# **C24A Clue Game - Let's Play (Clue Pair)**

**Originally written by Mark Baldwin but revised by Michael Crews and Jhon Malagon.**

### Purpose:

* Finish game functionality

### Assignment:

This assignment will add the final functionality. The specific items you need to add are:

* Finish the computer player
* Finish the human player
* Respond to game control make accusation button

### Player Turn - Computer

In the previous assignment, we implemented code so that the computer player selects a target and moves to that location. Now we'll add functionality so that:

* if the player is ready to make an accusation, that is done at the beginning of the turn
* otherwise the player picks a target and moves to it (this was done last time)
* if the target is a room, the player needs to:
  + update the ControlPanel with the guess and the result
  + move the suggested person to the room
  + let all players know which card was "shown"
  + set a flag for last room visited (to keep players from continually going into the same room)
  + create a suggestion (this was tested in Clue Players)
  + call the Board method to handle disproving suggestions. This was tested in Clue Players, but must be updated with GUI functionality:
  + set a flag if no one can disprove the suggestion (see Hint)

#### Hint:

* How do we know whether the player should make an accusation? This will happen if on the previous turn the player made a suggestion that was not disproved (unless the player has the room card). In that case, I set a flag that the player should make an accusation, and the "suggestion" then becomes the "accusation." For example, if the suggestion is Professor Plum, Library and Knife, and no one can disprove (and I don't have the Library card), then on the next turn use that suggestion as an accusation.

### Player Turn - Human

In the previous lab, we implemented functionality so that when it's the human's turn, the board cells are highlighted and the human clicks and moves to a location. In this lab we will update the process to allow for suggestions (accusations for the human are handled by pressing the Make Accusation button, so are not part of this discussion).

When the human player enters a room, the program should automatically:

* Display a modal dialog, similar to the one shown below (modal means the user must finish the dialog before the program can continue).
* When the dialog is submitted, get the Solution from the dialog
* Call the Board method to handle disproving suggestions (same as for computer players)

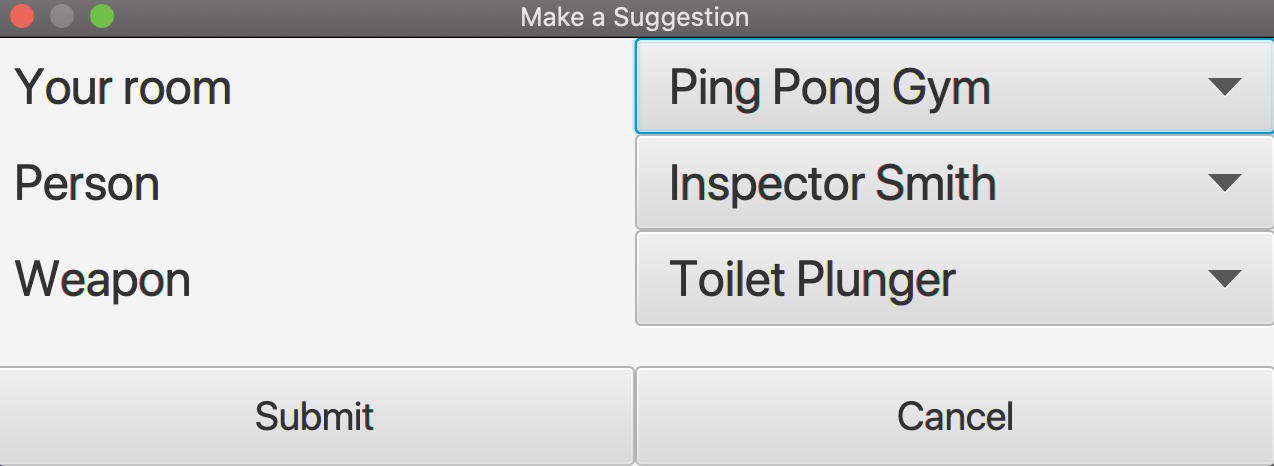


Figure 1. Guess Dialog

### Making an Accusation - Human

The human is allowed to make an accusation at the beginning of their turn. When the player presses the Make Accusation button, a dialog is displayed that is similar to the one shown above, except that the player should also select the room. When the dialog is submitted, the accusation is checked (this functionality was tested in a previous lab). An appropriate dialog displays to let the user know whether the accusation was correct. Note that, unlike the board game, play does not end when the human makes an incorrect accusation (because, unlike the board game, the player doesn't see the answer - just knows that the answer is not correct).

Hint: you may want to include both a submit and a cancel button, and only process the accusation if the user presses submit.

### Awesome Points

My jar file contains the basic requirements for your Clue game (although a slighly older set of requirements). There are clearly many ways this could be extended. You can earn 5 "awesome points" (i.e., extra credit) for implementing one of those extensions. Some ideas that have been mentioned:

* Dynamically recalculating the grid size when the user resizes the window
* Updating the display so that it's apparent to the user when two players are in the same room (e.g., when a suggestion is made and the player's token gets moved to the room).
* Dynamically determining where to display the room name.
* Setting up automated testing so the game will automatically play against itself for 100 games.
* Any other equivalent awesome extension.

To get credit, you must describe your extension in a text file named "ExtraCredit.txt" when you submit.

### Specific Requirements

This assignment is worth 100 points. Note that one requirement is that all tests must still pass.

#### Computer Player Turn

* (6) If the computer player is in a room, it makes a suggestion

#### Human Player Turn

* (6) Ensure that human is prompted for a suggestion when enter a room.

#### Making a Suggestion - Both

* (6) When a player enters a suggestion, it is shown on the control panel.
* (6) Other players try to disprove.
* (6) Response displays the card that disproved the suggestion, or "no new clue" if none of the players were able to disprove
* Accused player should move to the room. (Optional, no points)

#### Making a Suggestion - Human

* (10) Room should match the current room location. Combo boxes should be used for Person and Weapon. You could also use a combo box for Room that only has one option.

#### Making a Suggestion - Computer

* (6) When a computer player enters a room, it makes a suggestion. Room should match the current location.

#### Making an Accusation - Human

* (10) Human is allowed to make accusation at beginning of their turn. An error message displays if the Make Accusation button is pressed and it is not the human's turn.
* (6) Accusation includes person, room and weapon.
* (10) When the human player makes an accusation, a message is displayed to inform whether it is correct or not. The game does not end if the accusation is incorrect, but that completes the player's turn.

#### Making an Accusation - Computer

* (10) When the computer player makes an accusation, a message is displayed that includes the person, room and weapon and whether the accusation is correct or incorrect.

#### Other

* (8) Functions that were tested previously (e.g., dealing, computer selecting a target, disproving a suggestion, and computer making a suggestion) all work as intended.
* (10) Coding standards, cleanliness of code, documentation.

### Submit

Zip your project and submit.