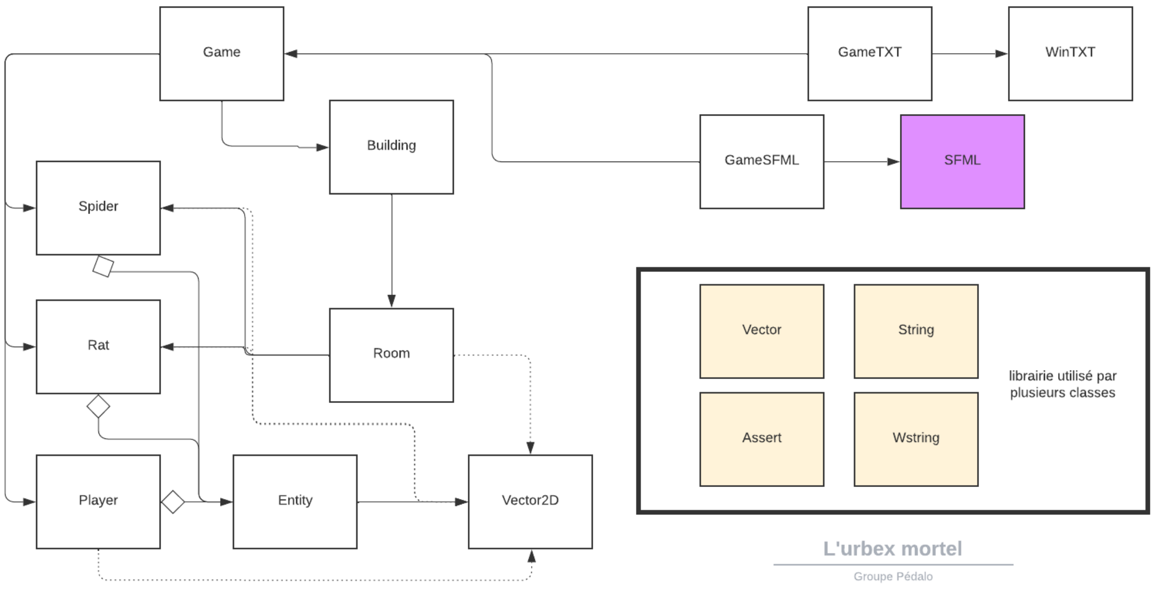

L'urbex mortel

Groupe Pédalo

Core :

```
Game
--------------------------------------------
-   building : Building*
-   room : unsigned int
-   player : Player*
-   rats : vector<Rat*>
-   spiders : vector<Spider*>
--------------------------------------------
+   Game()
+   ~Game()

+   setDifficulty(difficulty : unsigned int)
+   getBuilding() : Building*
+   getPlayer() : Player*

+   getNbRat() : unsigned int
+   getRat(i : unsigned int) : Rat*
+   addRat() : void
+   removeRat() : void
+   collisionRat() : void

+   getNbSpider() : unsigned int
+   getSpider(i : unsigned int) : Spider*
+   addSpider() : void
+   removeSpider() : void
+   collisionSpider() : void

+   update(time : time) : int
-   changeRoom() : bool

+   regressionTest() : void
```

building :

```
Building
--------------------------------------------
-   arrayRoom : vector<Room>
-   currentRoom : unsigned int
-   nbRoom : unsigned int
-   totalTime : unsigned int
--------------------------------------------
+   Building(nb : unsigned int)
+   Building(filename : string)
+   getCurrentRoom() : Room*
+   getIntCurrentRoom() : unsigned int
+   getNbRoom() : unsigned int
+   getTotalTime() : unsigned int
+   finishRoom() : bool
+   regressionTest() : void
```

PATH_ROOMS : const string
Obstacle : enum

Room
- dimX : unsigned int - dimY : unsigned int - time : unsigned int - arrayObstacle : vector<Obstacle> - arrayRat : vector<Vector2D> - arraySpider : vector<Vector2D>
+ Room() + Room(filename : string) + getDimX() : unsigned int + getDimY() : unsigned int + getObstacle(V : Vector2D) : Obstacle + setObstacle(V : Vector2D, o : Obstacle) : void + getNbRat() : unsigned int + getRat(i : unsigned int) : Vector2D* + getNbSpider() : unsigned int + getSpider(i : unsigned int) : Vector2D* + getTime() : unsigned int + regressionTest() : void

entity

Entity
- position : Vector2D - height : unsigned int - width : unsigned int
+ Entity() + Entity(p : Vector2D, w : unsigned int, h : unsigned int) + getHeight() : unsigned int + getWidth() : unsigned int + getPosition() : Vector2D + setPosition(const Vector2D & p) : void + up(const Room & R) : void + right(const Room & R) : void + down(const Room & R) : void + left(const Room & R) : void + gravity(const Room & R) : void + regressionTest() : void

enum Skin

Player : public Entity
- skin : Skin - hp : unsigned int - timeInvincible : unsigned int - orientation : bool
+ Player(p : Vector2D, s : Skin s, health : unsigned int) + getSkin() : Skin + setSkin(S : Skin) : void + getHp() : unsigned int + decreaseHp(h : unsigned int) : bool + getTimeInvincible() : unsigned int + decreaseTimeInvincible() : void + getOrientation() : bool + up(B : Building) : void + right(B : Building) : void + down(B : Building) : void + left(B : Building) : void + gravity(R : Room) : void - isMovePossibleUp(R : Room, V: Vector2D) : int - isMovePossibleSide(R : Room, V : Vector2D) : int - isMovePossibleDown(R : Room, V : Vector2D) : int - isMovePossibleGravity(R : Room, V : Vector2D) : int + standingOnBlock(R : Room) : bool + standingOnGhostBlock(R: Room) : void - drinkPotion(B : Building) : void + regressionTest() : void

Rat : public Entity
- direction : int - time : unsigned int
+ Rat(p : vector2D) + getDirection() : int + move(R : Room, P : Player) : void + gravity(R : Room) : void - findDirection(P : Player) : void - isPlayerArround(P : Player) : bool - isMovePossible(R : Room, V : Vector2D) : bool - isMovePossibleGravity(R : Room, V : Vector2D) : bool + regressionTest() : void

```
Spider : public Entity
--------------------------------------------------------
    -   direction : int
    -   time : unsigned int
--------------------------------------------------------
    +   Spider(p : Vector2D)
    +   getDirection() : int

    +   move(R : Room) : void
    -   isMovePossible(R : Room, V : Vector2D) : bool

    +   regressionTest() : void
```

```
Vector2D
--------------------------------------------------------
    -   x : unsigned int
    -   y : unsigned int
--------------------------------------------------------
    +   Vector2D()
    +   Vector2D(Vx : unsigned int, Vy : unsigned int)
    +   getX() : unsigned int
    +   getY() : unsigned int
    +   setX() : unsigned int
    +   setY() : unsigned int
    +   operator+(V : Vector2D) : Vector2D
    +   distance(V : Vector2D) : float
    +   regressionTest() : void
```

TXT

```
GameTXT
--------------------------------------------------------
    -   window : WinTXT
--------------------------------------------------------
    +   GameTXT(G : Game)
    +   loop(G : Game) : void
    -   draw(G : Game)
```

SFML

const string PATH_FONTS
const string PATH_MUSIC
const string PATH_SKINS
const string PATH_TEXTURES

GameSFML
- window : RenderWindow - textures : vector<Texture> - skins : vector<Skin> - spriteSize : unsigned int - close : bool
+ GameSFML(game : Game) + ~GameSFML() + loadTextures() : void + loadSkins() : void + loop(game : Game) : void - draw(game : Game) : void - drawBackground(dimX : unsigned int, dimY : unsigned int) - drawObstacles(room : Room) : void - drawPlayer(player : Player*) : void - drawSpider(spider : Spider*, room : Room) :void - drawRat(rat : Rat*) : void - drawInfoPlayer(game : Game) : void - drawString(str : wstring, y : unsigned int) - drawMenu() : void - drawEnd(victory : bool) : void + drawStory() : void + drawDifficultyMenu(game : game) : void + drawSkinMenu(game : Game) : void + randomizeTextures() : void + randomizeSkins() : void +