

Use Case 2: Take a Turn

Primary Actor: Player

Stakeholders and Interests:

- **Player:** Wants to roll dice, make the best possible move, and place a stone strategically.
 - **Other Players (including AI):** Expect fair gameplay, adherence to rules, and smooth turn transitions.
 - **System:** Ensures all moves are valid, updates the board and scores correctly, and tracks game progress.
-

Preconditions:

- It is the Player's turn.
 - The game board is set up, and all required components (dice, stones) are in place.
 - The System is actively tracking scores and game status.
-

Success Guarantee (Postconditions):

- The Player has completed their turn (rolled dice, selected a combination, placed a stone if possible).
 - The board and scores are updated correctly.
 - The turn passes to the next player.
 - If a **win condition is met**, the System declares the winner.
-

Main Success Scenario:

1. **The System notifies the Player that it is their turn.**
2. **The Player rolls six dice.**

3. **The System displays possible valid combinations** based on the roll.
 4. **The Player decides to:**
 - Proceed with the initial roll **(Alt 1)**, or
 - Re-roll **all** dice (up to two times) **(Alt 2)**, or
 - Re-roll **selected** dice (up to two times) **(Alt 3)**.
 5. **The Player selects a valid combination and attempts to place a stone.**
 - **(Alt 4: No valid space available)** – The Player cannot place a stone, and their turn ends.
 6. **The System updates the board and scores.**
 7. **The turn passes to the next player.**
-

Alternative Flows:

- **Alt 1: Player proceeds with first roll**
 - The Player skips re-rolls and immediately selects a combination.
 - Flow resumes at Step 5.
 - **Alt 2: Player re-rolls all dice (up to 2 times)**
 - The System updates dice values and displays new valid combinations.
 - Flow resumes at Step 5.
 - **Alt 3: Player re-rolls selected dice (up to 2 times)**
 - The System updates only the chosen dice and recalculates valid combinations.
 - Flow resumes at Step 5.
 - **Alt 4: No valid space available for the chosen combination**
 - The System notifies the Player.
 - The Player's turn ends, and the next player proceeds.
-

Exceptions:

- **Game Already Has a Winner**
 - If a Player reaches the winning condition, the System **declares the winner immediately** and ends the game.
 - **Player Exceeds Turn Time Limit**
 - A countdown timer (e.g., **30 seconds**) is applied.
 - If time runs out, the Player **automatically loses** and is removed from the game.
 - If the last remaining opponent wins by default, the game ends.
 - **Player Leaves Mid-Turn**
 - A countdown timer begins.
 - If the Player does not return before time expires, they **automatically lose** and are removed.
 - The next player takes their turn.
 - **System Error During Dice Roll or Re-roll**
 - If a technical issue occurs (e.g., dice not displaying correctly), the System attempts **automatic recovery** or prompts a game restart.
-

Special Requirements:

- The interface should clearly indicate **whose turn it is** and highlight valid moves.
 - **Colorblind-friendly design** should be applied to dice values and board indicators.
 - AI players should make decisions quickly to avoid delays.
 - Turn processing (rolling, scoring, turn transitions) should be **smooth and fast** (ideally under 2 seconds).
-

Open Issues:

1. **Should the Player be allowed to undo a move before finalizing?**

2. **Should AI players always re-roll if given the option, or should they sometimes accept the first roll?**
3. **How should ties be handled if multiple players reach the same score at the end?**