**Project Title: Minesweeper Cricket**

Your goal in this project is to create a game that is a simple combination of two classic games, Minesweeper and Cricket. . The project will involve designing and implementing the game logic, creating a responsive and interactive user interface using HTML and CSS, and incorporating dynamic functionality using JavaScript.

**Basic Requirement:** The below mentioned description is something a student has to build and on top of this, they can add extra customization as mentioned below.

The game will be played on a grid of either 6x6 or 7x7 blocks acting as cricket ground. Eleven fielders will be randomly distributed on the grid. The player (assume a single player for now) will click on a block at random. The block is revealed and if it does not contain a fielder, the player can score 1 run and the player can click on another random block. If the block contains a fielder the game ends and the final score is displayed. Note, please do not implement minesweeper style revealing of multiple blocks on one click, even as part of customization!

**Customization:** Here are some sample customization, but really you should come up with your own. Marks will be given per YOUR creativity.

1. The player can choose a grid size.
2. Score can be random, instead of a single run.
3. Convert from single player to two player or even multiplayer.
4. Power-ups can be randomly distributed on the grid, and when a player reveals a block containing a power-up, they can gain an advantage in the game such as score double the run or reveal the position of fielders for 1-2 sec.

**Process:**

You are welcome to browse for code on the web and use it. You will not however find something exactly the same as the description above. What we are looking for is your ability to go through existing code and modifying it as per need (this is what we do in real life as well, rarely do we write stuff from scratch :-)

Use HTML/CSS to design the layout and visual elements of the game like the game board, scorecard etc. Use JavaScript to implement the game rules, including random player distribution, block generation, and scoring.

**Marks Distribution (20 marks):** These marks will be awarded via a viva, where you demonstrate the project and TAs will ask questions.

* The layout of the user interface, overall visual appeal: 4 marks
* Functionality and correctness of the code logic : 6 marks
* Customization: 4 marks
* Viva: 4 marks
* Code quality, organization, and comments, latex based report: 2 marks

Note: All submitted code in a given project will be checked for plagiarism and if caught, your case will be forwarded to DDAC.

**Upload:** Please upload all relevant files, including report as a zip file on BodhiTree before the deadline. NO CHANGES TO THE FILES WILL BE PERMITTED POST THIS, INCLUDING DURING EVALUATION. During evaluation, TAs will download the zip from Bodhitree, set it up on their machine and then interact with you for evaluation. They will use their own input/testcases during this time.