

XVelocity, YVelocity: int
maxVelocity: int
height, width: int
observer: Observer
entitySprite: Sprite
collider: Collider

+ stopDownwardsMovement()
+ stopUpwardsMovement()
+ stopLeftwardsMovement()
+ stopRightwardsMovement()
+ notifyObserver()
+ getCollisionBox(int): Rectangle
+ destroy()
+ clone():GameEntity
+ attachObserver(Observer)
+ removeObserver()
+ setXPosition(int)
+ setYPosition(int)
+ addXOffset(int)
+ addYOffset(int)
+ getXPos(): int
+ getYPos(): int
+ getXVelocity(): int
+ setXVelocity(int)
+ getYVelocity(): int
+ setYVelocity(int)
+ getMaxVelocity(): int
+ setMaxVelocity(int)
+ getHeight(): int

+ setButtonTransparent(JButton)
+ decorateStartGameButton()
+ setBackgroundImage()

EntityObserver

<<Interface>>
Visitable
+ accept(Visitor,String)

Ranking
data: Map<String, Integer>
+ getTop5(): Iterable<String>
+ addPlayer(Player)
+ saveData()
+ loadData()

Enemy *
+ moveLeft()
+ moveRight()
+ attack(PlayerCharacter)
+ clone():Enemy
+ getShouldTakeDamage(PlayerCharacter):boolean

Projectile *
+ bounce()

Fireball
+ bounce() +

Shell
+ bounce()

Goomba
+ attack(PlayerCharacter): boolean +

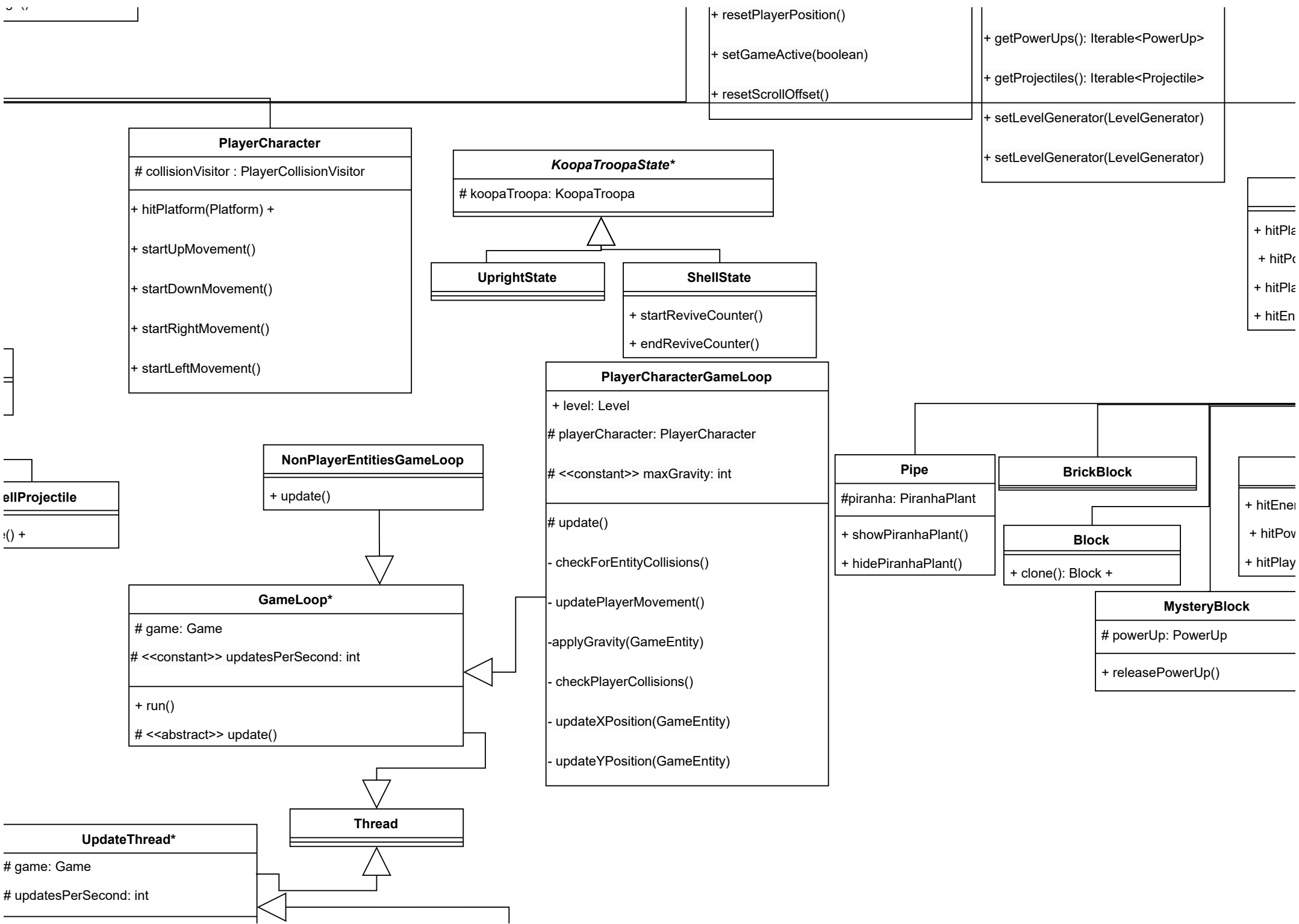
Spiny
+ attack(PlayerCharacter): boolean +

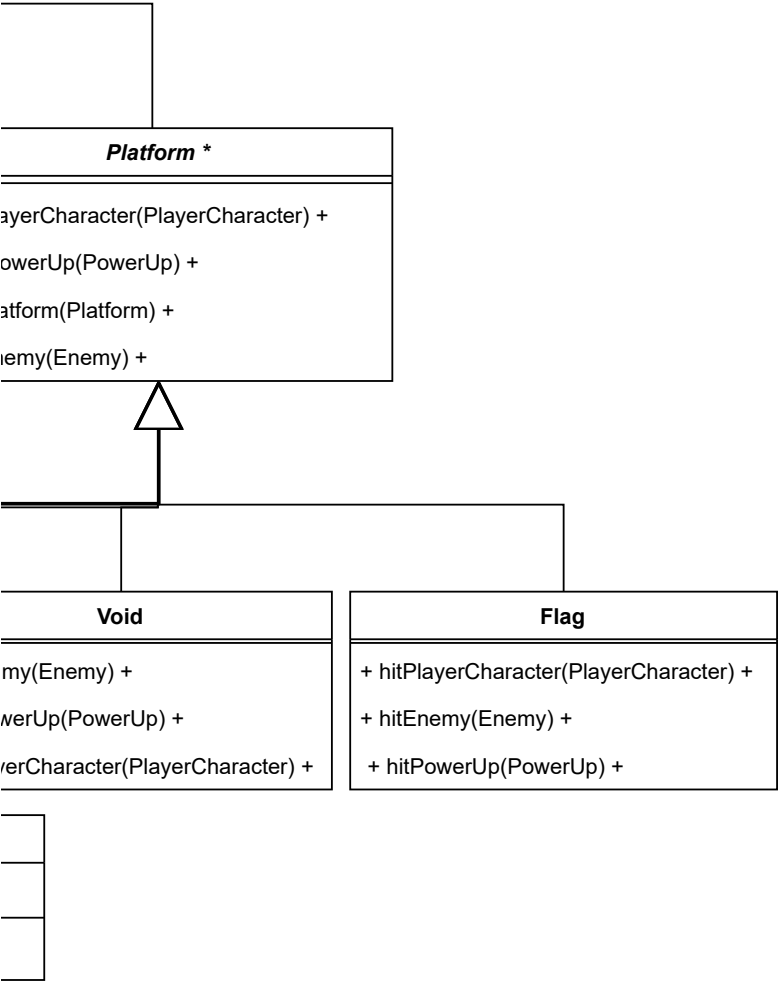
BuzzyBeetle
+ attack(PlayerCharacter): boolean +

KoopaTroopa
+ turnIntoShell()
+ attack(PlayerCharacter): boolean +

Lakitu
+ throwSpiny(): Spiny
+ attack(PlayerCharacter): boolean +

PiranhaPlant
hidden: boolean
+ emerge()
+ hide()
+ attack(PlayerCharacter): boolean +
+ getHidden(): boolean





# visualController: VisualGameController	- myEntity:GameEntity
# level: Level	- visitor: CollisionVisitor
# levelGenerator: LevelGenerator	- isCollidingFromBelow: boolean
# spriteFactory: SpriteFactory	- isCollidingFromAbover:boolean
# entityFactory: EntityFactory	- isCollidingFromRight:boolean
# playerCharacterGameLoop:PlayerCharacterGameLoop	- isCollidingFromLeft:boolean
# nonPlayerEntityGameLoop:NonPlayerEntityGameLoop	-directions: String []
+ setVisualController(VisualController)	+ checkPlataformCollisions(Plataform [] [])
+ startGame()	+ doCollisionIfColliding(GameEntity,int,GameEntity)
+ startGameLoop()	+ setCollisionDirection(int)
- registerObservers()	- getPlatformsToCheck(Platform[][], int, int):Platform []
+ registerPlatformObservers()	- doCollision(GameEntity)
+ registerEnemyObservers()	+ getEntity():GameEntity
+ registerPowerUpObservers()	+ setEntity(GameEntity)
+ registerProjectileObservers()	+ getIsCollidingFromBelow():boolean
+ registerPlayerObserver(PlayerCharacter)	+ setCollidingFromBellow(boolean)
+ registerEntityObservers(GameEntity)	+ getIsCollidingFromAbove():boolean
+ getLevel():Level	+ setCollidingFromAbove(boolean)
+ startUpPlayerMovement()	+ getIsCollidingFromRight():boolean
+ startDownPlayerMovement()	+ setCollidingFromRight(boolean)
+ startRightPlayerMovement()	+ getIsCollidingFromLeft():boolean
+ startLeftPlayerMovement()	+ setCollidingFromLeft(boolean)
+ stopRightPlayerMovement()	+ resetCollisionDirections()
+ stopLeftPlayerMovement()	
+ resetPlayerPosition()	

```
+ setHeight(int)

+ getWidth(): int

+ setWidth(int)

+ getObserver(): Observer

+ getSprite(): Sprite

+ hitPlayerCharacter(PlayerCharacter)

+ hitPowerUp(PowerUp)

+ hitPlatform(Platform)

+ hitEnemy(Enemy)

+ getCollider(): Collider

    +updateXPosition()

    +updateYPosition()
```

