Entity
- x: double
- y: double
- width: double
- height: double

+ getBounds: Rectangle

Inanimate
- x: double
- y: double
- width: double
- height: double

+ getBounds(): Rectangle

Animate
- x: double
- y: double
- width: double
- height: double
- speed: int

+ getBounds(): Rectangle

Game
+ field: type

+ method(type): type

Power-up
- x: double
- y: double
- width: double
- height: double

+ getBounds(): Rectangle
+ afterInteraction: void

Traps
- x: double
- y: double
- width: double
- height: double
- damage: int

+ getBounds(): Rectangle
+ getDamage(): int
+ afterInteraction: void

Scientist
- x: double
- y: double
- width: double
- height: double
- detectionWidth: double
- detectionHeight: double
- damage: int
- isSeekingPlayer: bool

+ getBounds(): Rectangle
+ getDetectionBounds: Rectangle
+ onDetection(): void
+ getDamage(): int
+ afterInteraction: void

Player
- x: double
- y: double
- width: double
- height: double
- health: int
- stamina: int
- score: int

+ getBounds(): rectangle
+ interactWithEntity(Entity entity): void

Potion
- x: double
- y: double
- width: double
- height: double

+ getBounds(): Rectangle
+ afterInteraction: void

Spikes
- x: double
- y: double
- width: double
- height: double

+ getBounds(): Rectangle
+ getDamage(): int
+ afterInteraction: void

PlayerInteractionChecker
- player: Player

+ PlayerInteractionChecker(): void
+ checkInteraction(Player player)
+ checkInteraction(Scientist scientist): void
+ checkInteraction(Spike spike): void
+ checkInteraction(Potion potion): void

Drawing
+ field: type

+ method(type): type

Maze
- nonPlayerEntities: Entity[]
- playerInteractionChecker: PlayerInteractionChecker

+ Maze(): void
+ checkInteractions: void
- createScientist():Scientist
- createSpike(): Spike
- createTile(int x, int y): void
- createWall(int x, int y, Direction direction): void

Classname
+ field: type

+ method(type): type

GameWindow
+ JFrame jframe

+ GameWindow( );

Tile
+ field: type

+ method(type): type

MazeObject
- x: int
- y: int

+ MazeObject(x, y): void

Wall
+ field: type

+ method(type): type

GameWindow
+ field: type

+ method(type): type

GraphicPanel
+ mouseInputs: private MouseInputs
+ game: private Game

+ GamePanel(Game): public
+ SetPanelSize( ): private void
+

Game
+ field: type

+ method(type): type

Main
+ main( )

+ field: type