Ship

#_pi: float
#_name: string
#_state: bool
#_health: int
#_battlefieldCentre_x: int
#_battlefieldCentre_y: int
#_startingCoordinates: Coordinates
#_currentCoordinates: Coordinates
#_ellipseSize: EllipseSize
#_colour: Colour

<<create>>-Ship()
<<create>>-Ship(name: string, radiusFromCentre: int, startingAngle: float, battlefieldCentre_x: int, battlefieldCentre_y: int, size_x: int, size_y: int)
<<create>>-Ship(name: string, radiusFromCentre: int, startingAngle: float, battlefieldCentre_x: int, battlefieldCentre_y: int, size_x: int, size_y: int, health: int)
+ReduceHealth(reduceAmount: int): void
+DestroyShip(): void
+get_pi(): float
+get_name(): string
+get_state(): bool
+get_health(): int
+get_startingRadiusFromCentre(): int
+get_startingAngleFromCentre(): float
+get_startingCoordinate_x(): int
+get_startingCoordinate_y(): int
+get_radiusFromCentre(): int
+get_angleFromCentre(): float
+get_coordinate_x(): int
+get_coordinate_y(): int
+get_sizeRadius_x(): int
+get_sizeRadius_y(): int
+get_colourRed(): int
+get_colourGreen(): int
+get_colourBlue(): int
+get_colourTransparency(): int

AlienMissile

<<create>>-AlienMissile()
<<create>>-AlienMissile(start_radius: int, start_angle: float, start_x: int, start_y: int)
+MoveContinuous(gameBattlefield: Battlefield): void

Missile

#_pi: float
#_state: bool
#_startCoordinates: Coordinates
#_currentCoordinates: Coordinates
#_ellipseSize: EllipseSize
#_colour: Colour

<<create>>-Missile()
<<create>>-Missile(start_radius: int, start_angle: float, start_x: int, start_y: int)
+set_state(newState: bool): void
+DestroyMissile(): void
+get_pi(): float
+get_state(): bool
+get_startingRadiusFromCentre(): int
+get_startingAngleFromCentre(): float
+get_startingCoordinate_x(): int
+get_startingCoordinate_y(): int
+get_radiusFromCentre(): int
+get_angleFromCentre(): float
+get_coordinate_x(): int
+get_coordinate_y(): int
+get_sizeRadius_x(): int
+get_sizeRadius_y(): int
+get_colourRed(): int
+get_colourGreen(): int
+get_colourBlue(): int
+get_colourTransparency(): int

PlayerMissile

-_crossCentre: bool

<<create>>-PlayerMissile()
<<create>>-PlayerMissile(gamePlayerShip: PlayerShip)
+MoveAlongCircles(gameBattlefield: Battlefield): void
+MoveContinuous(gameBattlefield: Battlefield): void

AlienShip

-_circleIterator: int

<<create>>-AlienShip()
<<create>>-AlienShip(name: string, radiusFromCentre: int, startingAngle: float, battlefieldCentre_x: int, battlefieldCentre_y: int, size_x: int, size_y: int, red: int, green: int, blue: int)
<<create>>-AlienShip(name: string, radiusFromCentre: int, startingAngle: float, battlefieldCentre_x: int, battlefieldCentre_y: int, size_x: int, size_y: int, red: int, green: int, blue: int, health: int)
+MoveClockwise(gameBattlefield: Battlefield): void
+MoveCounterClockwise(gameBattlefield: Battlefield): void
+get_circleIterator(): int

PlayerShip

-_numberOfLives: int
-_score: int

<<create>>-PlayerShip()
<<create>>-PlayerShip(name: string, radiusFromCentre: int, startingAngle: float, battlefieldCentre_x: int, battlefieldCentre_y: int, size_x: int, size_y: int)
+MoveClockwise(gameBattlefield: Battlefield): void
+MoveCounterClockwise(gameBattlefield: Battlefield): void
+AddScore(howMuch: int): void
+get_score(): int

<<CppStruct>> Line

+start_x: int
+start_y: int
+end_x: int
+end_y: int
+colour_red: int
+colour_green: int
+colour_blue: int
+colour_transparency: int

<<CppStruct>> Circle

+centre_x: int
+centre_y: int
+radius_x: int
+radius_y: int
+colour_red: int
+colour_green: int
+colour_blue: int
+colour_transparency: int

<<CppStruct>> Coordinates

+radiusFromCentre: int
+angleFromCentre: float
+x: int
+y: int

<<CppStruct>> Colour

+red: int
+green: int
+blue: int
+transparency: int

<<CppStruct>> EllipseSize

+radius_x: int
+radius_y: int

<<enumeration>> Keyboard

+key_none
+key_left
+key_right
+key_space
+key_s
+key_l
+key_enter
+key_esc
+key_r
+key_q
+key_1
+key_2
+key_3

<<enumeration>> Action

+noAction
+movePlayerLeft
+movePlayerRight
+playerShoot
+start
+pause
+resume
+restart
+save
+load
+quit
+settings_1
+settings_2
+settings_3

<<enumeration>> WinCondition

+PlayerDestroyed
+AllAliensDestroyed
+AliensOnOuterCircle

Figure 3 (cont.): UML diagram illustrating class structure of the solution