Sprint 1 Challenge

Your challenge is to create a games website, we will start by implementing a Mine Sweeper game, then we will implement a Sokoban game, then we will create a main page that show links to those 2 games. Wow! Can you do that? Let us see then.

Please note: some of you will only have time to complete part of the challenge, this is fine, make the best of your time and try to enjoy.

First thing first, build the mine sweeper:

Mine Sweeper

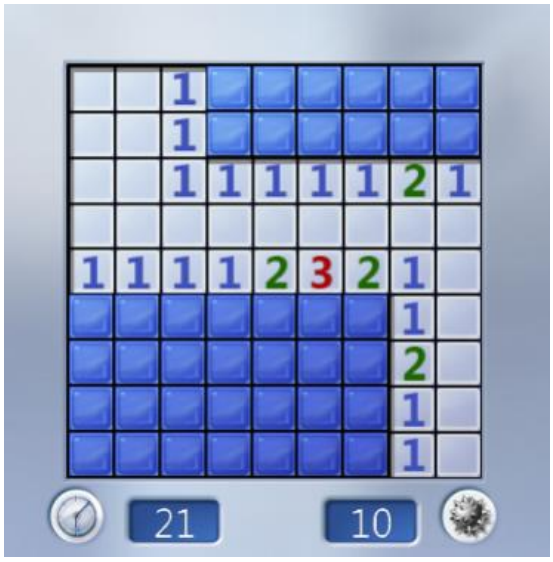
Blow your Mind

Preview

Your challenge is to create the Minesweeper game, and it's not an easy one. Let's practice some breaths.

Good.

Play the game a little bit and relax (you can play online here: http://minesweeperonline.com/).



It’s a good thing we have studied about Matrixes. Isn’t it?

Tips and Guidelines

As you know, there is usually more than one way to approach a challenge.

But as a guideline, we suggest having the following functions (it is ok to have more functions as needed).

`initGame()`	This is called when page loads
`buildBoard()`	Builds the board by setting mines at random locations, and then calling the setMinesNegsCount() Then return the created board
`setMinesNegsCount(board)`	Sets mines-count to neighbours
`renderBoard``(board)`	Print the board as a <table> to the page
`cellClicked(elCell, i, j)`	Called when a cell (td) is clicked
`cellMarked``(elCell)`	Called on right click to mark a cell as suspected to have a mine
`checkGameOver()`	Game ends when all mines are marked and all the other cells are shown
`expandShown``(board, elCell, i, j)`	Expand the shown class to neighbors At this point I needed to give each cell an ID (or a class) that looks like that: "*cell-3-2*" (3 and 2 are just examples)

Here are the **globals** I found myself using:

`gBoard`	The model
`gLevel = { SIZE: 4, MINES: 2 };`	This is an object by which the board size is set (in this case: 4*4), and how many mines to put
`gState = { isGameOn: false, shownCount: 0, markedCount: 0, secsPassed: 0 }`	This is an object in which I keep and update the current state: **isGameOn** – boolean, when true we let the user play **shownCount**: how many cells are shown **markedCount**: how many cells are marked (with a flag) **secsPassed**: how many seconds passed, you need to show it on the page

- Support 3 levels of the game
  - Beginner (4*4 with ~2 MINES)
  - Medium (6 * 6 with ~5 MINES)
  - Expert (8 * 8 with ~15 MINES)
- game ends when:
  - user clicked a bomb
  - all the mines are flagged and all the other cells are shown

That's it about the MineSweeper.