|  |  |
| --- | --- |
| Use Case: Play (Single Player)  Description: A one-person local game of Sudoku  Actors: Player  Pre-Conditions: Player has knowledge of Sudoku gameplay  Post-Conditions: Player is rewarded for completing the game | |
| Main Success Scenario: Player is rewarded | |
| Player | System |
| 3) Enters board size (A1)  8) Enters board coordinates (A1)  12) Enters board coordinate value (A1)  17) Repeats 8 and 12 until game ends | 1) Welcomes user  2) Prompts user for board size  4) Displays user input (E0)  5) Creates empty board  6) Displays board  7) Prompts user for board coordinate to modify  9) Displays user input  10) Checks for valid coordinates (E1)  11) Prompts user for coordinate value (E2)  13) Displays user input  14) Checks for valid coordinate value (E1)  15) Edits board coordinate with user input  16) Displays board with Player’s modifications (E2)  18) Repeats 6, 7, 9, 10, 11, 13, 14, 15, 16 until game ends  19) Congratulates player  20) Ends Game |
| Alternatives  A1: User may quit application, if so 20) | |
| Exceptions  E0: Board size may not be valid, set the board to a default size  E1: Move might not be valid; the system repeats 7) for 10) or 11) for 14)  E2: System checks if board has been solved, if so 19) | |

pg. 41 – 46 USE CASE DIAGRAMS (PICTURES)

|  |  |  |
| --- | --- | --- |
| Use Case: Play (Two Player)  Description: A multiplayer network game of Sudoku  Actors: Player1, Player2  Pre-Conditions: Players have knowledge of Sudoku gameplay and active P2P network connection between their machines  Post-Conditions: One player wins the game and is rewarded for their achievement | | |
| Main Success Scenario: One player is rewarded | | |
| Player1 | Player2 | System |
| 3.a) Chooses multiplayer option (A1)  5.a) Enters Player2’s IP address  8) Enters board size (A2)  14) Enters board coordinates (A3)  17) Enters coordinate value (A3)  23b) Repeats 14 and 17 until game ends | 3.b) Chooses multiplayer option (A1)  5.b) Enters Player1’s IP address  23a) Repeats 14 and 17 until game ends | 1) Welcomes Player1, Player2  2) Prompts Player1 for single player or multiplayer gameplay (E0)  4) Prompts Player1, Player2 of IP address of peer (E1)  6) Creates connection between Player1 and Player2 (E2)  7) Alerts user of successful connection  9) Prompts Player1 for board size  10) Informs Player2 of board size  11) Creates partially filled board (E2)  12) Displays board for Player1 and Player2  13) Prompts Player1 for puzzle coordinate to modify  14) Displays Player1’s coordinates  15) Checks for valid coordinates (E4)  16) Prompts Player1 for coordinate value  18) Displays Player1’s coordinate value  19) Checks for valid coordinate value (E4)  20) Edits board coordinate with user input  21) Displays board for Player1 and Player2 (E5)  22) Prompts the other player for move  24) Repeats 12, 13, 15, 16, 18, and 19 for Player2, Player1 respectively until game ends  25) Ends game  26) Displays congratulates message for winning player  27) Terminates network connection |
| Alternatives  A1: Player may choose single player option, then the *Single Player Use Case* is followed  A2: Board size may not be valid, then the system set board to default value  A3: Player may quit the game; the system goes straight to 22) and 23) | | |
| Exceptions  E0: Gameplay type may be invalid; repeats question to player.  E1: IP address may not be found; in this case display error message  E2: Connection many not be possible to make  E3: Board size may not be valid, set the board to a default size  E4: Move might not be valid; the system repeats 13) for 15) or 16) for 19)  E5: System checks if board has been solved, if so 24) | | |

pg. 41 – 46 USE CASE DIAGRAMS (PICTURES)