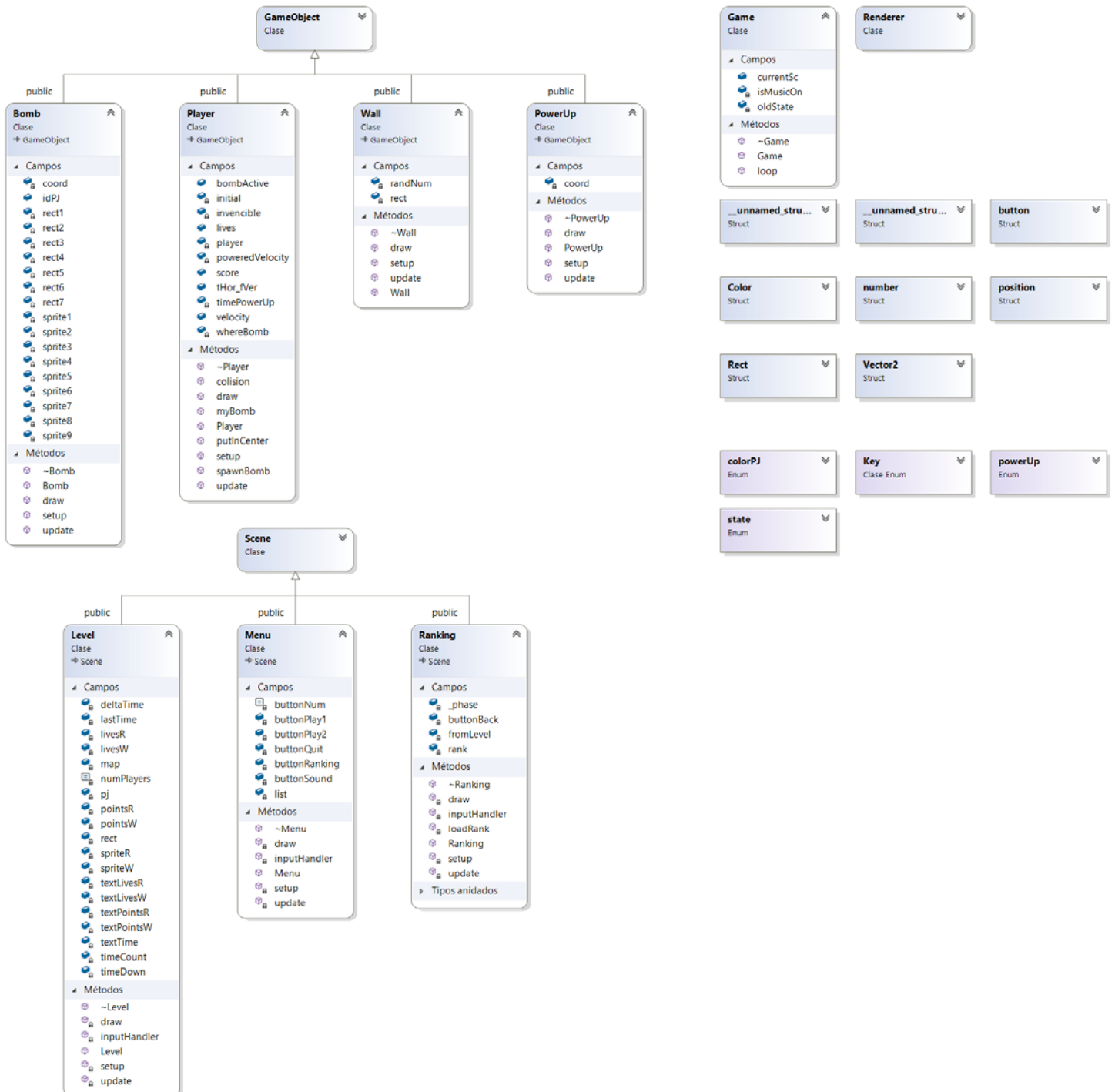


## Diagrama de classes generat des del VS2017



## Post-mortem

Objectes de la classe Player fora del mapa de GameObjects

Degut a que incloure als jugadors com a elements del mapa dificultaria el fet de que poguessin conviure més d'un element en cada posició del mapa (cosa necessària per a que objectes com les bombes es puguin superposar als jugadors) i que faria el moviment dels jugadors tosc (casella a casella en comptes de píxel a píxel) em optat per tractar als jugadors com a elements externs al mapa de GameObjects.

Això ens ha obligat a tractar les col·lisions comprovant els jugadors i els elements amb els que poden col·lidir píxel a píxel en comptes de a caselles mitjançant les funcions de l'arxiu collisions.hpp:

```
bool pointToRect(SDL_Point point, SDL_Rect rect)
    // detects if a SDL Point it's inside a SDL Rect

bool rectToRect(SDL_Rect rect1, SDL_Rect rect2)
    // detects if one of the four corner points of a SDL Rect(1) it's
    inside a SDL Rect(2)

bool doubleRectToRect(SDL_Rect rect1, SDL_Rect rect2)
    // detects if the two SDL Rects are colliding
```

A més de per aquesta tasca, també han facilitat altres aspectes com comprovar que l'usuari fa click sobre un botó determinat.

Funció put in center

Donat el cas que l'usuari vulgui passar entres dos objectes Wall, degut al moviment píxel a píxel hauria de trobar el píxel exacte per a que l'sprite del jugador encaixés exactament entre els dos Walls, cosa que faria la navegació pel mapa extremadament incòmode. Per pal·liar aquest defecte em definit la funció putInCenter(), que col·loca al jugador en la cel·la que més espai ocupa:

```
void putInCenter();
```

Cridem a la funció cada cop que el jugador canvia de direcció, per tal de centrar-lo en el carril que comença a transcorrer.

A més d'aquesta tasca, ha facilitat la col·locació dels nous objectes Bomb spawnjats per cada jugador, ja que al donar l'usuari la instrucció de fer-ho, el codi centra al jugador en la seva casella mitjançant aquesta funció i aleshores copia la posició del objecte Player en qüestió al nou objecte Bomb, fent que aquest quedi perfectament alineat en la seva casella.

Inicialització del Player

Com que la estructura del mapa està basada en un fitxer extern, que pot ser modificat per qualsevol, és susceptible de que un mur aparegui on el jugador havia de fer "spawn". Per tant, hem preferit per a inicialitzar la posició del jugador fer una funció que busqui l'espai buit més a prop de la cantonada corresponent a cada jugador.

### Comprovació del ranking.bin

En la execució del programa, pot ser que el binari del ranking no existeixi, perquè és la primera vegada que s'executa el programa en l'ordinador corresponent o perquè l'usuari l'ha destruït. Per tant, hem programat l'input del fitxer de manera que si no troba a ranking.bin es guardi a la memòria del programa 10 jugadors amb el nom "Empty Slot" i puntuació 0. D'aquesta manera, quan es tanqui el programa, es guardarà aquest binari. Durant la lectura del fitxer, també comprova que no sigui corrupte. Això es fa comprovant que la primera dada que s'agafa de cada jugador, la longitud del string del seu nom, sigui inferior a 100 chars. Per tal de que el jugador no pugui introduir un nom superior a 100chars, es redimensionarà el nom just després de que l'introdueixi.