```
                                                                                              ┌──────────────┐
┌──────────────┐              ┌──────────────┐              ┌──────────────┐                  │
│: ScrabbleSystem│            │   : Board    │              │   : Move     │                  │
└──────────────┘              └──────────────┘              └──────────────┘                  └

playMove(Move move)
●───────────────────┐
                    │ isValid(Move move)
                    │◄──┐
                    │   │
                    │

      ┌──── [is the first move]
      │ if   isOnStart(Move move,Square star)
      │     ├──────────────────────────────►│
      │     │           isContained(Square star)
      │     │         ├──────────────────────────────►│
      │     │   return │                              │
      │     │◄─ ─ ─ ─ ─ ┤                              │

                checkNear(Move move)
            ├──────────────────────────────►│
            │              getSquare()       │
            │         ├──────────────────────────────►│
            │         │    return squares              │
            │         │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┤
            │         ┌──── loop                                    isO
            │         │   ├──────────────────────────────────────────────►
            │         │   │                 retur
            │         │   │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─
            │         │   │  [!sign] contain(squares)
            │         │   ├──────────────────────────────►│
            │  return │   │        return                  │
            │◄─ ─ ─ ─ ┤   │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┤

       addSpecialTile(Move move)
   ├──────────────────────────────►│
   │              getSpecialTile()   │
   │         ├──────────────────────────────►│
   │         │   return specialTile           │
   │         │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┤
   │         │        getSpecialTileSquare()   │
   │         ├──────────────────────────────►│
   │         │      return square              │
   │         │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┤
   │         │                                         setSpecia
   │         ├──────────────────────────────────────────────►

      activeSpecialTile(Move move)
   ├──────────────────────────────►│
   │               getSquare()       │
   │         ├──────────────────────────────►│
   │         │     return squares              │
   │         │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┤
   │         ┌──── loop  [squares.size()]
   │         │                                          special =
   │         │   ├──────────────────────────────────────────────►
   │         │   ┌──── if  [special]
   │         │   │                                      getS
   │         │   │   ├──────────────────────────────────────────►
   │         │   │   │              return
   │         │   │   │◄─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─
   │         │   │   ├──────────────────────────────────────────►
   │         │   │   │                                  remove
   │         │   │   ├──────────────────────────────────────────►

getMoveScore(Move move, Player player)
├──────────────────────────────►│
                                 │◄──┐ makeWords(Move move)
```

Sequence diagram with lifelines:

- : Word
- : Tile
- : Square
- : SpecialTile
- : Ti...

Messages:
- occupied()
- return sign
- alTile(specialTile)
- hasSpecialTile()
- pecialTile()
- a specialTile
- makeSpecialEffect(System system)
- eSpecialTile()

imerTile	: Player	:LetterBag	: TurnControl

makeWords(Move move)

getTiles()

return tiles

getSquare()

return squares

loop [TilesSize]

loop

loop [isOccupied]

squareTemp = getSquare(x,y)

isO

containSquare(squareTemp)

return

create（words）

loop

return word

loop [wordsListSize]

singleValue = getValue()

Occupied()

return

setValue()

[wordsSize]

tileValue= getValue

addValue(tileValue)

flag = isOccupied()

if   [!flag]

timer = hasTilesTimer()

changeWordValue(word)

clearAllWords()

setLastScore(0)

addScore(singleValue)

addLastScore(singleValue)

addWord(word)

addBoardTiles(Move move)

tiles = getTiles()

squares = getSquares()

loop [squareSize]

add

over = isGameOver()

outTilesSize = getTilesSize()

alt [over== false]

outTilesSize = getTilesSize()

runGame()

[over == true]

getWinner()

loop [playerNum]

dTiles(tile)

getPlayerTilesSize()

enough = hasEnoughTiles(outTileSize)

time = getContinuePassTime()

rTiles = getRandomTiles(outTilesSize)

addTiles(rTiles)

setMove(move)

clearLastRandomTiles()

addLastRandomTiles(rTiles)

updateOrder()

score = getScore()

bagScore = getBagScore()