

# From Marcus

Marcus Galdino

## Summary of the Program

Josh, this project is awesome! You've built a multiplayer chess game in Blazor, complete with a lobby system, player authentication, and even a way for spectators to watch games. The amount of thought and effort you put into this is clear—it goes way beyond what's expected for a class project. Honestly, this is the kind of project that people could actually use and enjoy playing. Great work!

## Feedback on the Running Program

### 1. How Easy Was It to Use?

- The lobby setup is really smooth. I like how you can see who's playing or waiting for a match. Creating and joining games felt straightforward, which is great!

### 2. How to Exit:

- I didn't notice a way to leave a game and go back to the lobby. Adding a "Return to Lobby" button would make it easier to navigate, especially if someone wants to start over or watch another game.

### 3. Feeling Lost or Out of Control:

- The game itself feels smooth, but sometimes when selecting a piece or trying a move, it wasn't clear why something didn't work. Maybe showing a quick message like "Invalid move" or keeping the invalid action grayed out would help.

### 4. Visual Appeal:

- The chessboard looks good! The alternating black-and-white tiles are classic, and I like how the pieces are displayed. Adding highlights for selected pieces and valid moves would make it pop even more.

### 5. Did It Crash?

- Nope! Everything worked perfectly during testing. That's a big win.

### 6. Flaws:

- Honestly, the only small thing I noticed was the lack of feedback for invalid moves or actions, which made me second-guess what I was doing at times.

## Feedback on the Code

### 1. Variable Names:

- Your variable names are clear and easy to understand. Things like `AllowedMovement`, `CurrentPlayer`, and `PlayerID` make the code very readable.
2. Class and Method Names:
    - Your method names, like `HandleCellSelection` and `CalculateValidMoves`, are spot-on. They're easy to follow and describe exactly what they're doing.
  3. Indentation & Whitespace:
    - The code is super clean and well-organized. I didn't see any major issues here—nice work!
  4. Method Length:
    - Some methods, like `LayoutGamePieces` and `CalculateValidMoves`, are pretty long. Breaking them into smaller helper methods could make them easier to read and maintain, but they're still solid as-is.
  5. Was It Easy to Read?
    - Absolutely! The logic is clear, and it's easy to see how everything connects. You've done a great job of keeping things organized.
  6. Suggestions:
    - The CSS strings in Blazor components work, but moving them to a separate CSS file would make them easier to tweak and reuse later.

## Suggestions for Improvement

- **User Feedback:** Adding clear visual feedback for actions like invalid moves or showing valid tiles when selecting a piece would make the game feel even smoother.
- **Code Refactoring:** Breaking up longer methods into smaller ones would help keep things modular and easier to tweak in the future.

## Final Thoughts

Josh, this project is incredible! The multiplayer aspect, the integration with Blazor, and the thought you put into the backend logic are all super impressive. Keep crushing it, man.