**AP's XO Game User Story**

**As a player, I want to:**

1. **Start a new game**:
   * When I open the game, I should see an empty 3x3 grid and a message indicating that it is player X's turn.
2. **Take turns with another player to place Xs and Os on a 3x3 grid**:
   * I should be able to click on any empty cell in the grid to place my symbol (X or O).
   * After I place my symbol, it should immediately become the other player's turn.
3. **See whose turn it is to play**:
   * There should be a clear message or indicator showing whose turn it is (either player X or player O).
4. **Check for a winner after each move**:
   * After each move, the game should automatically check if there is a winner.
   * A player wins if they have three of their symbols aligned horizontally, vertically, or diagonally.
5. **See the game end with a win or a draw**:
   * If a player wins, a message should announce the winner.
   * If all cells are filled and there is no winner, the game should announce a draw.
6. **Reset the game to start over**:
   * There should be a reset button that clears the board and starts a new game with player X's turn.

**Acceptance Criteria:**

1. **Game Start**:
   * The game begins with an empty 3x3 grid.
   * The initial message indicates that it is player X's turn.
2. **Taking Turns**:
   * Players alternate turns, starting with player X.
   * Only empty cells can be clicked to place a symbol.
   * After placing a symbol, the turn indicator updates to show the next player's turn.
3. **Checking for a Winner**:
   * The game checks for a winner after each move.
   * A winning message is displayed when three of the same symbols align horizontally, vertically, or diagonally.
   * The game stops accepting moves once a winner is declared.
4. **Draw Condition**:
   * The game checks for a draw when all cells are filled.
   * A draw message is displayed if there is no winner and the board is full.
5. **Game Reset**:
   * A reset button is visible at all times.
   * Clicking the reset button clears the board and starts a new game with player X's turn.
   * The turn indicator resets to show that it is player X's turn.

**Dependencies:**

* Node.js
* React

**Technical Details:**

1. **Frontend**:
   * Use React for the user interface.
2. **Backend**:
   * Use Node.js to manage the game state and logic.
   * Implement logic to track the state of the game, determine the winner, and handle game resets.
   * Ensure the game state is persistent during a session.
3. **Persistent State**:
   * Use local storage or session storage to persist the game state during a session.
   * Load the game state from storage when the component mounts to ensure continuity.
4. **Game Logic**:
   * Implement the logic to check for a winner after each move.
   * Implement the logic to detect a draw condition.
   * Implement the logic to reset the game state when the reset button is clicked.