Java Term Project UML Diagram

Irene Cheung A01349998

<<enumeration>>
ColorEnum

+ TEAL: String
+ ORANGE: String
+ LIME: String
+ RED: String

<<Interface>>
Collision

+ move(Group, Circle): void
+ checkCollision(ImageView, Circle, Group): void

<>
Enemy

- colorChoices: colorEnum[]

 # createCircle(): Circle
+ move(Group, Circle): void
# generateEnemies(Group): void
+ checkCollision(ImageView, Circle, Group): void
 # setSpecificMovementDirection(Group, Circle): void
- convertColorEnumToJavaFx(colorEnum): Color
- generateRandomColor(): colorEnum
+ getColorChoices(): colourEnum[]
+ toString(): String
+ equals(Object): boolean
+ hashCode: int

VerticalEnemy

setSpecificMovementDirection(Group, Circle): void
+createCircle(): Circle

HorizontalEnemy

setSpecificMovementDirection(Group, Circle): void
+ createCircle(): Circle

GameLoop

- root: Group

+ run(): void
+ toString(): String
+ hashCode: int
+ equals

Run

+ app_width: int
+ app_height: int
# platypus: ImageView
# gameOver: boolean
# cyclesCount: int
# cyclesText: CycleText
# levelUpText: LevelUpText
+ speed: int

- startGameLoop(Group): void
+ start(Stage): void
+ main(String[]): void

Player

- player: Player
level: int

+ processKeyPress(KeyEvent): void
+ getInstance(): Player
+ toString(): String
+ getPlayer(): Player
+ getLevel(): int
+ setLevel(): int

<<Interface>>
PlayerBehavior

+ processKeyPress(KeyEvent): void

GameOverText

+ display(Group): void

LevelUpText

updateLevel(): void

CycleText

incrementCyclesCount(): void