**Operation Contract**

****

**Team number : 8**

**Course : Software Engineering**

|  |  |
| --- | --- |
| Operation | createNewGame(numOfUser : integer, numOfGamePiece: integer) |
| Cross Reference | Use Cases : Yut Management, Appoint the number of user and piece |
| Preconditions | Before this operation starts, the yut game doesn’t also start, either. |
| Postconditions | Yut main stage was executed, Player number and Game piece for each player of gameManager was initialized as numOfUser and numOfGamePiece. And then first player got the turn to throw the yut |

|  |  |
| --- | --- |
| Operation | throwYut() |
| Cross Reference | Use Cases : Throw a Yut |
| Preconditions | Player is assigned to throw a yut by gameManager |
| Postconditions | User pushed the throw butten and called the throwYut() function of gameManager and then, gameManager called the Yut class and it created the yut value and stored it into Yut.Value. |

|  |  |
| --- | --- |
| Operation | requestMovement(destination:integer, node: Node) |
| Cross Reference | Use Cases : Yut Management |
| Preconditions | This case is divided into two cases. First is the new piece appointment and second is the existing piece in the game board move from current node to the node where user want to move |
| Postconditions | User threw the destination and node object to Movement System. Movement System object got the value and then stored it into each value of the object |

|  |  |
| --- | --- |
| Operation | moveGamePiece(destination:integer, node:Node) |
| Cross Reference | Use Cases : Piece Management, Movement of the new Piece |
| Preconditions | Movement System had the two value |
| Postconditions | Game Manager put the two value and request that the current node changed the gamepiece-On state from TRUE to FALSE. After this execution, the destination node was executed vice versa. If a game piece goaled in into start point, the piece goal-in flag changed to TRUE |

|  |  |
| --- | --- |
| Operation | movementExcept() |
| Cross Reference | Use Cases : Piece Management |
| Preconditions | User appoint a wrong node to move the piece. |
| Postconditions | Movement System alarmed the message “This game piece shouldn’t move to the node.” to player. |

|  |  |
| --- | --- |
| Operation | returnWinFlag(Flag:boolean, UserID:integer) |
| Cross Reference | Use Cases : Express the result of game |
| Preconditions | Every gamePiece of one player’s goal-in flag is TRUE. |
| Postconditions | Result system called the RequestFinishGame(UserID) |

|  |  |
| --- | --- |
| Operation | RequestFinishGame(UserID:integer) |
| Cross Reference | Use Cases : Express the result of game |
| Preconditions | Result System call this operation |
| Postconditions | Game finished and printed the winner player number out |