**Santiago Hernandez Saavedra A00382538**

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc001  Create a Player | Class VideoGame | CreatePlayer(NamePlayer : String, nickname : String))  The method creates the object, with data that the user entered in ui |
| Class VideoGame | searchPlayer(nickname : String)  the method checks if the nickname entered by the user already exists |
| Class VideoGame | AvailablePlayer()  check if there are players available in the array |
|  | Class Player | Player(nickname : String, name : String, score : int, numberLifes : String)  receive the parameters of the new player |
|  | Class SystemApp  Package ui | createPlayer()  call all methods to create the player |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc002  Register Level |  |  |
| Class Level | Level(id;int, points\_level : int)  receive the parameters of the new player |
| Class SystemApp  Package ui | registerLevel() : void  create the level |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc003  Register enemy to a level | Class VideoGame | generatePosition(x : int)  generates the position according to the resolution |
|  |  |
| Class VideoGame | addenemy(name : String, typeEnemy : String, damage : int;score:int, level : int, positionX : int, positionY : int)  takes the parameters entered in the main by the user, to register the enemy according to the wanted level |
|  | Class Level | addEnemy(name : String, typeEnemy : String, damage : int;score:int, positionX : int, positionY : int)  create the enemy object |
|  | Class Level | searchEnemy(name : String)  see if that enemy already exists |
|  | Class Level | availableEnemy()  see if there are slots available to create enemies |
|  | Class Enemy | Enemy(name : String, typeEnemy : String, damage : int;score:int, positionX : int, positionY : int)  receive the parameters of the new Enemy |
|  | Class SystemApp  Package ui | registerEnemy()  call all methots for create the enemy |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc004  Register treasure to a level | Class VideoGame | addTresuare(name : String, url : String, score : int, level : int:positionX : int, positionY : int, numberTreasure : int)  the method collects the data collected by the user and calls the other methods |
|  |  |
| Class VideoGame | generatePosition(x : int)  generates the position according to the resolution |
|  | Class Level | addTreasure(name : String, url : String, score : int, positionX : int, positionY : int, numberTreasure : int)  create the Treasure object |
|  | Class Level | availableTreasure()  see if there are slots available to create Treasure |
|  | Class Treasure | Treasure(name : String, url : String, score : int, positionX : int, positionY : int, numberTreasure : int)  receive the parameters of the new Treasure |
|  | Class SystemApp  Package ui | registerTreasure()  call all methos for create Treasure |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc005  Modify a player's score. | Class VideoGame | changeScorePlayer(scoreNew : int, nickname : String)  collects the data entered by the user to call the other methods, to change the score of a player |
| Class Player | getNickname()  get player id |
| Class Player | getScore()  get player score |
|  | Class VideoGame | searchPlayer(nickname : String)  get the player |
|  | Class Player | setScore(score : int)  change player score |
|  | Class SystemApp  Package ui | changePointPlayer()  call all methots |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc006  Increase level for a player, in case you cannot increase the level, you must inform the user what score is required to increase. | Class VideoGame | increaseLevelPlayer(nickname : String)  takes as parameter the nickname of the entered player to compare the player's score with that of the next level, to then send a message about whether or not the level could be increased |
| Class VideoGame | searchPlayer(nickname : String)  get the player  looks for the entered player and returns his ob |
| Class Player | getNickname()  get player id |
|  | Class Player | getScore()  get player score |
|  | Class Level | getPoints\_level()  get level points |
|  | Class Level | getId()  get level id |
|  | Class Player | setLevel(level : int)  change player level |
|  | Class SystemApp  Package ui | increaseLevel()  call all methots |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc007  Report the treasures and enemies (separated by commas) of a level given by the user | Class VideoGame | allEnemy(level : int)  calls the level entered by the user and shows all enemies |
| Class VideoGame | allTreasure(level : int)  calls the level entered by the user and shows all Treasure |
| Class Level | getEnemy()  get all the names of the registered enemies and put them together in a string |
|  | Class Level | getTreasure()  get all the names of the registered Treasures and put them together in a string |
|  | Class Level | countTreasureRepit()  Count how many treasures is repeated |
|  | Class Enemy | getName()  get enemy name |
|  | Class Treasure | getName()  get Treasure name |
|  | Class SystemApp  Package ui | amountTreasureAndEnemy()  call all methots  and ask the user for the level |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc008  Report the amount found of a treasure at all levels | Class VideoGame | amountTreasure(nameTresure : String  I save the name of the treasure that the user wants to search for, to then call the other methods to see how many treasures of that name are in the level |
| Class level | countTreasure(nameTreasure : String)  count how many treasures of that name there are |
| Class Treasure | getName()  get name Treasure |
|  | Class Treasure | getNumberTreasure()  get number of one treasures |
|  | Class SystemApp  Package ui | countTreasure()  call methots , and ask for the name of the treasure |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc009  Report the amount found of a type of enemy in all level | Class Videogame | amountTypeEnemy(enemyType : String)  collects the type entered by the user and calls a method to search the levels |
| Class level | countEnemy(enemyType : String)  Search the level for how many enemies there are of that type |
| Class Enemy | getTypeEnemy()  get enemy type |
|  | Class SystemApp  Package ui | countTypeEnemy()  call methots , and ask for the type Treasure |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc0010  Report the most repeated treasure in all levels | Class VideoGame | mostTresure()  method send the parameter to a method that returns an array where the names of the treasures are and compares to say which is the most repeated treasure |
| Class level | mostTresure()  will take the name of each treasure item and add it to a list |
| Class level | countTreasureRepit()  Count how many treasures is repeated |
| ClassTreasure | getName()  get Treasure name |
|  | ClassTreasure | getNumberTreasure()  get number of one treasure |
|  | Class SystemApp  Package ui | mostTreasure()  call all methots |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc0011  Report the enemy that gives the highest score and the level where it is located. | Class VideoGame | enemymostscore()  searches each level for the enemy with the most points and stores them in a variable to identify their level, and then compares them, to get the enemy with the most points |
| Class Level | enemyscore()  search in the level which is the enemy with the highest score |
| Class level | enemyNameScore(name : String)  receives the name of a treasure, looks for it and returns its score |
|  | Class Enemy | getName()  get Enemy name |
|  | Class Enemy | getScore()  get enemy score |
|  | Class SystemApp  Package ui | Enemymostscore()  Call alls methots |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc0012  Report the number of consonants found in the names of the enemies in the game. | Class VideoGame | amountConsonants()  get the consonants of the names of the enemies in each level and then add them |
| Class Enemy | getName()  get Enemy name |
| Class level | consonantsEnemy()  count how many consonants are in each enemy |
|  | Class SystemApp  Package ui | consonantsEnemy()  call all methods |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc0013  Inform the top 5 of the players according to the score. | Class VideoGame | topPlayer()  calls each registered player their name and score and compares their scores, showing the top 5 |
| Class Player | getScore()  get player Score |
| Class Player | getNickname()  get player nickname |
|  | Class SystemApp  Package ui | topPlayer()  call all methods |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| **Ejemplo:**  ReqFunc0014  . ask the user what is the resolution on which the game will run | Class VideoGame | VideoGame(x : int, y : int)  create the videoGame object |
| Class SystemApp  Package ui | askResolution()  Query the resolution of the game and call the method to create the object |