**WORDY GAME**

The implementation of the Wordy Game is focused on below aspects

1. The Random Board Generator of Uppercase Alphabets
2. Incorporation of the ‘Dictionary’ file of Words for validation.
3. Basic WordValidator Implementation : It takes care of validating a word checking if its valid word present in Dictionary or not.
4. **WordInBoardValidator** : Below checks are handled –
   1. The Word is from the Board
   2. The Word is a valid Dictionary word
   3. The Word is formed by adjoining Phrases (Diagonally / Vertically / Horizontally)

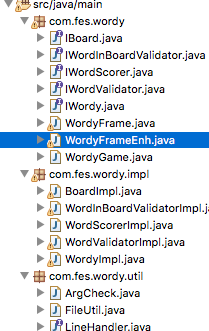
The Word Scorer Implementation that was provided takes care of Scoring word based on the length of the word. Any word length, if less than 3, score is Zero. Else Score is as defined in the requirements.

(In WordyImpl.java, we can choose which Validator to use at line nums – 83)

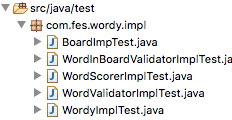
**CLASS STRUCTURE**

I have used Eclipse IDE for writing the Implementation Classes & JUnit test classes.

The main Implementators are under package – ‘com.fes.wordy.impl’



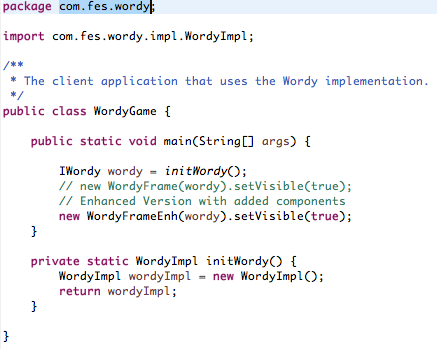
**The Test Classes are written for the Implementators and below is the list of TestClasses :**



The main Application class is “**com.fes.wordy/WordyGame**”.

The GUI components Layout is handled by the class – “com.fes.wordy/WordyFrame”

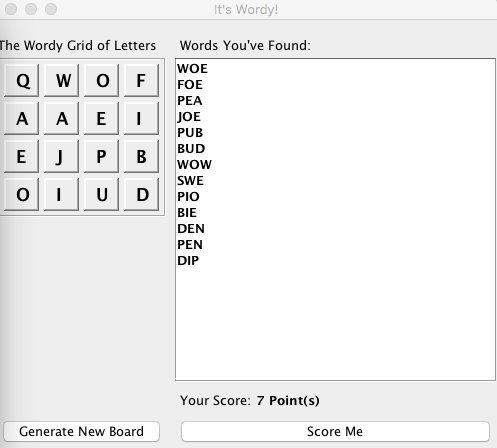
I have done another version of the Frame class and added few components. (as an extra attempt) , for enabling user to type into the WordArea by just clicking the Cells on the Board, Also a ‘RTN’ and ‘DEL’ key panel is provided to help User in editing. And ‘Clear’ button resets the Panel Contents and resets the ‘Score’. We can change WordyImpl to use WordyFrameEnh.java, if we would like to see its functionality.



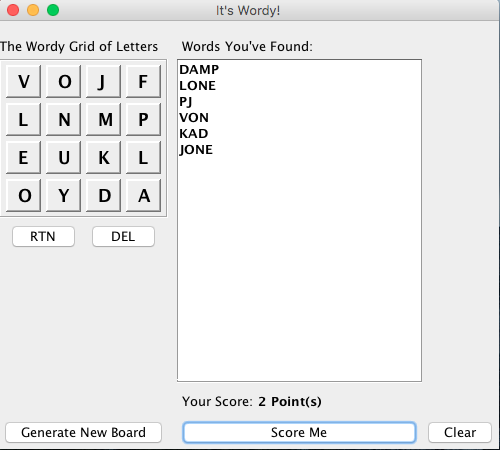
The ‘Generate Board’ Button generates the new Board on every invocation and a new Board of Cells is presented to the User to explore.

**SCREEN SHOTS**

**WordyFrame**

****

**WordyFrameEnh**

****