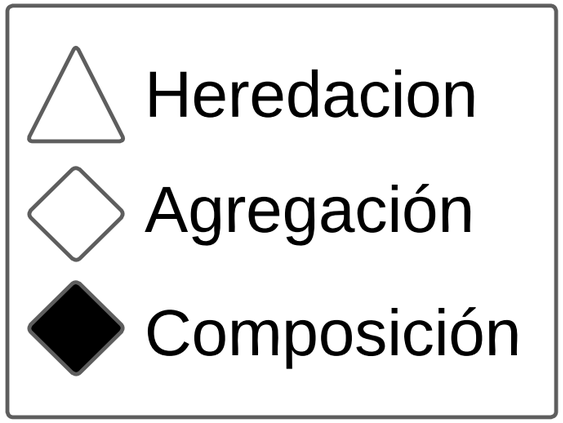

Hitbox

- hitbox: sf::FloatRect

+Hitbox(x: double, y: double, width: double, height: double)
+update(x: double, y: double): void
+checkCollision(asteroidPosition: const sf::Vector2f&, asteroidRadius: float): bool

Ship

- SCREEN_WIDTH: double
- SCREEN_HEIGHT: double
- SHIP_RADIUS: float
- hitbox: Hitbox
- collisionCount: int
- bullets: std::vector<Bullet>
- shootTimer: sf::Clock
- shootInterval: sf::Time
- hasCollided: bool
- shape: sf::ConvexShape
- angle: double
- x: double
- y: double
- speed: double
- acceleration: double
- rotationSpeed: double

+getX(): double
+getY(): double
+checkCollision(objectPosition: sf::Vector2f, objectSize: float): bool
+incrementCollisionCount(): void
+getCollisionCount(): int
+move(): void
+setAngle(newAngle: double): void
+accelerate(): void
+decelerate(): void
+rotateLeft(): void
+rotateRight(): void
+update(window: sf::RenderWindow&): void

Asteroid

- x: float
- y: float
- velocity: float
- directionX: float
- directionY: float
- shape: sf::CircleShape
- size: float

+Asteroid(x: float, y: float, velocity: float, directionX: float, directionY: float, size: float)
+checkCollision(bulletPosition: const sf::Vector2f&, bulletRadius: float): bool
+operator==(other: const Asteroid&): bool
+update(): void
+getShape(): const sf::CircleShape&
+getX(): float
+getY(): float
+getSize(): float

Bullet

- BULLET_SPEED: float
- BULLET_RADIUS: float
- shape: sf::CircleShape
- position: sf::Vector2f
- angle: float

+Bullet(x: float, y: float, angle: float)
+update(): void
+isOutsideWindow(window: const sf::RenderWindow&): bool
+operator==(other: const Bullet&): bool

Text

- tamaño: int
- color: Color
- fontObjeto: Font
- font: String

+ text()

Main

- score: int
- asteroids: std::vector<Asteroid>
- window: sf::RenderWindow
- playerShip: Ship
- coordinatesText: Text
- scoreText: Text
- collisionText: Text
- gameText: Text
- asteroidSpawnClock: sf::Clock

+main()