Design Document for Undo Functionality

Rules:
The undo functionality allows a player to undo the previous move done if both players have done one or more moves and have not undone them.
If there are no more moves to be undone, the undo functionality should be disallowed.

The design of the undo functionality can follow the Model-View-Controller (MVC) architecture pattern in the following manner.

Model

The Model component of the application is responsible for managing the game state, which includes tracking the current layout of the game board, the players' moves, and the number of moves played so far. It will also incorporate an undo method to eliminate the most recent player move. Along with the board and player moves, it will also store the undo history. The undo history will be stored as a stack of tuples, with each tuple containing the row and column indices of the cell that was modified during the move that was undone.

View:

The view component displays the game board with other visual elements and receives user input from the controller component. The view component will display the undo button below the game board. At first, the undo button is enabled, however, the controller component may disable it in case there are no moves in the undo history.

Controller

The Controller component acts as a bridge between the View and the Model, updating the game state as per the user input. On clicking the undo button, the Controller calls the Model's undo method to remove the last move made. The View reflects the updated state. When there are moves in the undo history, the Controller pops the most recent move from the stack and updates the game board. If there are no moves, the undo button is disabled.