

Use Case Description – Move a Frog

Primary Actor: The player(s)

Stakeholders and Interests:

- Player(s): Want to move one of their frogs strategically to reach an opponent's home leaves, using bridges or jumps according to the game rules. They want the system to clearly show legal moves and to respond accurately to their actions.
- Other Players: Want the system to correctly enforce movement rules to ensure fairness and avoid illegal moves that could impact gameplay.

Preconditions:

- The game has been successfully set up.
- It is the player's turn.
- The player has at least one frog he can legally move or jump to another lily pad.

Postconditions:

- The frog has successfully moved to another lily pad.
- The board and game state updates to reflect the new positions.
- If applicable, the turn has passed to the next player.

Main Success Scenario:

1. The system displays the current game board and indicates that it is the player's turn
2. The system highlights all frogs belonging to the current player
3. The player selects one of their frogs to move
4. The system highlights all legal destinations for the selected frog based on current bridges and game rules.
5. The player selects a valid destination lily pad connected by a bridge
6. The system moves the selected frog to the chosen lily pad
7. The system removes the bridge that was used to perform the move
8. The system updates the game board to reflect the new frog positions and bridges that have been removed
9. The system verifies that all movements were valid and that no frog remains on an invalid or disconnected lily pad
10. The system confirms the move's completion and ends the player's turn
11. The system updates the display, showing the new frog and bridge layout

Alternative Flows:

2a. Player Selects an Invalid Frog

1. The player selects a frog that does not belong to them or cannot move
2. The system displays an error message indicating the frog is invalid
3. The system returns to Step 2 of the main success scenario

4a. The player selects an invalid Destination

1. The player selects a lilypad not connected by a bridge or occupied by another frog without a valid jump path
2. The system signals an invalid move and re-highlights valid destinations
3. The system returns to Step 4 of the main success scenario

6a. Player selects a lilypad occupied by another frog (of their own)

1. The system detects that the destination lilypad is already occupied by one of the player's own frogs
2. The system initiates a chain jump sequence
3. The system prompts the player to choose an adjacent lilypad connected by a bridge for the displaced frog to move
4. The system moves the displaced frog to the chosen adjacent lilypad and removes the bridge used in that move
5. If the new lilypad is also occupied by another frog, the system repeats this process for that frog
6. The chain jump continues until all displaced frogs have valid landing positions and no further jumps are required
7. The system returns to Step 6 of the main success scenario

6b. Player selects a lilypad occupied by another frog (of an opponent)

1. The system detects that the destination lilypad is occupied by a frog belonging to another player.
2. The system initiates a chain jump sequence.
3. The system prompts the player to choose an adjacent lilypad connected by a bridge for the displaced frog to move to.
4. The system moves the displaced frog to the chosen adjacent lilypad and removes the bridge used in that move.
5. If the new lilypad is also occupied by another frog, the system repeats this process for that frog.
6. The chain jump continues until all displaced frogs have valid landing positions and no further jumps are required.
7. The system returns to Step 6 of the main success scenario.

6c. Frog lands on a lilypad without a valid relocation option

1. The player's frog lands on a lilypad that is already occupied, but the displaced frog has no adjacent lily pads connected by bridges to move to.
2. The system declares the move illegal.

3. The system reverts all frogs and bridges to their original positions before the attempted move.
4. The system prompts the player to select another valid move.
5. The system returns to Step 4 of the main success scenario.

6d. Players select an invalid relocation during a chain jump

1. The player attempts to move a displaced frog to a lilypad that is occupied or not connected by a bridge.
2. The system signals that the relocation is invalid.
3. The system prompts the player to select a different adjacent lilypad.
4. The system returns to Step 3 of flow 6a or 6b.

7a. Destination is the opponent's home base

1. The player moves a frog to a lilypad that is part of the opposing player's home base.
2. The system recognizes the destination as a home base and does not remove the bridge used for that move.
3. The system continues to Step 8 of the main success scenario.

8a. Frog reaches opponent's home base

1. After completing a move or chain jump, the system detects that the frog has landed on the opponent's home base.
2. The system locks that frog in place for the remainder of the game.
3. The system checks whether all frogs of that player's colour have reached their opponent's home base.
4. If all frogs have reached their destination, the system triggers the end of the game
5. If not, the system proceeds to the next player's turn.
6. The system continues to Step 8 of the main success scenario

Exceptions

- The player attempts to move when it is not their turn.
- The system detects inconsistent bridge states after a move
- The player is mid-turn, the system pauses and waits for confirmation before reverting or saving progress

Special Requirements

- The system must visually display all valid moves and possible chain jump paths before confirmation.
- Bridge removals must be clearly animated to reinforce understanding of the rules.
- The system must support colour-accessible design and clear frog and bridge indicators for all players.
- The chain jump sequence should be intuitive, allowing the player to make each relocation choice easily.

Open Issues

- Should the system display a preview of all chain jumps before final confirmation?
- Should bridges connected to home bases remain permanently protected in the user interface?
- How should the system handle simultaneous jumps in a 4-player game when multiple frogs could be displaced?