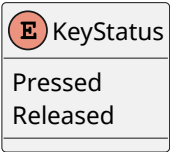
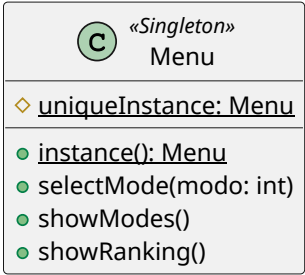
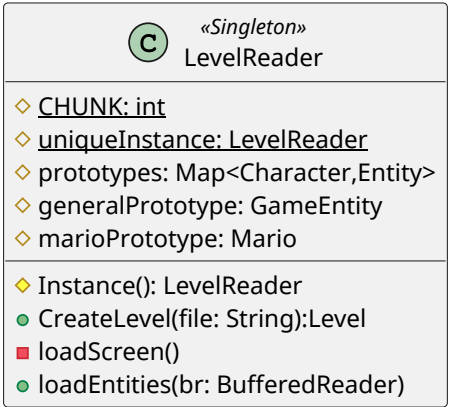
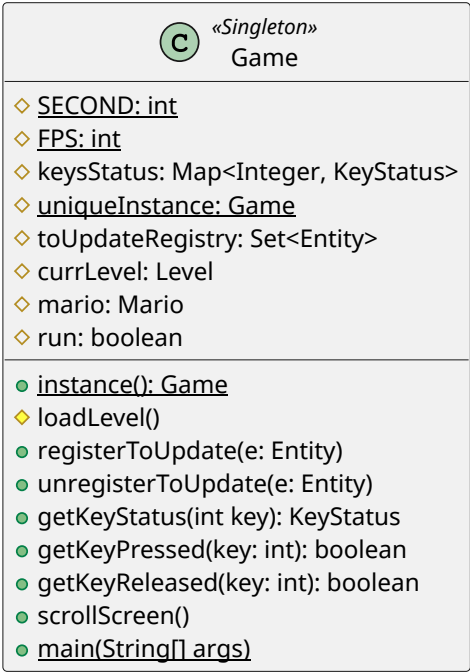
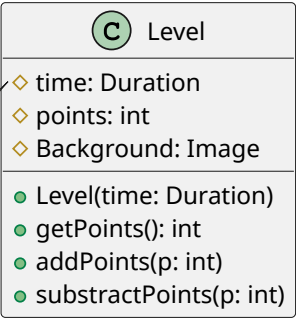


Duration es la clase de Java



C Mario

◇ marioStates: List<MarioState>

◇ brickCollisions: List<BrickCollision>

◇ enemyCollisions: List<EnemyCollision>

◇ graphicElement: GraphicElement

◇ collider: Collider

◇ position: Point

◇ loaded: boolean

◇ speedX: int

◇ lives: int

● Mario(gE: GraphicElement, collider: Collider, p: Point)

● clone(): Entity

● update()

● updateScreenPosition(levelWidth: int)

● jump()

● die()

● collide(b: SolidBrick)

● collide(b: SolidBlock)

● collide(e: Enemy)

● getCollider(): Collider

● getGraphicElement(): GraphicElement

● addMarioState(s: MarioState)

● removeMarioState(s: MarioState)

● addEnemyCollision(c: EnemyCollision)

● removeEnemyCollision(c: EnemyCollision)

● addBrickCollision(c: BrickCollision)

● removeBrickCollision(c: BrickCollision)

● getMarioStates(): Iterable<MarioState>

● setSpeedX(i: Int)

● addLife()

C «Singleton» FireBallThrower

◇ uniqueInstance: FireBallThrower

◇ mario: Mario

● instance(): FireBallThrower

● update()

C «Singleton» Croucher

◇ uniqueInstance: Croucher

◇ mario: Mario

◇ crouched: boolean

● instance(): Croucher

● getCrouch(): boolean

● update()

I «Command» MarioState

● setState()

● removeState()

I «Strategy» EnemyCollision

● collide(e: Enemy)

I «Strategy» BrickCollision

● collide(b: SolidBrick)

C Fire

C Invincible

C Super

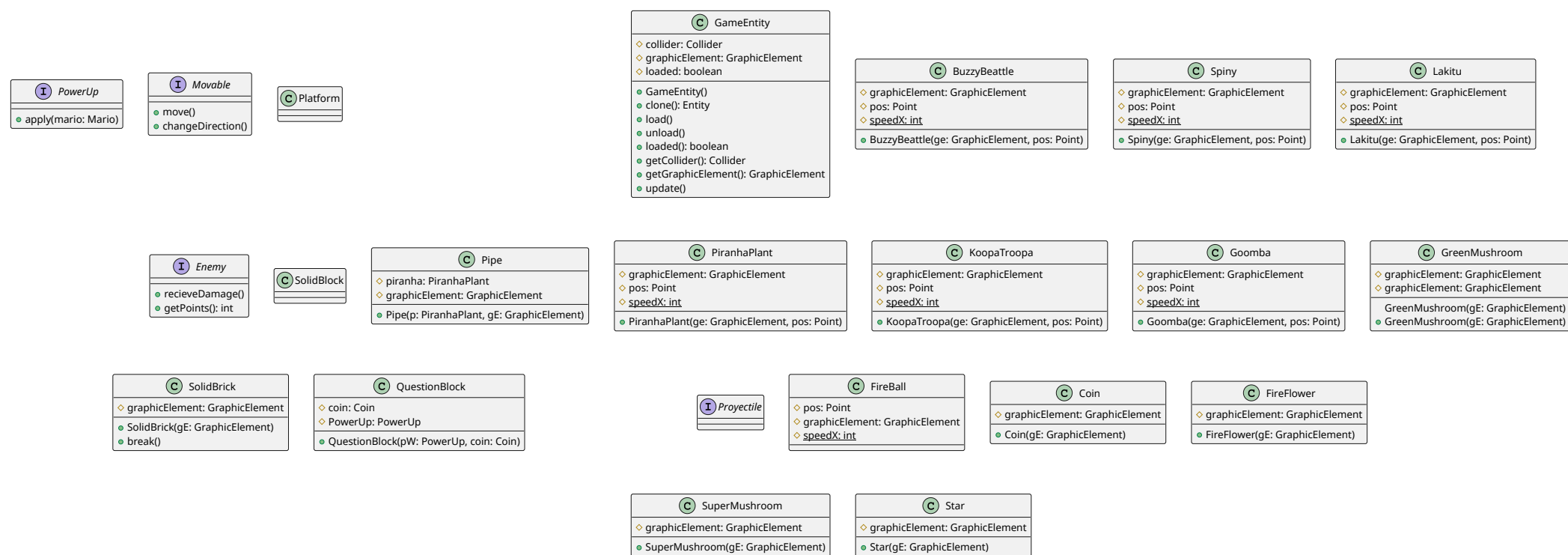
C DefaultEnemyCollision

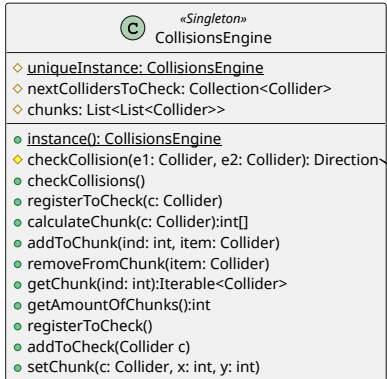
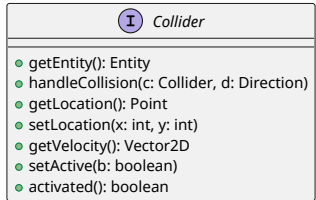
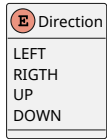
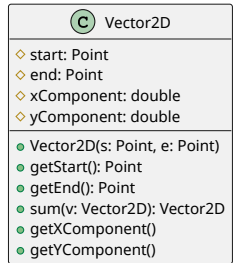
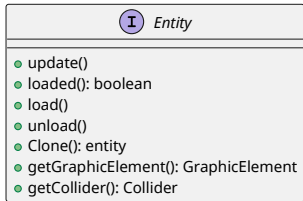
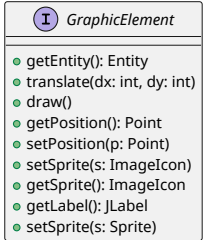
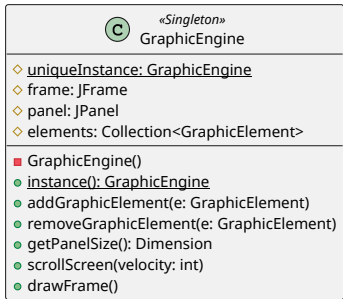
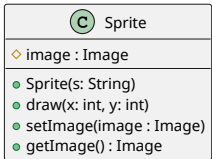
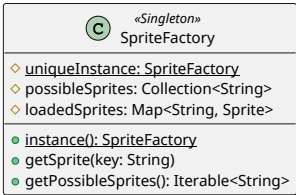
C SuperEnemyCollision

C InvincibleEnemyCollision

C DefaultBrickCollision

C BreakBrickCollision





if (hayCollision) {  
    c2.collide(c1, d1);  
    c1.collide(c2, d2);  
}

