UML Diagram

Guy

+ moveLeft: boolean
+ moveDown: boolean
+ moveRight: boolean
+ moveUp: boolean
+ canMove: boolean
+ beKilled: boolean
+ SECONDS_BETWEEN_RENDERED: double
# x: int
# y: int
- timer: double
+ sprite: PImage
+ sprite1: PImage
+ sprite2: PImage
+ sprite3: PImage
+ sprite4: PImage
+ spriteDown1: PImage
+ spriteDown2: PImage
+ spriteDown3: PImage
+ spriteDown4: PImage
+ spriteRight1: PImage
+ spriteRight2: PImage
+ spriteRight3: PImage
+ spriteRight4: PImage
+ spriteLeft1: PImage
+ spriteLeft2: PImage
+ spriteLeft3: PImage
+ spriteLeft4: PImage
+ spriteUp1: PImage
+ spriteUp2: PImage
+ spriteUp3: PImage
+ spriteUp4: PImage

+ tick(app:App): void
+ direction(app:App): void
+ drawGuy(app:App): void
+ tick(app:App, sprite:PImage): void
+ getX(): int
+ getY(): int
+ canMove(app:App): boolean)

BombGuy

+ putBomb: boolean

+ direction(app:App): void
+ tick(app:App): void
+ reachGoal(app:App): void

Enemies

+ enemyMoveDown: boolean
+ enemyMoveUp: boolean
+ enemyMoveLeft: boolean
+ enemyMoveRight: boolean
+ canMoveDown: boolean
+ canMoveUp: boolean
+ canMoveLeft: boolean
+ canMoveRight: boolean
+ canMove: boolean
+ nextX: int
+ nextY: int
+ beKilled: boolean

+ enemiesCanMove(app:App): void
+ direction(app:App): void
+ removeEnemy(app:App): void
+ tick(app:App): void
+ killBombGuy(app:App): void
+ beKilled(app:App): void

YellowEnemies

+ turnDirection(): void
+ enemiesCanMove(app:App): void
+ direction(app:App): void
+ removeEnemy(app:App): void

RedEnemies

+ turnDirection(): void
+ enemiesCanMove(app:App): void
+ direction(app:App): void
+ removeEnemy(app:App): void

Bomb

# x: int
# y: int
+ top1Y: int
+ top2Y: int
+ bottom1Y: int
+ bottom2Y: int
+ left1X: int
+ left2X: int
+ ritht1X: int
+ right2X: int
- timer: double
+ period: double
+ frameStart: int
+ sprite: PImage
+ sprite1: PImage
+ sprite2: PImage
+ sprite3: PImage
+ sprite4: PImage
+ sprite5: PImage
+ sprite6: PImage
+ sprite7: PImage
+ sprite8: PImage
+ explosionCentreSprite: PImage
+ explosionBottomSprite: PImage
+ explosionLeftSprite: PImage
+ explosionRightSprite: PImage
+ explosionTopSprite: PImage
+ explosionHorizontalSprite: PImage
+ explosionVerticalSprite: PImage
+ explosion: boolean
+ top1Wall: boolean
+ top2Wall: boolean
+ bottom1Wall: boolean
+ bottom2Wall: boolean
+ left1Wall: boolean
+ left2Wall: boolean
+ right1Wall: boolean
+ right2Wall: boolean
+ topBrokenWal: boolean
+ bottomBrokenWall: boolean
+ leftBrokenWall: boolean
+ rightBrokenWall: boolean
+ explosion: boolean
+ removeBrokenWallList: ArrayList<Integer>
+ locationX: ArrayList<Integer>
+ locationY: ArrayList<Integer>

+ setSprite(app:App): void
+ haveWall(app:App): void
+ loadLocation(app:App): ArrayList<ArrayList>
+ rangeExplosion(app:App): void
+ drawBomb(app:App): void
+ draw(app:App, sprite:PImage): void
+ getX(): void
+ getY(): void

LoadConfig

+ emptySprite: PImage
+ solidSprite: PImage
+ brokenSprite: PImage
+ goalSprite: PImage
+ clockSprite: PImage
+ playerLogoSprite: PImage
+ down1RED: PImage
+ down2RED: PImage
+ down3RED: PImage
+ down4RED: PImage
+ up1RED: PImage
+ up2RED: PImage
+ up3RED: PImage
+ up4RED: PImage
+ left1RED: PImage
+ left2RED: PImage
+ left3RED: PImage
+ left4RED: PImage
+ right1RED: PImage
+ right2RED: PImage
+ right3RED: PImage
+ right4RED: PImage
+ down1YELLOW: PImage
+ down2YELLOW: PImage
+ down3YELLOW: PImage
+ down4YELLOW: PImage
+ up1YELLOW: PImage
+ up2YELLOW: PImage
+ up3YELLOW: PImage
+ up4YELLOW: PImage
+ left1YELLOW: PImage
+ left2YELLOW: PImage
+ left3YELLOW: PImage
+ left4YELLOW: PImage
+ right1YELLOW: PImage
+ right2YELLOW: PImage
+ right3YELLOW: PImage
+ right4YELLOW: PImage
+ bombsprite: PImage
+ bombsprite1: PImage
+ bombsprite2: PImage
+ bombsprite3: PImage
+ bombsprite4: PImage
+ bombsprite5: PImage
+ bombsprite6: PImage
+ bombsprite7: PImage
+ bombsprite8: PImage
+ explosionCentreSprite: PImage
+ explosionBottomSprite: PImage
+ explosionLeftSprite: PImage
+ explosionRightSprite: PImage
+ explosionTopSprite: PImage
+ explosionHorizontalSprite: PImage
+ explosionVerticalSprite: PImage
+ bombsprite7: PImage
+ bombsprite8: PImage
+ redEnemiesList: ArrayList<RedEnemies>
+ yellowEnemiesList: ArrayList<YellowEnemies>
+ brokenWallList: ArrayList<Integer>
+ solidWallList: ArrayList<Integer>
+ goalTileList: ArrayList<Integer>
+ bombList: ArrayList<Bomb>
+ time: int
+ level: int
+ lives: int
+ gameOver: boolean
+ loseLives: boolean
+ nextLevel: boolean

+ loadConfig(path:String): HashMap<String, String>
+ loadMap(path:String, configList:HashMap<String, String>): ArrayList<String>
+ locationBombGuy(mapList:ArrayList<String>, bombGuy:BombGuy): void
+ enemiesIndexRED(mapList:ArrayList<String>): ArrayList<Integer>
+ enemiesIndexYELLOW(mapList:ArrayList<String>): ArrayList<Integer>
+ drawMap(mapList:ArrayList<String>, app:App): void
+ drawScreen(stateBoolean:boolean, app:App): void
+ drawUI(app:App, level:int): void
+ createEnemiesRED(app:App): void
+ createEnemiesYELLOW(app:App): void
+ loadBrokenWall(app:App): void
+ loadSolidWall(app:App): void
+ loadGoalTile(app:App): void
+ setSpriteBomb(app:App): void
+ setSpriteEnemies(app:App): void
+ setSpriteBasic(app:App): void
+ createBombList(app:App): void
+ clearInformation(app:App): void
+ gameOver(app:App): void
+ loadLevel1(app:App): void
+ loadLevel2(app:App): void
+ loadLevel3(app:App): void
+ drawAll(app:App): void