COMP90041, Sem1, 2023: Final Project
UNIFIED MODELLING LANGUAGE (UML)

commandInput

+ argsGet(args: String[], path: String): String

+ printHelp(): Void

RescueBot

- showScenarios : bool

- option : String

- mode : String

- scenariosMatrix : ArrayList<Scenario>

- ordered : ArrayList< ArrayList<<String>>

...

+ main(args:String[]): void

- inputedScenarios(rescBot:RescueBot, readFile:FileReading, inLine:commandInput, args:String[]): void

- mainMenuOptions(rescBot:RescueBot, statistics:GetStatistics, inLine:commandInput, args:String[],write:FileWriting, audit:Audit): void

- instanciateScenarios(rescBot:RescueBot, newScen:CreateScenarios): void

- deployBots(rescBot:RescueBot, statistics:GetStatistics, write:FileWriting): void

+ simulateLocation(rescBot:RescueBot, statistics:GetStatistics): void

+ showStatistics(rescBot:RescueBot, statistics:GetStatistics, write:FileWriting): void

...

→ Use → **commandInput**

FileWriting

- outputStream : PrintWriter

+ setWriteFile(filePath:String) : Void

- PrintWriter newFileOrExit(filePath :String) : Void

+ writeLines(inputToWrite: String) : Void

← Use (from RescueBot)

FileReading

- inputStream : Scanner

- scen : Scenario

- loc : Lcoation

...

+ readFile(filename:String) : Scanner

+ printWelcome(): Void

+ readScenarios(filename:String): ArrayList<Scenario>

...

← Use (from RescueBot)

Audit

- inputStream : Scanner

- logPath :String

- fileGood :boolean

+ getFile(rescBot:RescueBot, inLine:commandInput,args:String[], input:Scanner) : Void

+ auditFile(rescBot:RescueBot): Void

...

← Implementation (from RescueBot)

CreateScenarios

+ scenarios(scenariosMatrix:ArrayList<Scenario>, totScenarios: int) : Void

- createScenario(): String

- createLocation(countLoc: int): String

...

→ Implementation (from RescueBot)

GetStatistics

- getSelectedFeatures(rescBot:RescueBot , location:Location, selectedFeatures: ArrayList<String> , rescuedAge:ArrayList<Integer>) : Void

- getGeneralFeatures(rescBot: RescueBot, processedScenarios: ArrayList<Scenario>, generalFeatures: ArrayList<String>) : Void

+ calculateStatistics(rescBot:RescueBot, processedScenarios :ArrayList<Scenario> , generalFeatures: ArrayList<String>, selectedFeatures:ArrayList<String> , uniqueFeatures:LinkedHashSet<String> , featPropResc:ArrayList<String>) : Void

...

Implementation (from Audit)

→ Use → Scenario

Scenario

- name : String

- locationArray : ArrayList<Location>

+ Scenario(name: String)

+ add(location:Location): Void

+ getLocations(): ArrayList< Location >

...

Implementation (from RescueBot)

← Use (from CreateScenarios)

Agregation

Location

- loc : String

- tresspasing : String

- characters : ArrayList<Character>

+ Location(loc:String, trespassing:String)

+ add(character:Character): Void

+ getCharacters(): ArrayList<Character>

...

Agregation

Human

- profession : String

- gender : String

- pregnant : String

+ Human(bodyType:String,age:int,specie:String, gender: String, pet: String)

+ describe(): String

+ tipeChar(): String

+ getFeatures(): ArrayList<String>

...

— Extends → Character

Character

- bodyType : String

- age : int

- gender : String

+ Character(bodyType:String, age:int, gender:String)

+ describe(): String

+ tipeChar(): String

-+getFeatures(): ArrayList<String>

...

← Extends — Animal

Animal

- specie : String

- pet : String

+ Animal(bodyType:String,age:int,specie:String, gender: String, pet: String)

+ describe(): String

+ tipeChar(): String

+ getFeatures(): ArrayList<String>

...