

## How our OOD will work

1. How we keep track of all the current status after a user enters a move
  - After the user enters a move (e.g. b2), we capture the input as a String, first validate the user input and if it is invalid the system would prompt the user to enter again. Then we separate the first character and the rest of the String into a String array (String[]).
  - With the String array we can use a method in the Board class to locate the specific cell in the grid and get that Cell instance. Since Board class contains a 2-D Cell array (Cell[][] grid), we can first translate the String array into integer coordinate and use the coordinate to locate the cell in the 2-D Cell array (grid[row][col])
  - After getting the Cell instance that is hit by the user, we set the cell's field isHit to "true" (it's set to "false" when initiated). After that, we get the Tank object from that Cell instance (there is a Tank field in the Cell class to keep track of which tank the Cell instance belongs to). If the Tank object we equals to "null", that means the cell doesn't belong to any Tank and the user misses.
  - After getting the Tank object, we use the "eliminateCell()" method in Tank class to reduce a tank cell, the Tank class has an ArrayList keeping track of all the cells it has. The damage field of the tank is then recalculated by how many remaining cells it has.
2. How do we draw the board each time?
  - In the Board class we have a 2-D array (Cell[][] grid) which contains all the 100 cells. Also we assign each tank a letter starting from A, which is used in cheat mode.
  - In Cell class, there is a Tank field specifying which Tank instance the Cell instance belongs to; There is also an isHit field which indicate if the Cell instance get hit by user.
  - In normal mode (non-cheating), for each Cell instance in the grid
    - If (!cell.isHit) which means the Cell instance is not hit by user, we will draw ~ to represent it (fog).
    - if (cell.isHit() && cell.getTank() == null) which means the Cell instance is not a tank cell and was hit by the user, so we draw ' ' (space) to represent this kind of Cell instance.
    - If (cell.isHit() && cell.getTank() != null) which means that the Cell instance is a tank cell and was hit by the user, we draw X for this Cell instance on the board.
  - In cheat mode, for each Cell instance:
    - If (!cell.isHit && cell.getTank() != null) which means the Cell instance is not hit by user and is a tank cell, we will draw the letter of the tank for this Cell instance (cell.getTank().getTankLetter()).
    - if (!cell.isHit() && cell.getTank() == null) which means the Cell instance is not a tank cell and wasn't hit by the user, so we draw '.' (a dot) to represent this kind of Cell instance.
    - If (cell.isHit() && cell.getTank() != null) which means that the Cell instance is a tank cell and was hit by the user, we draw X for this Cell instance on the board.

- If (cell.isHit() && cell.getTank() == null) which means the Cell instance is not a tank cell and was hit by the user, we draw ' ' (space) to represent it.