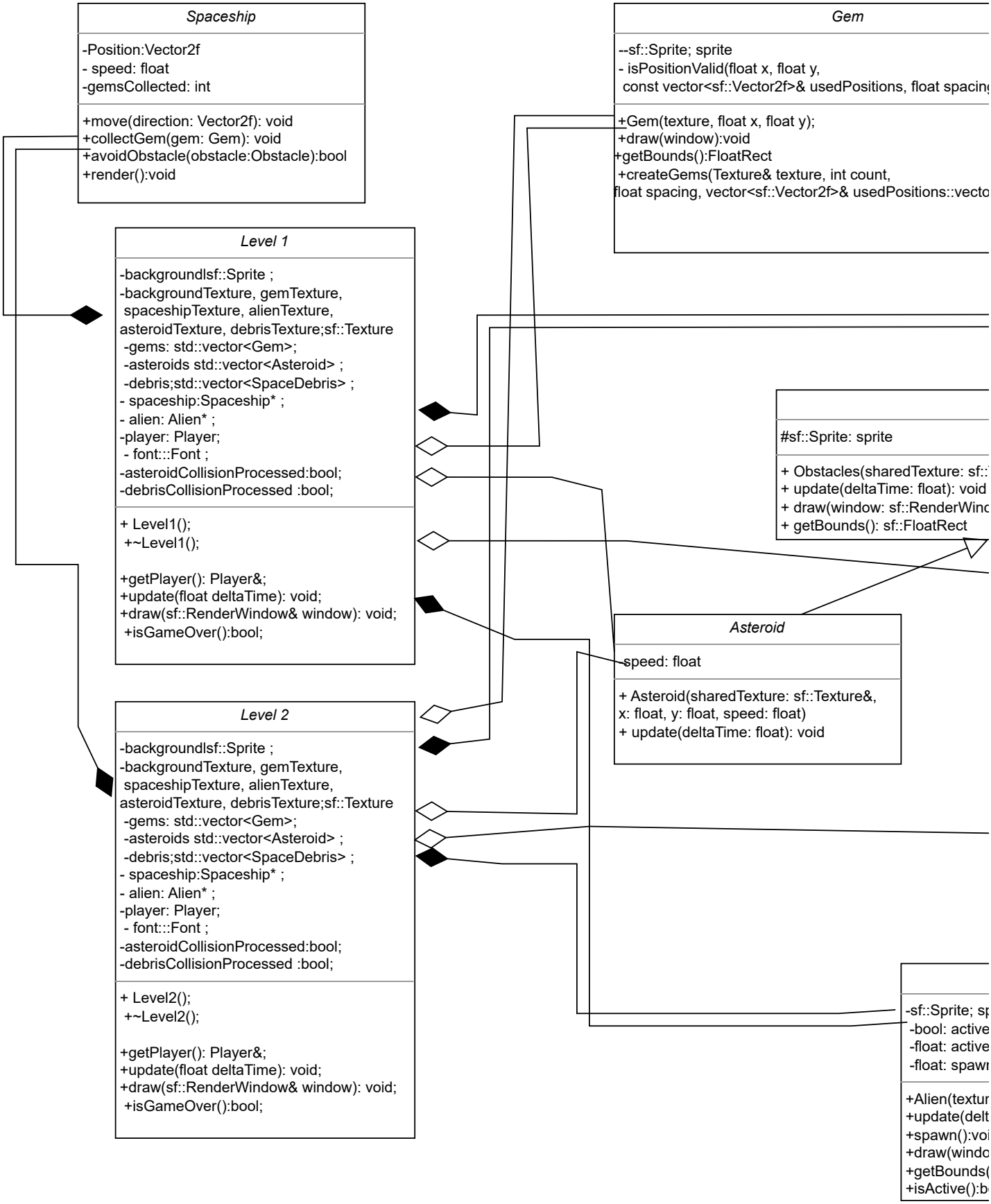
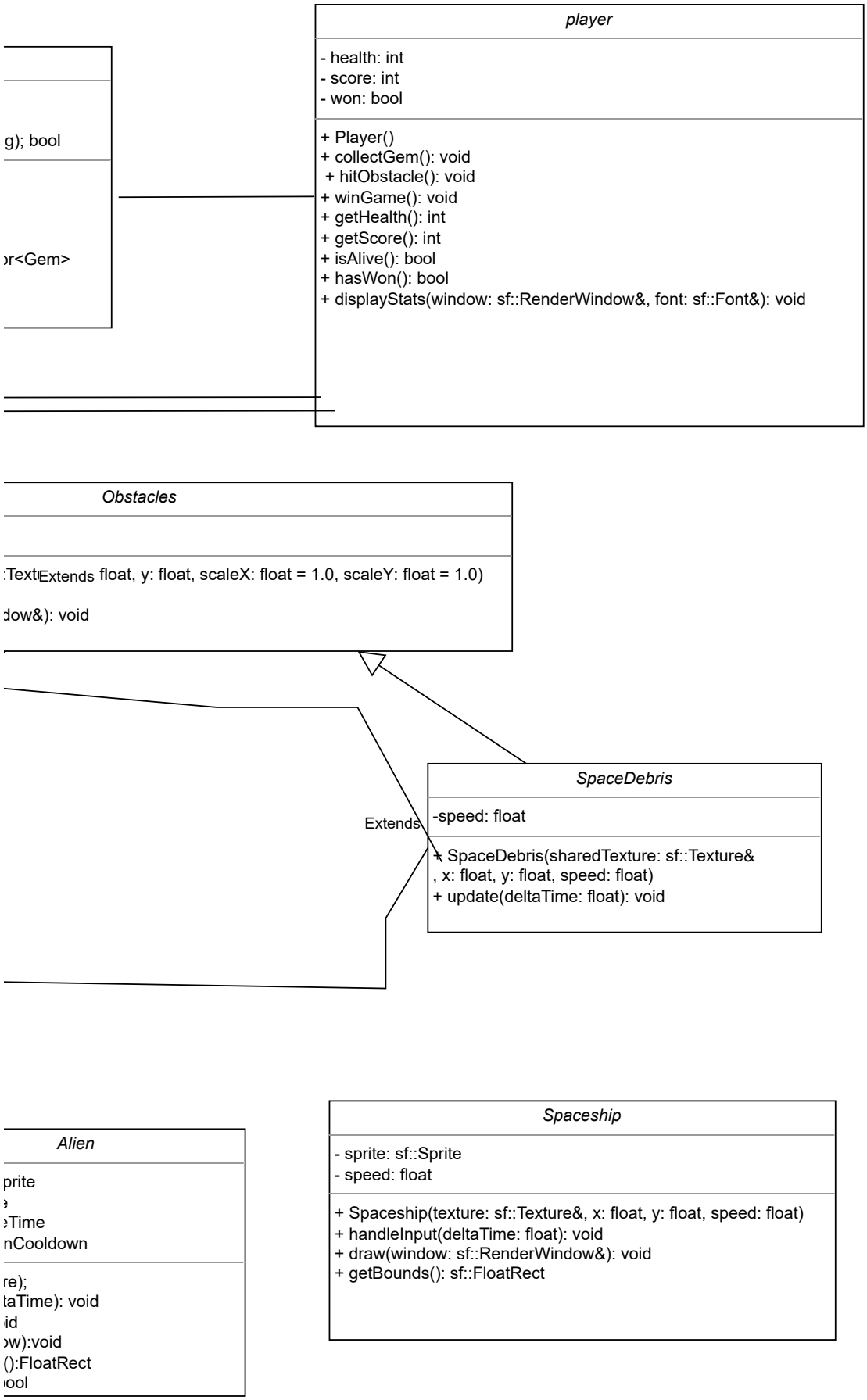

Spaceship
-Position:Vector2f
- speed: float
-gemsCollected: int
+move(direction: Vector2f): void
+collectGem(gem: Gem): void
+avoidObstacle(obstacle:Obstacle):bool
+render():void
Gem
--sf::Sprite; sprite
- isPositionValid(float x, float y,
const vector<sf::Vector2f>& usedPositions, float spacing
+Gem(texture, float x, float y);
+draw(window):void
+getBounds():FloatRect
+createGems(Texture& texture, int count,
float spacing, vector<sf::Vector2f>& usedPositions::vecto
Level 1
-backgroundlsf::Sprite ;
-backgroundTexture, gemTexture,
spaceshipTexture, alienTexture,
asteroidTexture, debrisTexture;sf::Texture
-gems: std::vector<Gem>;
-asteroids std::vector<Asteroid> ;
-debris;std::vector<SpaceDebris> ;
- spaceship:Spaceship* ;
- alien: Alien* ;
-player: Player;
- font:::Font ;
-asteroidCollisionProcessed:bool;
-debrisCollisionProcessed :bool;
+ Level1();
+~Level1();
+getPlayer(): Player&;
+update(float deltaTime): void;
+draw(sf::RenderWindow& window): void;
+isGameOver():bool;
#sf::Sprite: sprite
+ Obstacles(sharedTexture: sf::
+ update(deltaTime: float): void
+ draw(window: sf::RenderWind
+ getBounds(): sf::FloatRect
Asteroid
-speed: float
+ Asteroid(sharedTexture: sf::Texture&,
x: float, y: float, speed: float)
+ update(deltaTime: float): void
Level 2
-backgroundlsf::Sprite ;
-backgroundTexture, gemTexture,
spaceshipTexture, alienTexture,
asteroidTexture, debrisTexture;sf::Texture
-gems: std::vector<Gem>;
-asteroids std::vector<Asteroid> ;
-debris;std::vector<SpaceDebris> ;
- spaceship:Spaceship* ;
- alien: Alien* ;
-player: Player;
- font:::Font ;
-asteroidCollisionProcessed:bool;
-debrisCollisionProcessed :bool;
+ Level2();
+~Level2();
+getPlayer(): Player&;
+update(float deltaTime): void;
+draw(sf::RenderWindow& window): void;
+isGameOver():bool;
-sf::Sprite; sp
-bool: active
-float: active
-float: spawn
+Alien(textur
+update(delt
+spawn():voi
+draw(windo
+getBounds(
+isActive():b


player
- health: int
- score: int
- won: bool
+ Player()
+ collectGem(): void
+ hitObstacle(): void
+ winGame(): void
+ getHealth(): int
+ getScore(): int
+ isAlive(): bool
+ hasWon(): bool
+ displayStats(window: sf::RenderWindow&, font: sf::Font&): void
g); bool
or<Gem>
Obstacles
:Text
Extends
float, y: float, scaleX: float = 1.0, scaleY: float = 1.0)
dow&): void
SpaceDebris
-speed: float
+ SpaceDebris(sharedTexture: sf::Texture&
, x: float, y: float, speed: float)
+ update(deltaTime: float): void
Extends
Alien
prite
eTime
nCooldown
re);
taTime): void
id
ow):void
():FloatRect
ool
Spaceship
- sprite: sf::Sprite
- speed: float
+ Spaceship(texture: sf::Texture&, x: float, y: float, speed: float)
+ handleInput(deltaTime: float): void
+ draw(window: sf::RenderWindow&): void
+ getBounds(): sf::FloatRect