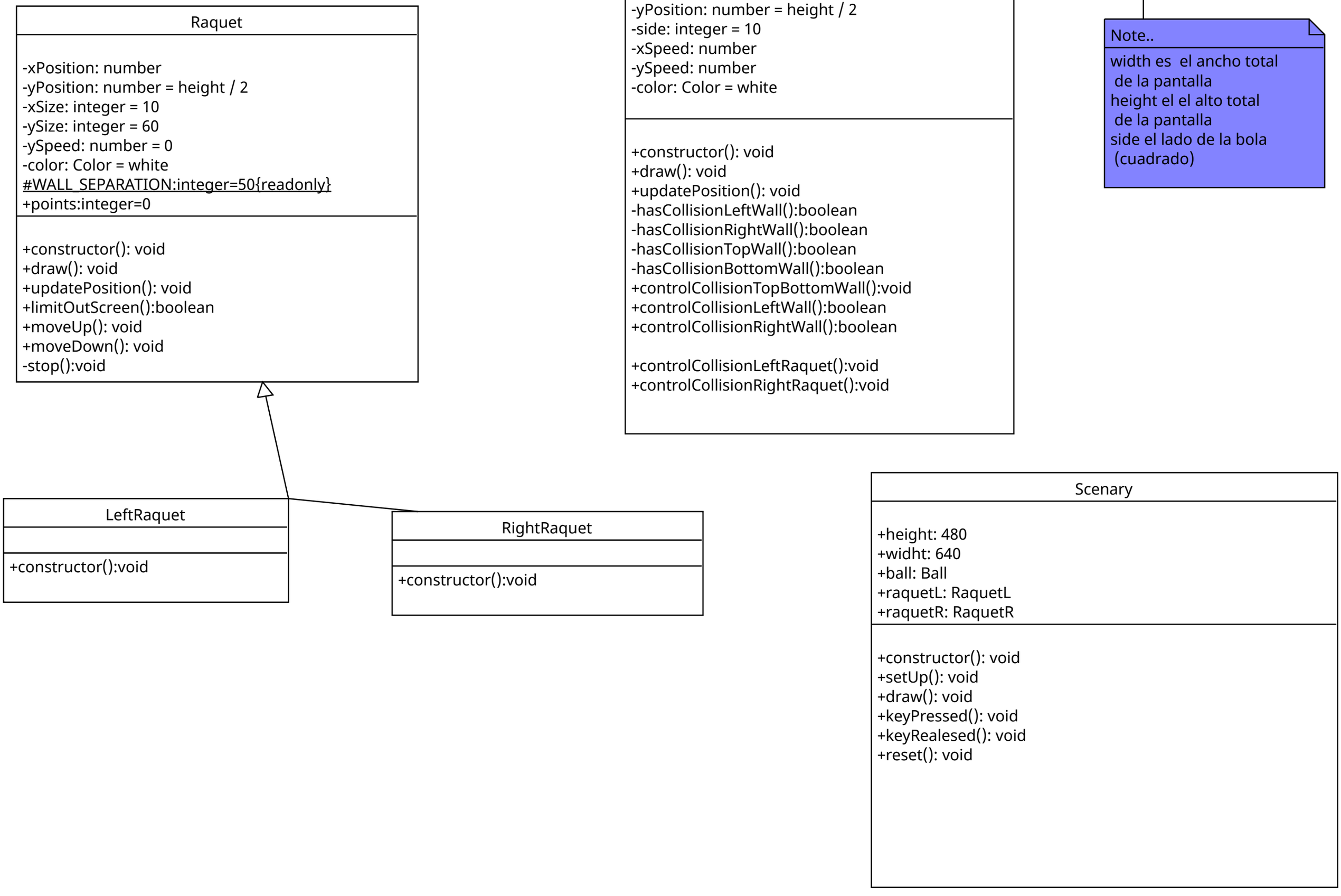
Raquet
-xPosition: number
-yPosition: number = height / 2
-xSize: integer = 10
-ySize: integer = 60
-ySpeed: number = 0
-color: Color = white
#WALL_SEPARATION:integer=50{readonly}
+points:integer=0
+constructor(): void
+draw(): void
+updatePosition(): void
+limitOutScreen():boolean
+moveUp(): void
+moveDown(): void
-stop():void
Ball
-xPosition: number = widht / 2
-yPosition: number = height / 2
-side: integer = 10
-xSpeed: number
-ySpeed: number
-color: Color = white
+constructor(): void
+draw(): void
+updatePosition(): void
-hasCollisionLeftWall():boolean
-hasCollisionRightWall():boolean
-hasCollisionTopWall():boolean
-hasCollisionBottomWall():boolean
+controlCollisionTopBottomWall():void
+controlCollisionLeftWall():boolean
+controlCollisionRightWall():boolean
+controlCollisionLeftRaquet():void
+controlCollisionRightRaquet():void
Note..
width es el ancho total
de la pantalla
height el el alto total
de la pantalla
side el lado de la bola
(cuadrado)
LeftRaquet
+constructor():void
RightRaquet
+constructor():void
Scenary
+height: 480
+widht: 640
+ball: Ball
+raquetL: RaquetL
+raquetR: RaquetR
+constructor(): void
+setUp(): void
+draw(): void
+keyPressed(): void
+keyRealesed(): void
+reset(): void