

Documento di System Test

Test-Case:

- T_01. Test starting a new game
- T_02. Test choosing the start board configuration between many alternatives.
- T_03. Test loading a saved game.
- T_04. Test moving blocks by GUI interactions and moves counter.
- T_05. Test undo previous moves up to the game's start.
- T_06. Test help to solve the puzzle.
- T_07. Test saving the actual game's state.
- T_08. Test game reset.

ID	T_01
Name	Test starting a new game.
Summary	Test if program correctly starts a new game
Test Case Design	The player must start a new game. From the main window, the player presses the "Start Game" button. The default game window (Level1) is displayed. Now the player can play the game.
Pre-Conditions	Program is in main window
Post-Conditions	Game window is displayed and the player can play the game.
Test Script	<ul style="list-style-type: none">- Board is displayed in the initial state- Moves Counter is displayed to 0- Help, Save, Undo, Reset, Back options are displayed and active
Test case Execution Record	System correctly starts a new game PASS
Execution Variables	No execution variables needed for this test.

ID	T_02
Name	Test choosing the start board configuration between many alternatives.
Summary	Test if program correctly starts a new game choosing from different alternatives
Test Case Design	The player has to start a new game choosing from many alternatives. From the main window, the player presses the "Choose configuration" button. A list with the available configurations is displayed. The user chooses what he wants to be started. The chosen game is displayed. Now the player can play.
Pre-Conditions	Some initial configurations of games must be saved.
Post-Conditions	Chosen game window is displayed and the player can play the game.
Test Script	<ul style="list-style-type: none"> - Chosen board is displayed in the initial state - Moves Counter is displayed to 0 - Help, Save, Undo, Reset, Back options are displayed and active
Test case Execution Record	System correctly starts the chosen game PASS
Execution Variables	Test must be done for all available configurations (level1, level2, level3, level4).

ID	T_03
Name	Test loading a saved game.
Summary	Test if program correctly loads a saved game.
Test Case Design	Player must press "Load Game" button from the main window. A list with the available saved game is displayed. Player chooses what game he wants to continue the chosen game from the last save point. The chosen saved game is displayed. Now the player can play the game.
Pre-Conditions	A previous partial game must be saved and program is in main window.
Post-Conditions	Chosen game window is displayed and the player can play the game.
Test Script	<ul style="list-style-type: none"> - Board is displayed in the last state it was in before saving. - Moves Counter is displayed to the number of moves made before saving. - Help, Save, Undo, Reset, Back options are displayed and active.
Test case Execution Record	System correctly loads the chosen saved game. PASS
Execution Variables	No execution variables needed for this test. Any of available saved game can be used for this test.

ID	T_04
Name	Test moving blocks by GUI interactions and moves counter.
Summary	Test if program correctly moves blocks and if correctly increment moves counter.
Test Case Design	Player must start a new game by selecting one of the available configurations or load a saved game and make some moves by tapping and dragging a block in the direction he wants to move it.
Pre-Conditions	New game has to be started or a saved game has to be loaded. The game is displayed.
Post-Conditions	If there is enough free space, the move is made, board is updated with the move made and the moves counter is incremented, otherwise the block is not moved and board and moves counter stay in the current state.
Test Script	<ul style="list-style-type: none"> - Moves counter must be incremented if the moves was made - Board must be update with the move made if there was enough free space
Test case Execution Record	System correctly moves blocks and updates moves counter. PASS
Execution Variables	No execution variables needed for this test. Any moves can be made for this test.

ID	T_05
Name	Test undo previous moves up to the game's start.
Summary	Test if program correctly undo moves up to game start condition
Test Case Design	Player must start a new game selecting one of available configuration and make some moves to change the game state. Then player have to press the undo button until the moves counter is greater than zero.
Pre-Conditions	New game has to be started and at least a move has to be made
Post-Conditions	Moves counter has to be decreased by number of clicks of undo button.
Test Script	<ul style="list-style-type: none"> - Moves counter decreased by number of clicks of undo button
Test case Execution Record	System correctly undo moves up to game start conditions PASS
Execution Variables	No execution variables needed for this test. Any of available configuration can be used for this test.

ID	T_06
Name	Test help to solve the puzzle.
Summary	Test if program correctly suggest move to solve the level
Test Case Design	Player must start a new game selecting one of available configuration and press help button.
Pre-Conditions	New game has to be started
Post-Conditions	Moves counter has to be increased and a move has to be automatically done or last move has to be canceled if no best move is available
Test Script	<ul style="list-style-type: none"> - Moves counter increased - Move automatically done in the board
Test case Execution Record	System correctly suggests a move or undo moves if it cannot suggest best moves. PASS
Execution Variables	No execution variables needed for this test. Any of available configuration can be used for this test.

ID	T_07
Name	Test saving the actual game's state.
Summary	Test if program correctly save actual game state
Test Case Design	Player must start a new game selecting one of available configuration and make some moves to change the game state. Then player have to press the save button and wait message box with saving to save game state into a file located at user home directory
Pre-Conditions	New game has to be started and at least a move has to be made
Post-Conditions	New saving file has to be created into user home directory and a message box has to appear to communicate saving results
Test Script	<ul style="list-style-type: none"> - MessageBox communicating save succes - New saving file has to appear at User Home/KlotskiGame/Saving/
Test case Execution Record	System correctly saves the game after creating the default directory User Home/KlotskiGame/Saving PASS
Execution Variables	No execution variables needed for this test. Any of available configuration can be used for this test.

ID	T_08
Name	Test game reset.
Summary	Test if game can be correctly reset to initial status of chosen configuration
Test Case Design	Player must start a new game by selecting one of available configuration and make some moves to change the game state. Then player have to press the reset button to go back to initial status with no moves done.
Pre-Conditions	New game has to be started and at least a move has to be made or a saved game has to be loaded correctly.
Post-Conditions	Moves counter must be reset and board must return to initial state of the select configuration
Test Script	<ul style="list-style-type: none"> - Moves counter must be zero - Board must be the initial board of selected configuration
Test case Execution Record	System reset moves counter and board to initial status PASS
Execution Variables	No execution variables needed for this test. Any of available configuration or any saved game can be used for this test.