Backlog Game Project

The team decided to use MoSCoW method to separate and prioritize project parts in order to ensure the completion of basic deliverables.

Must have:

Description	How to demo	Notes
Game menu	The game starts from the game menu where the user presses “PLAY” button to start the game. In addition, “RULES” button would show the game rules for the user.	It would prevent the confusion for players on when the game actually starts
The game is turn based, giving the player the choice of either walking or shooting one soldier per round	Player 1 chooses to walk and where he wants to walk to, after which the game switches rounds, not allowing Player 1 to interact. Now Player 2 tries to shoot, and then the reverse happens… Player 1 can choose an action again. Etc.	We want to implement some basic coding together with learning how to properly integrate this with Swing to get a desired look and feel when doing an action within the game.
A player wins by eliminating all of the opponent’s army OR A player wins by capturing the opponents flag	One team shoots all the enemy soldiers, and a pop up appears announcing the winning player/team. One team walks one of their soldiers onto the enemy flag, which ends the game and the winning player/team gets announced.	This is simple programming with simple Swing additions for the pop ups.
The map the players play on has obstacles that block the path and line of sight of soldiers	After starting up the game we try shooting past an obstacle and we try walking straight through one. It should block the soldier’s action and not let anything through.	For this learning goal we want to try to utilize interfaces and inheritance to the fullest.

Should have:

Description	How to demo	Notes
Advanced UI and appearance	We want to achieve a game that is not comprised of just	Our learning goal is to get more acquainted with Swing

	one coloured squares but more ‘complex’ looking objects that are more distinct from one another.	and use it’s capabilities to the fullest so we can have as much freedom as we want when creating the graphics for our game.
Multiple kinds of soldiers to that can complement each other and give the players more possibilities for strategies	Each kind of soldier has different parameters when doing an action: e.g. the infantry can shoot up to 2 tiles, while a captain can shoot up to 3 tiles! We can test this by simply trying each action on each kind of soldier	Again interface and inheritance is a key part of this. The methods for each soldier are the same, they are just slightly different. So using these things are the easiest way to implement it.

Can have:

Description	How to demo	Notes
Starting positions of each team is up to the choice of the player	At the start of the game each player gets to choose the positions of all the soldiers in their designated team grounds (the outer most 2 tiles, depending on the side you play at)	

Won’t have:

Description	How to demo	Notes
Be too complex	The code must be doable in the time we are given to create it and it has to be able to be done with knowledge we have and a little extra knowledge we will gain in the next two weeks.	Learning goal: How to go for realistic goals that are doable in a time slot that is given to us. (Not be too passionate that we cannot finish it)