: mira els controls d'una **Interface** i mou un objecte físic i fa accions en un **PLevel** de manera corresponent.
 - *PPlayerBoxejador*
 - *PPlayerCavallerGel*
 - *etc.* tots els jugadors específics. Cadascun en una llibreria compartida.
- **NPlayer**: es mou basat en un *NetworkManager*
 - *NPlayerBoxejador*
 - *NPlayerCavallerGel*
 - *etc.* tots els jugadors específics. Cadascun en una llibreria compartida.

Interface: tradueix algun tipus d'input en dades identificables per a un **PPlayer**

- *JoystickInterface*: agafa un *OIS::Joystick*
- *KeyboardInterface*: agafa un *OIS::Keyboard*
- *NetworkInterface*: agafa dades d'un *NetworkManager*

PlayerRenderer: A partir d'un **Player**, dibuixa el jugador en la pantalla

- *PlayerRendererBoxejador*
- *PlayerRendererCavallerGel*
- *etc.*

LevelRenderer: A partir d'un **Level**, dibuixa el nivell en la pantalla

- *LevelRendererDeProva*
- *LevelRendererDeBatalla*
- *etc.*

GameSetup: (semi-abstracta) A partir de l'entrada de ratolí i teclat i les dades de jugadors i nivells dibuixa la HUD (barres de vida, n° de vides, opcions quan es criden...) i mou el “flow” del loop corresponent. També representa cadascun dels modes de joc.

Hereda de *Ogre::FrameListener*, *Ogre::WindowEventListener*, *OIS::KeyboardListener* i *OIS::MouseListener*. Cada subclasse té la seva llibreria compartida pròpia.

- *GameSetupLocalProva*: partida local de prova. Un sol jugador local. Conté:
 - *PLevel*: crea
 - *PPlayer*: crea i registra al *PLevel*
 - *Interface*: crea i registra al *PPlayer*
 - *PlayerRenderer*: crea i registra-li el *PPlayer*
 - *LevelRenderer*: crea i registra-li el *PLevel*
- *GameSetupLocal1v1*:
 - 1 *PLevel*
 - 2 *PPlayer*
 - 2 *Interface*
 - 2 *PlayerRenderer*
 - 1 *LevelRenderer*
- *GameSetupServer1v1*:
 - 1 *PLevel*
 - 2 *PPlayer*
 - 1 *Interface*
 - 1 *NetworkInterface*
 - 1 *NetworkManager*

- 1 LevelRenderer
- 2 PlayerRenderer
- *GameSetupClient1v1:*
 - 1 NLevel
 - 2 NPlayer
 - 1 Interface: ficar pel NetworkManager
 - 1 NetworkManager
 - 1 LevelRenderer
 - 2 PlayerRenderer
- *GameSetupServer1v1:*
 - 1 PLevel
 - 2 PPlayer
 - 2 NetworkInterface
 - 1 NetworkManager
- *GameSetupSpectator1v1:*
 - 1 NLevel
 - 2 NPlayer
 - 1 NetworkManager

ClassLoader: s'ocupa de carregar llibreries dinàmiques on es troben cada una de les subclasses de jugadors i nivells, dinàmicament.

CameraMan: es dedica a moure la càmera seguint a un jugador que se li indica

NetworkManager: s'ocupa de la comunicació via UDP i TCP/IP amb altres *NetworkManager*.
(TODO: EXTENDRE)

GUILauncher: S'ocupa de la GUI que s'inicia quan s'inicia el joc. Segons les opcions seleccionades, configura els gràfics, updateja el client al principi, fer login quan selecciones jugar en xarxa, configura el so, els controls, ensenya les taules de puntuacions... Bàsicament conté tota la GUI que no està dins de la partida. Quan es selecciona l'opció carrega recursos, inicia el GameSetup corresponent i li passa control.

Hereda de *Ogre::FrameListener*, *Ogre::WindowEventListener*, *OIS::KeyboardListener* i *OIS::MouseListener*.
(TODO: EXTENDRE)

2 La funció main

2.1 El Client

Inicia *Ogre* i el *GUILauncher*. Registra el *GUILauncher* com a *FrameListener*.

2.2 El Servidor

Inicia el *NetworkManager*, i segons les peticions que rep aquest crea el *GameSetupServer* corresponent.