# Assignment 2 Test Plans

**Common file cases (Common to both SudokuInterface and SudokuFrame)**

| **Scenario** | **Expected Results** | | |
| --- | --- | --- | --- |
| **Row** | **Column** | **Value** | **filename** | **Result** |
| 1. File is missing |  |  |  | notSudoku.txt | Error: notSudoku.txt does not exist |
| 1. File has the wrong format |  |  |  | easySudoku.txt | Error: sudoku.txt does not have the correct file format |
| 1. No file input |  |  |  |  | Message: sudoku.txt will be opened |

**Error cases (CLI)**

| **Scenario** | **Expected Results** | | |
| --- | --- | --- | --- |
| **Row** | **Column** | **Value** | **filename** | **Result** |
| 1. Invalid column number | 8 | 10 |  | sudoku.txt | Error: Invalid column number. Needs to be between 1 and 9. Please try again. |
| 2. Invalid row number | 0 | 5 |  | sudoku.txt | Error: Invalid row number. Needs to be between 1 and 9. Please try again. |
| 3. Invalid value | 1 | 6 | 11 | sudoku.txt | Error: Invalid value. Needs to be between 1 and 9. Please try again. |
| 4. Invalid move (location already used) | 8 | 9 |  | sudoku.txt | Error: Invalid move. That location already has a value. Please try again. |
| 5. Invalid move (the value already exists in that row) | 1 | 4 | 7 | sudoku.txt | Error: Invalid move. There is already a 7 in that row. Please try again. |
| 6. Invalid move (the value already exists in that column) | 2 | 5 | 4 | sudoku.txt | Error: Invalid move. There is already a 4 in that column. Please try again. |
| 7. Invalid move (the value already exists in that square) | 8 | 4 | 3 | sudoku.txt | Error: Invalid move. There is already a 3 in that square. Please try again. |
| 8. Last square - lose |  |  |  | sudoku.txt | The last square is invalid, and the player loses |

**Error cases (Frame)**

| **Scenario** | **Expected Results** | | |
| --- | --- | --- | --- |
| **Row** | **Column** | **Value** | **filename** | **Result** |
| 1. Invalid move (the value already exists in that row) | 4 | 8 | 1 | sudoku.txt | Pops up an error message: There is already a 1 in that row. |
| 2. Invalid move (the value already exists in that column) | 5 | 6 | 4 | sudoku.txt | Pops up an error message: There is already a 4 in that column. |
| 3. Invalid move (the value already exists in that square) | 9 | 7 | 5 | sudoku.txt | Pops up an error message: There is already a 5 in that square. |
| 4. Last square - lose |  |  |  | sudoku.txt | The last square is invalid, and the player loses |

**Success cases (both CLI and Frame)**

| **Scenario** | **Expected Results** | | |
| --- | --- | --- | --- |
| **Row** | **Column** | **Value** | **filename** | **Result** |
| 1. Correct move (row) | 4 | 8 | 2 | sudoku.txt | No error is shown, and the entered value is displayed on its place. |
| 2. Correct move (column) | 2 | 1 | 6 | sudoku.txt | No error is shown, and the entered value is displayed on its place. |
| 3. Correct move (value) | 3 | 5 | 9 | sudoku.txt | No error is shown, and the entered value is displayed on its place. |
| 4. The user quits |  |  |  | sudoku.txt | If the game is not saved yet, the program will ask the user, either by a pop-up message or in the console, whether he or she wants to save the game. |
| 5. The user quits. Chooses to save the game first. |  |  |  | sudoku.txt | A success message is shown, when the file is saved, and the programs exits. |
| 6.The user quits.  Chooses NOT to save the game. |  |  |  | sudoku.txt | The program quits. |
| 7. The user saves the game manually |  |  |  | sudoku.txt | The program saves the current state of the game. A success message is shown and the “Save game” button is disabled. The player can continue to play. |
| 8. Last Square - win |  |  |  | sudoku.txt | The last square is valid, and the player wins. |