Team X: Margherita Cattapan, Mattia Sartori, Francesca Damian
Via Gradenigo 6B, Padua (PD)

Our Klotski manual

4th September 2023

PANORAMICS This is the manual for the presentation of the team and for the program of our Klotski.

Our goals

1. An intuitive and basic application interface.
2. Multiple game modes.
3. Some additional actions to increase the freedom and the fun for the user.
4. The possibility to play and to stop the game even at the start in order to never lose the hard progress obtained.

LIBRARIES AND SOLVER The libraries used in this project are java, JavaFX, JUnit, JSON, JavaIO and JavaUTIL. The algorithm used for the solver has been share on the world wide web by Andrew White on his blog.
andrewewhite.net/wordpress/2012/03/02/slider-solver-klotski-puzzles/

US AND THE GAME INTRODUCTION

How the project started

After long and sunny months, we were capable somehow to form a group of 4 students. Francesco arrived later full of good intentions. All smoke no roast in the end. We were forced to remove him from the project do to the fact that he didn't actively do anything. The first phases of the project were focused to develop ours individual ideas in order to find the best fit for our group project. We also together put down a program to split out the timeline and assignments to each of the group in order to finish the app.

Graphical User Interface

For the GUI's realization e for the buttons needed to finish the assignment has been used: SceneBuilder. The blocks have been implemented thanks to the class Rectangle.

THE REAL MANUAL OF THE GAME

At the opening of the application the game gives to the user the choice between many configurations. When the choice has been made the board shows it, it's always formed by the blocks of the game: the special one (4x4), 4 blocks 1x1, 4 blocks (2x1) and one block (1x2). The blocks cam be selected by the cursor of the mouse and can be moved by WASD, (up, left, down, right). Near by the board the user can find many buttons and the counter of the moves. Here following their description:

- **UNDO;**

This button if clicked unmoved the last move made. It can be clicked n times, with n the number of moves made until that moment. If it's clicked more times the user is alerted by a message.

- **BEST NEXT MOVE**;

This key option makes easier to complete the configuration that the user is playing now. The solver used has been already quoted many times. (Andrew White's blog) Thanks to this solver, every configuration is solvable, and the move can be executed and memorized in the History of the moves made.

- **RESET**

This push-button lets the board to be settled on the original configuration or if a saved has already been made to this last one.

- **CONFIGURATION**

This button can be clicked in any moment (even during in game) and let the used choose a new configuration, all memorized in the code. But, the user should know that the change of the configuration during a game lets to the erasing of the move's history of the configuration in game. For this case the user will be alert with a message.

- **MOVE'S COUNTER**

As the assignment asked, a move's counter has been implemented in the project.

- **DEFAULT**

There is the possibility with the use of this button to return at the basic configuration of the game.

- **SAVE**

This function is extremely connected with all the others. With this button the user can save the progress of the game till that moment. The memorization of the configuration saved is in a JSON file.