Dungeon

Venkata Rakesh Kumar Molakala | December 10, 2021

<<interface>> ReadOnlyDungeon

+List<String> getPlayerLocationStatus()
+List<String> getInitialGameStatus
+Location getPlayerCurrentLocation()
+Location getStartLocation()
+Location getEndLocation()
+int getInterconnectivityDegree()
+double getItemsPercentage()
+List<String> getPlayerDescription()
+List<Treasure> getPlayerTreasure()
+boolean isGameOver()
+boolean isPlayerAlive()
+int getRows()
+int getColumns()
+int getDifficulty()
+boolean isWrapping()
+Location[][] get2dDungeon()
+List<Integer> getVisited()
+int getPlayerHealth()
+Stench stenchCalculation(Location l)

<<interface>> ReadOnlyDungeonObstacle

+boolean isPitNearBy()
+int getThiefLocationId()
+int getMovingMonsterLocationId()
+int getMovingMonsterHealth()
+boolean isMovingMonsterAlive()

<<interface>> GuiFeatures

+void createDungeon(WrappingStyle wrap, int interconnectivity, int rows, int columns, double itemsPercentage, int difficulty)
+void startGame()
+void restart()
+void goToHome()
+void move(Direction d)
+void pickUpTreasure()
+void pickUpArrows()
+void shoot(Direction d, int caves)
+void exitProgram()
+void movePlayerTo(int locationId)
+void fightMonster()
+void reuse()

<<interface>> IView

+void makeVisible()
+void setFeaturesForHome(GuiFeatures f)
+void setFeaturesForGame(GuiFeatures f)
+void setModel(ReadOnlyDungeon d)
+void turnPageToHome()
+void turnPageToGame()
+void restartGameBoard()
+void displayError()
+void quitGame()

<<interface>> Location

+String getLocationId()
+boolean getIsTunnel()
+List<Treasure> getTreasureList()
+Map<Direction, Location> getNeighbours()
+Otyugh getMonster()
+int getArrowCount()

<<interface>> DungeonObstacle

+void fight()
+DungeonObstacle getInitialModel()
+void setRandom()
+List<Integer> getPrevRandom()

<<interface>> TextDungeonController

+void playGame(Dungeon m)

JFrameView

-homePanel: HomeBoard
-model: ReadOnlyDungeon
-gameboard: GameBoard
-f: GuiFeatures

+JFrameView()

<<interface>> Dungeon

+void movePlayerTo(Direction direction)
+void pickUpArrows()
+void pickUpTreasures()
+void shoot(Direction d, int s)
+void moveToLocation(int locationId)

TextDungeonConsoleController

-output: Appendable
-input: Scanner

+void playGame(Dungeon m)

GuiDungeonController

-view: IView
-model: Dungeon

+GuiDungeonController()
+GuiDungeonController(Dungeon m, IView v)

<<interface>> PlayerInterface

+List<Treasure> getTreasureCollected()
+List<String> getPlayerDescription()
+int getArrows()
+int getHealth()
+void reduceHealth()
+void addTreasure(TreasureType treasure)
+void updateArrows(int newArrowsCount)
+void lootTreasure()
+boolean isAlive()

LocationC

-locationId: String
-neighbours: Map<Direction, Location>
-treasureList: List<Treasure>
-int arrowCount
-Otyugh monster

+Location()
+String getLocationId()
+boolean getIsTunnel()
+List<Treasure> getTreasureList()
~void addTreasure()
~void removeTreasure()
+Map<Direction, Location> getNeighbours()
+int getArrowCount()
+Otyugh getMonster()
~void uodateArrowsCount()
~void addMonster()
~void killMonster()

«enumeration» Direction

NORTH
EAST
SOUTH
WEST

Maze

-player: PlayerInterface
-wrappingStyle: WrappingStyle
-interconnectivityDegree: int
-Location[][] dungeonMaze
-rand : RandomInterface
-gameOver: Boolean
-double: itemsPercentage
-Location: playerLocation
-List<String> playerLocationStatus
-start: Location
-end: Location
-caveId List: List<Integer>
-noOfMonsters: int
-rows: int
-columns: int

+Maze(WrappingStyle w, int interconnectivityDegree, int xLength, int yLength, double itemsPercentage, RandomInterface rand, int noOfMonsters)
-void kruskal()
-void setStartEnd()
-void setConnections()
-void addTreasures()
-void addMonsters()
-void addArrows()
-Stench stenchCalculation()

MazeWithObstacles

-isPlayerTurn: boolean
-IsMovingMonsterAlive: boolean
-prev: List<Integer>
-movingMonster: Monster
-thiefLocation: Location

+MazeWithObstacles(WrappingStyle w, int interconnectivityDegree, int xLength, int yLength, double itemsPercentage, RandomInterface rand, int noOfMonsters)
~void addMonsters(Direction d)
~void locationStatus()
-void pitDetection(Location l)
+void movePlayerTo(Direction d)

Player

-arrows: int
-treasureList: List<Treasure>
-health: int

+Player()

«enumeration» WrappingStyle

WRAPPING
NONWRAPPING

«enumeration» TreasureType

DIAMONDS
RUBIES
SAPPHIRES

<<interface>> RandomInterface

int getRandomNumber(int min, int max)

UnionFind

-parent: int[]

~UnionFind()
~void union(int x, int y)
~int find(int x)

<<interface>> Monster

+int getArrowHit()
+void recordHit()
+int getMonsterId()
+Location getLocation()
+void setLocation(Location l)

RandomNumber

-rand: Random

int getRandomNumber(int min, int max)

RandomPredictor

-preRand: LinkedList<Integer>

int getRandomNumber(int min, int max)

«enumeration» Stench

MOREPUNGENT
LESSPUNGENT
NOSTENCH

Otyugh

-arrowHit: int
-monsterId: int

+Otyugh(int monsterId, Location l)

Otyugh

-arrowHit: int
-monsterId: int
-location: Location

MovingMonster(int monsterId, Location location)