Clear separation of the UI and business logic will be very good approach in design as if we would need to change the UI it will be easier for us to do that. There would be no cyclic relation and UI uses logic while logic knows nothing about the UI.

To generation of the board would be broken into several steps. When the map will be created all of its entities would be assigned to null. When we will begin generation of the tank we will start at any random position and will fill up the neighbor cells with ‘X’ objects. When generation was complete and all of the coordinates were saved in the brand new tank object we will go around possible borders of the tank and assign entities to ‘.’ And this will be used for handling conflicts. Then we can proceed to generating second tank. As soon as we see ‘.’ value we prohibit generation in that way. If that will be not done than tanks would be adjacent to each other or may overlap. When all the tanks were placed it is possible that some entities are still null. To finish up generation replace all null values with ‘.’

The second feature we will have is going to be settings menu. User will have a choice to setting dimensions of the grid and amount of tanks generated. Definitely there would be a sensible limitation. It also may decrease or increase the health of the fortress as difficulty changes. You always can quit the game and play with the default settings.

Also one of the possible features that we can implement is to save the game into the file. And any time you start the game you can load and continue playing from where you stopped. So there should be another option added to the battle menu. So to sum up, user will be able to make a move, quit or save the game. In such way we would need to add additional class which will serve the role of database and upload or download information from/to the game engine.