Response

Domain Model:

I changed each arrow to the line segment without arrow.

I added cost to the special tile.

I added eggiest to each special tile.

I added a CurrentMove as I also did in object model, which handles the moves from the current player's round.

I updated my special tile, called RevengeTile now, because this is more creative than the old one. When the current player triggers the revenge tile, it will remove all the special tiles of the player that are already placed on the board.(not influence the holding special tiles).

System Sequence Diagram:

I deleted the interactions that the game would do to itself. (eg: countPass, nextPlayer, compareWinner)

I changed the methods that need to be captured and sent to the game. (eg: if any button is pressed)

I changed the loop specification as English, not implementation code.

Interaction Validate:

I added a class of CurrentMove, and assigned most of the check, sort and modification work to the CurrentMove class instance.

I added a method called checkFirstWordcol(game) inside CurrentMove class, to check if the first word is valid.

Interaction Move:

I added a CurrentMove class instance and update my design. I don't need to tell each kind of bonus square, just first execute letterBonus() then wordBonus(). Also, I don't need to tell each kind of special tiles, just execute the effect() method. Inside the effect(), it would do a lot of things that I showed on the diagram.

Object Model:

I made a new class called CurrentMove, and a class called ScoreWord. Inside ScoreWord, there are a pair consists of a square and a score, the score is responsible for the change of scored due to letter bonus, negative tiles, and etc., I do not want the score inside square to store that changed score. Inside CurrentMove, there are three instance variables, currentSquares is a list of squares, which are the squares the player placed at that round; scoreWordSquares is a List<List<ScoreWord>>, it stores all possible valid (or not) words, a

List<ScoreWord> consists a word; newWords is a List<String> corresponding to the scoreWordSquares, newWords is responsible for checking valid from dictionary.

I use WordTile the same as LetterTile (not change the name, but the same).

Explanation: I already made a SpecialBag, but I forgot to delete the documentation. From the object model, I have a class called SpecialBag as the special store.

I deleted bonusFlag and changed to two integer, letterBonus and word Bonus. They are default as 1. They are multipliers that applies to all squares. If we call getLetterBonus() and getWordBonus() from CurrentMove class, they would automatically calculate the bonus points.

Explanation: Squares in Board are all squares in the game, which is the number of 225. All other three lists are inside CurrentMove and are responsible for temporary moves.