

C

App

model : GameModel

view : GameView

controller : GameController

App()

main(args : String[]) : void

Model

C

GameModel

Tamagotchi : tamagotchi

gameManager : gameManager

GameModel()

Tamagotchi

C

Food

id : int

name : String

Food(id : int, name : String)

C

Tamagotchi

name : String

breed : String

favoriteFoods : String[]

health : Need

alive : boolean

mood : String

energy : Need

hunger : Need

cleanliness : Need

happiness : Need

CRITICAL\_THRESHOLD : integer

needs : Need[] = { energy, hunger, cleanliness, happiness }

Tamagotchi(String name) : Tamagotchi

Tamagotchi() : Tamagotchi

scanCritical() : Boolean

depleteNeeds() : void

setDefaultStats() : void

getName() : String

setName(String newName) : void

getBreed() : String

setBreed(newBreed : int)void

evalMood() : void

getMood() : String

setMood(newMood : String) : void

checkHealth() : void

kill() : void

hurt(int damage) : void

heal(int healValue) : void

setHealth(int newHealth) : void

boolean getAlive()

void setAlive(boolean alive)

C

Need

name : String

value : int

factor : float

critical : boolean

Need(name : String, value : int, factor : float, critical : boolean)

calcDepletion(time : float) : void

setVal(value : int) : void

setFactor(factor : float) : void

getVal() : int

GameManager

C

GameManager

saveLocation : String = "../saveData/"

saver : GameSaver

loader : GameLoader

timer : GameTimer

GameManager()

C

GameLoader

GameLoader(game : GameModel) : void

C

GameSaver

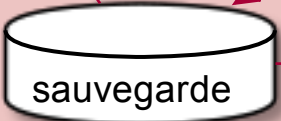
GameSaver(game : GameModel) : void

saveGame(model.tamagotchi) : void

writeFile

C

GameTimer



sauvegarde

Fichier de sauvegarde stocké sur le disque

I

Serializable