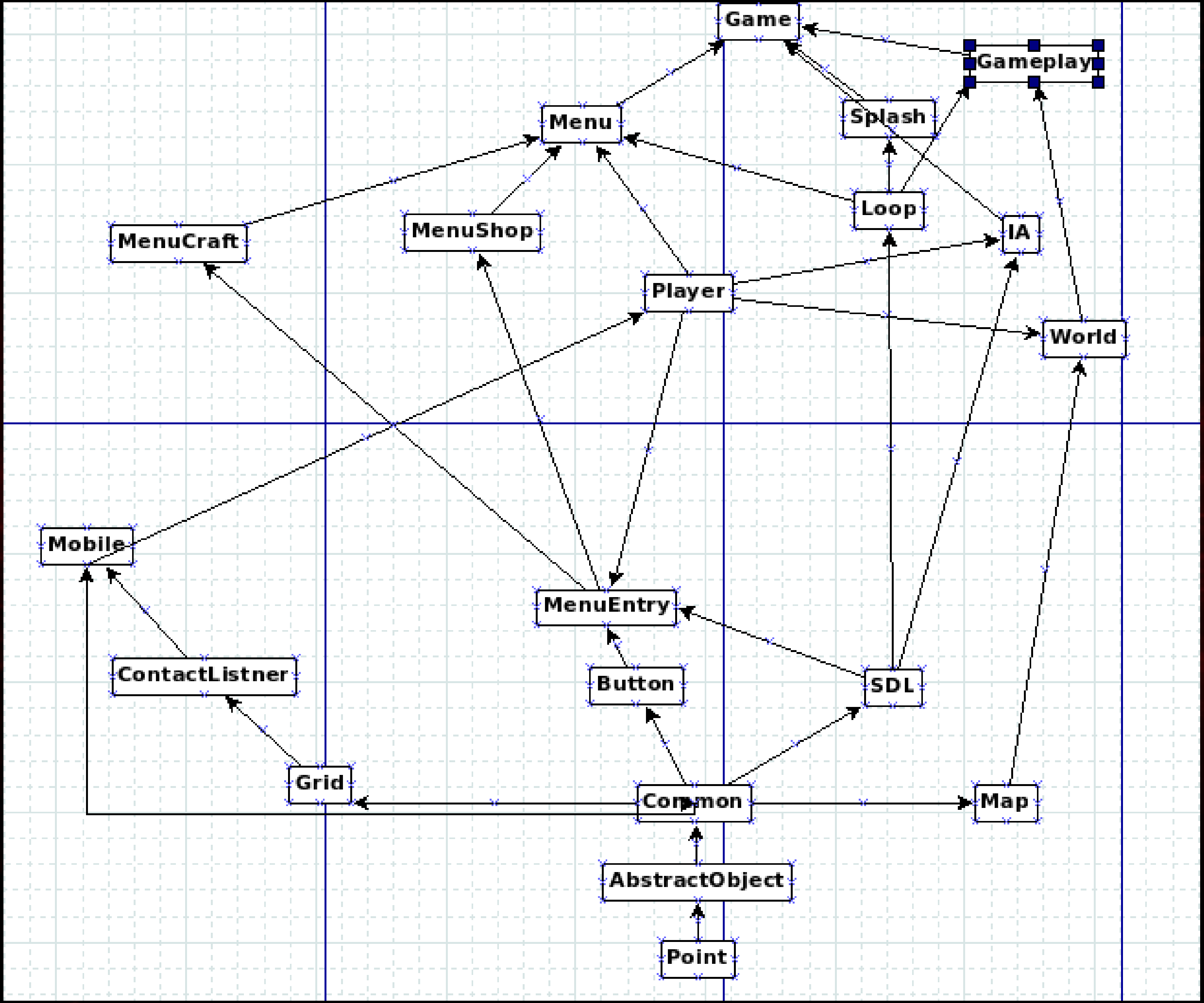
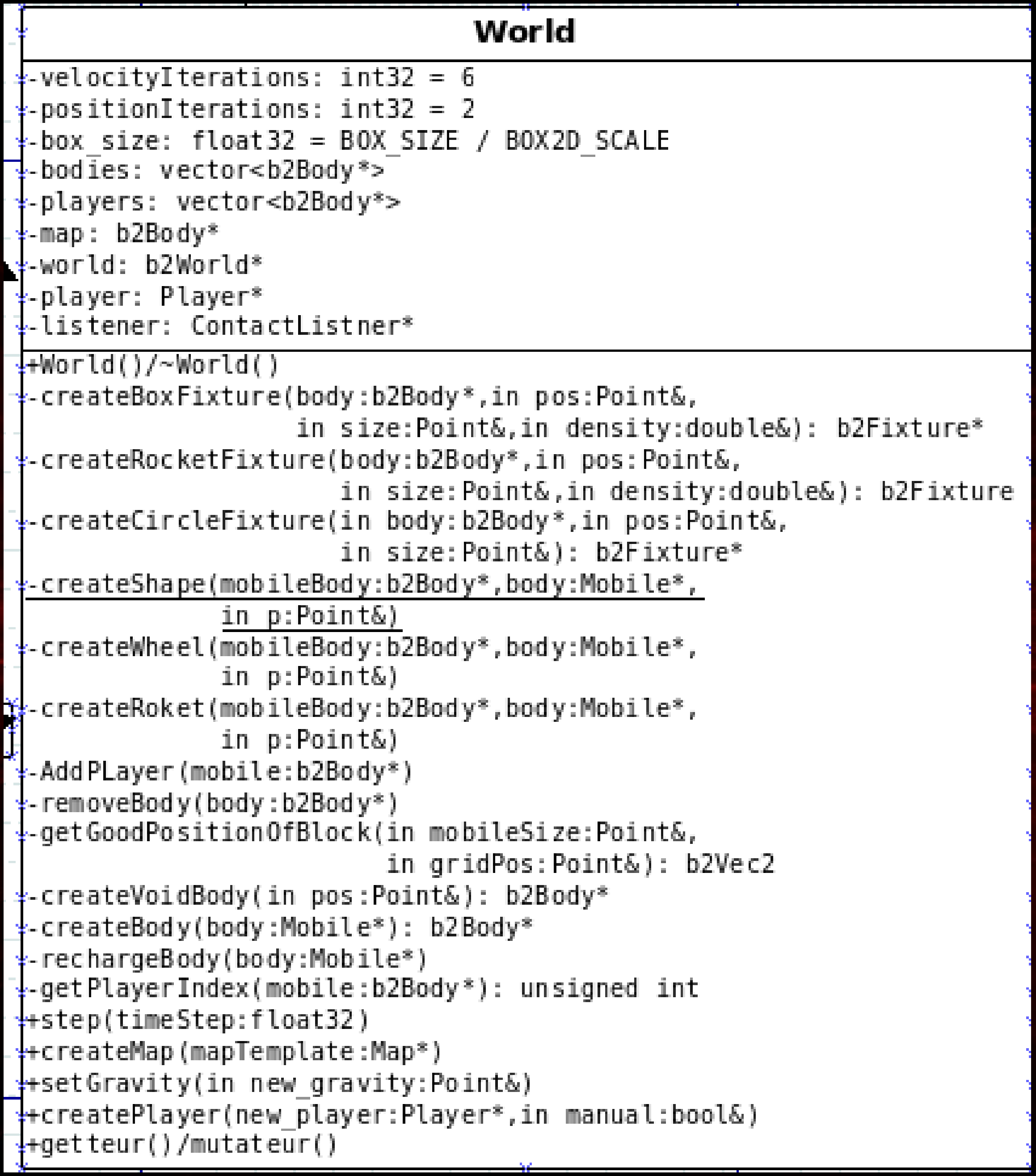
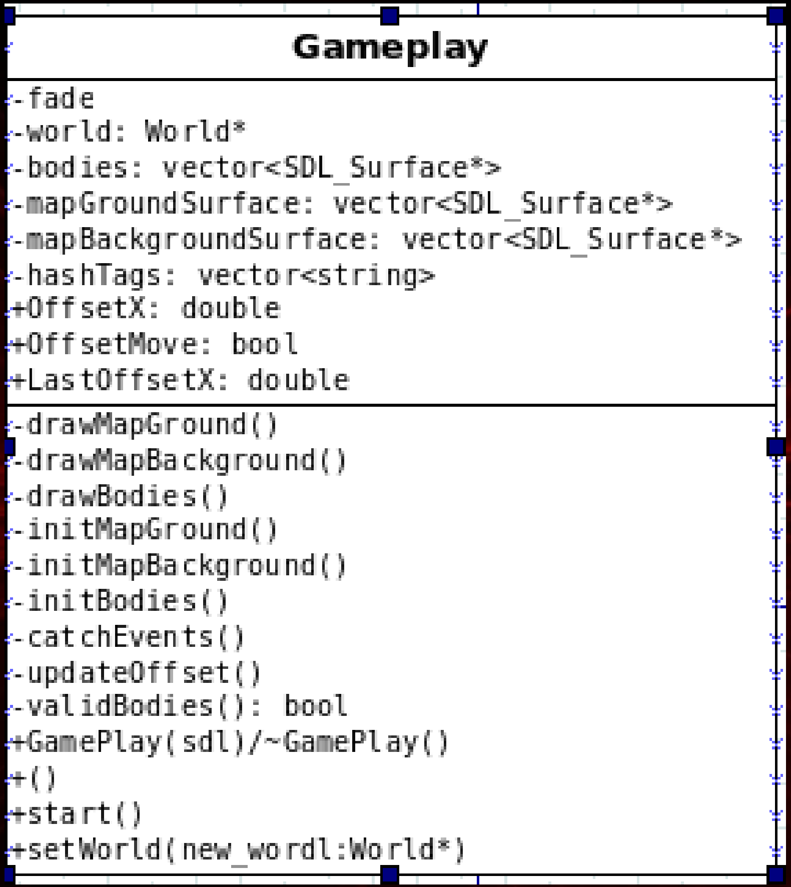
YOLOCRAFT


Game
Gameplay
Menu
Splash
Loop
IA
MenuCraft
MenuShop
Player
World
Mobile
MenuEntry
ContactListner
Button
SDL
Grid
Common
Map
AbstractObject
Point



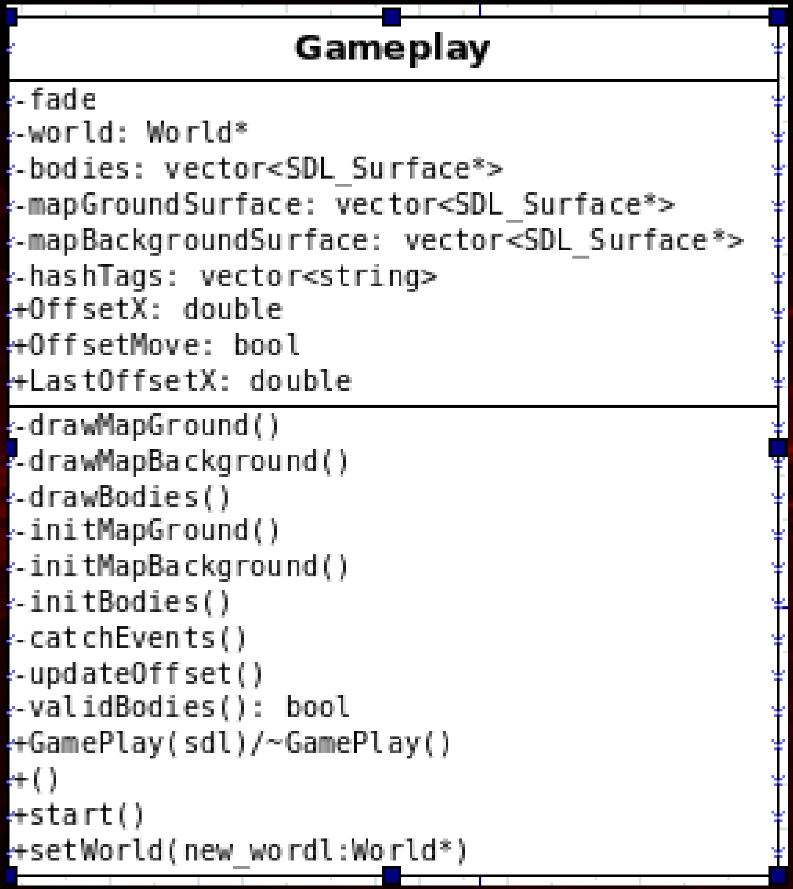
```
World
-velocityIterations: int32 = 6
-positionIterations: int32 = 2
-box_size: float32 = BOX_SIZE / BOX2D_SCALE
-bodies: vector<b2Body*>
-players: vector<b2Body*>
-map: b2Body*
-world: b2World*
-player: Player*
-listener: ContactListner*
+World()/~World()
-createBoxFixture(body:b2Body*,in pos:Point&,
                  in size:Point&,in density:double&): b2Fixture*
-createRocketFixture(body:b2Body*,in pos:Point&,
                     in size:Point&,in density:double&): b2Fixture
-createCircleFixture(in body:b2Body*,in pos:Point&,
                     in size:Point&): b2Fixture*
-createShape(mobileBody:b2Body*,body:Mobile*,
             in p:Point&)
-createWheel(mobileBody:b2Body*,body:Mobile*,
             in p:Point&)
-createRoket(mobileBody:b2Body*,body:Mobile*,
             in p:Point&)
-AddPLayer(mobile:b2Body*)
-removeBody(body:b2Body*)
-getGoodPositionOfBlock(in mobileSize:Point&,
                        in gridPos:Point&): b2Vec2
-createVoidBody(in pos:Point&): b2Body*
-createBody(body:Mobile*): b2Body*
-rechargeBody(body:Mobile*)
-getPlayerIndex(mobile:b2Body*): unsigned int
+step(timeStep:float32)
+createMap(mapTemplate:Map*)
+setGravity(in new_gravity:Point&)
+createPlayer(new_player:Player*,in manual:bool&)
+getteur()/mutateur()
```

Module World.
Includes :
Player.h
Map.h



```
Gameplay
-fade
-world: World*
-bodies: vector<SDL_Surface*>
-mapGroundSurface: vector<SDL_Surface*>
-mapBackgroundSurface: vector<SDL_Surface*>
-hashTags: vector<string>
+OffsetX: double
+OffsetMove: bool
+LastOffsetX: double
-drawMapGround()
-drawMapBackground()
-drawBodies()
-initMapGround()
-initMapBackground()
-initBodies()
-catchEvents()
-updateOffset()
-validBodies(): bool
+GamePlay(sdl)/~GamePlay()
+()
+start()
+setWorld(new_wordl:World*)
```

Modue GamePlay.
Includes : World.h
Loop.h



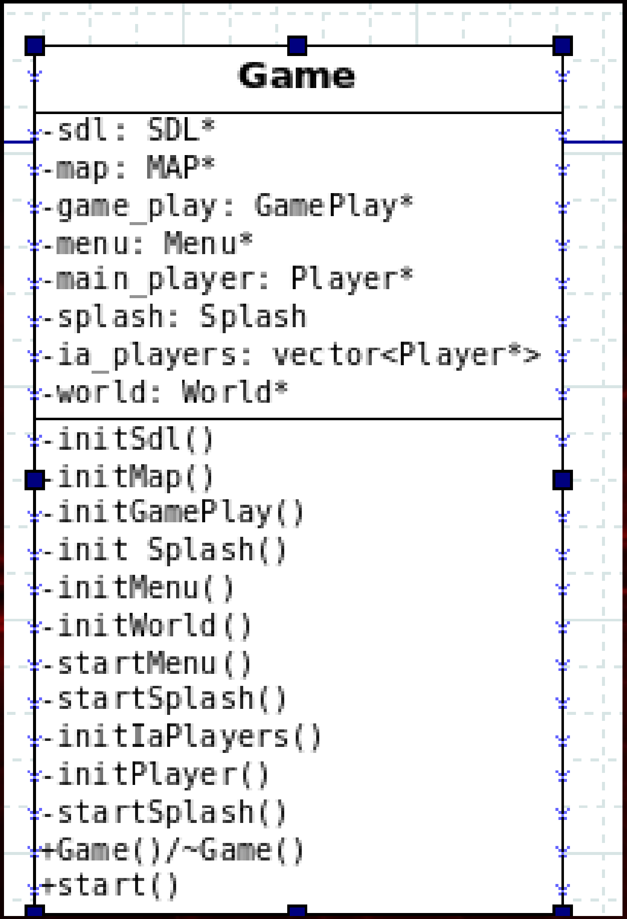
Gameplay

-fade
-world: World*
-bodies: vector<SDL_Surface*>
-mapGroundSurface: vector<SDL_Surface*>
-mapBackgroundSurface: vector<SDL_Surface*>
-hashTags: vector<string>
+OffsetX: double
+OffsetMove: bool
+LastOffsetX: double

-drawMapGround()
-drawMapBackground()
-drawBodies()
-initMapGround()
-initMapBackground()
-initBodies()
-catchEvents()
-updateOffset()
-validBodies(): bool
+GamePlay(sdl)/~GamePlay()
+()
+start()
+setWorld(new_wordl:World*)

Modue GamePlay.

Includes : World.h
Loop.h



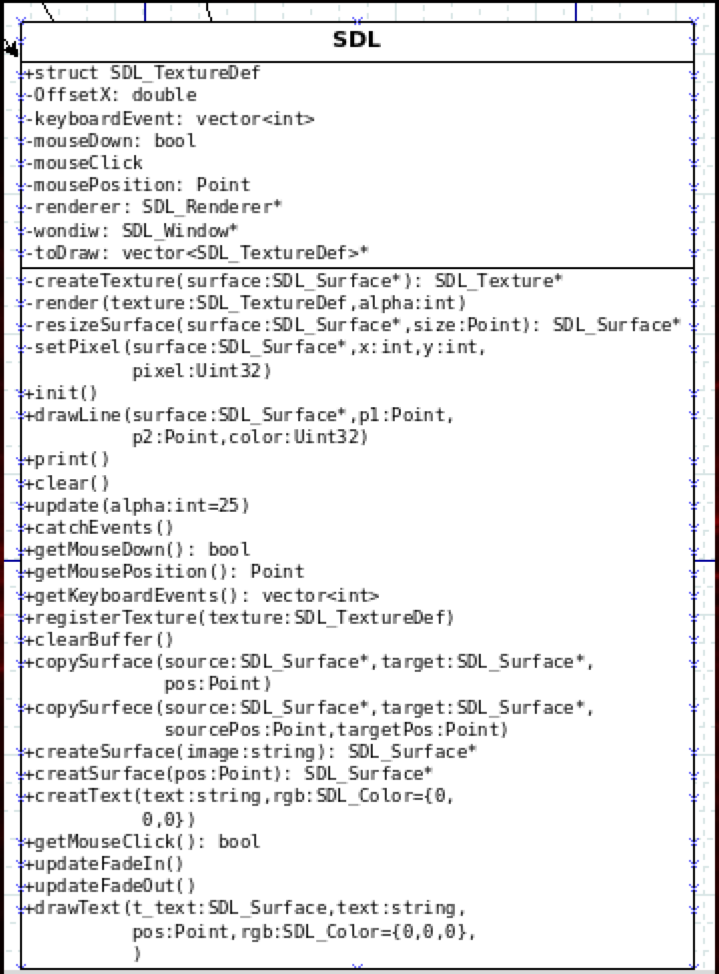
Game

-sdl: SDL*
-map: MAP*
-game_play: GamePlay*
-menu: Menu*
-main_player: Player*
-splash: Splash
-ia_players: vector<Player*>
-world: World*

-initSdl()
-initMap()
-initGamePlay()
-init Splash()
-initMenu()
-initWorld()
-startMenu()
-startSplash()
-initIaPlayers()
-initPlayer()
-startSplash()
+Game()/~Game()
+start()

Modue Game.

Includes : GamePlay
Menu
Splash



```
SDL
+struct SDL_TextureDef
-OffsetX: double
-keyboardEvent: vector<int>
-mouseDown: bool
-mouseClick
-mousePosition: Point
-renderer: SDL_Renderer*
-wondiw: SDL_Window*
-toDraw: vector<SDL_TextureDef>*
-createTexture(surface:SDL_Surface*): SDL_Texture*
-render(texture:SDL_TextureDef,alpha:int)
-resizeSurface(surface:SDL_Surface*,size:Point): SDL_Surface*
-setPixel(surface:SDL_Surface*,x:int,y:int,
          pixel:Uint32)
+init()
+drawLine(surface:SDL_Surface*,p1:Point,
          p2:Point,color:Uint32)
+print()
+clear()
+update(alpha:int=25)
+catchEvents()
+getMouseDown(): bool
+getMousePosition(): Point
+getKeyboardEvents(): vector<int>
+registerTexture(texture:SDL_TextureDef)
+clearBuffer()
+copySurface(source:SDL_Surface*,target:SDL_Surface*,
             pos:Point)
+copySurfece(source:SDL_Surface*,target:SDL_Surface*,
             sourcePos:Point,targetPos:Point)
+createSurface(image:string): SDL_Surface*
+creatSurface(pos:Point): SDL_Surface*
+creatText(text:string,rgb:SDL_Color={0,
           0,0})
+getMouseClick(): bool
+updateFadeIn()
+updateFadeOut()
+drawText(t_text:SDL_Surface,text:string,
          pos:Point,rgb:SDL_Color={0,0,0},
          )
```

<u>Modue SDL.</u>

Includes : Common.h

Yolocraftez joyeusement !