Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Game Interface

Team:

[Thomas Jones] - Id: [040953052]

Game Proposal - NumPuz

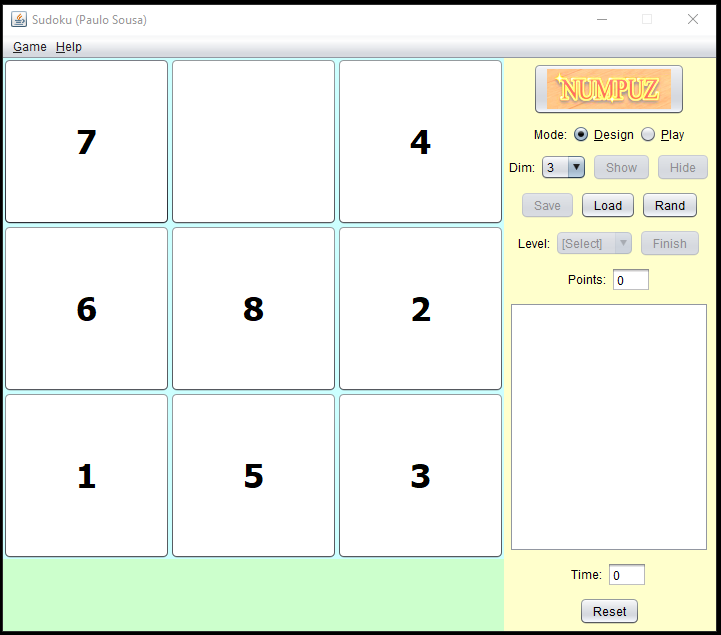
***This template is suggested (not mandatory) to answer A11 Specification.***

|  |  |
| --- | --- |
| **Part**  **1** | **GUI Definition** |

**EXPLANATION**

*The purpose of this assignment is to define the elements of the GUI application to be used in your game implementation.*

* ***Example****:*



* ***Note****: The professor interface is also a proposal. It means that your own implementation can be different. What does matter is that the game functionality will be respected.*
  1. **Defining the Components**

**List of components / Functionalities and Behaviors**

*Include the list of components that you will use (they can be from Swing or JavaFX).*

*What are the behaviors and functionalities that you will provide? How these elements are related with functionalities.*

***Example****: The game mode can be selected by RadioButtons, etc.*

Below is a list of Swing components that will be used to build the interface for my NumPuz game:

* JButton: These will make up the tiles in the game area. As the user presses one of these buttons its directive will be to move to the open slot on the board provided the button is next to the open slot on the board. Other buttons will allow the user to switch between the play and design options whether the user would like to play a solution or design their own solution by moving the tiles (Buttons) at their own leisure.
* JMenu: The JMenus will allow the user to select drop down menus that will allow the user to save or load solutions, randomize the board for a new solution and reset a current solution. A second Menu can be created for the user to select for documentation on the game including instructions and developer info.
* JComboBox: This will be used to implement dimension selection. The user will be able to click on this box and select between the different options to determine a dimension size for the solution they would either like to play or create.
* JLabel: These will be used to display text and graphics including, the in-game timer and points system. It is possible the points system can be upgraded to a JProgressBar to display the users progress as a percentage instead of direct numerical points per correct tile. A JLabel can also be used to display on screen the users previous moves and if a move is invalid.
* JFrame: To display the game within.

**Details**

*Drawn your interface (ex: in an image from Paint / Powerpoint slide, or any sketch tool), describing:*

* *The components;*
* *The properties (ex: size, dimension, color, position, etc)*
* *Additional GUI components (ex: the layout to be used).*

Calendar

Description automatically generated

* 1. **User Manual**

**Basic cycle**

*Create a brief description about how your game can be used.*

***Example****: If you have to design the solution to be saved and played later, how are the stems. Most importantly, how someone can play the* ***NumPuz****.*

* *Note: your process do not need to be followed exactly when you are going to the implementation. For while, it is only a script about how to play.;*

Initially on start up the game will function like any other NumPuz type game. The user will be put into a random 3-dimension game (3x3 area). The user has a few options from here across the top bar of the window the user can select a game menu that will allow the user to save, load, randomize, or reset their current game. If the user chooses to randomize or reset, they will be prompted if they are sure they would like to close the current game. If the user would like a more challenging game they can use a drop down menu to change the dimension of the game, (3x3), (5x5), ect. Additionally, the user can also select the help menu which will give the user options to look at the instructions for the game or the developer info. If the user would like to play the random game given to them on start up the user can click on squares next to the open space to move them. If a user clicks on an invalid button, the button will shake or give some indication that the move they attempted is invalid. A notice of an invalid move will be put into the console output on the right side of the screen. If the user would like to choose between play mode or design mode they can be selected using buttons located under the game play area.

**FINAL SUGGESTIONS**

*Here some ideas to think about your language....*

* *Try to create a game whose execution can be very intuitive (easy to be played).*
* *Remember that this game will be in fact implemented only in the next assignment.*

**References**

“A visual guide to swing components,” *A Visual Guide to Swing Components (from: The Java™ Tutorials > Graphical User Interfaces > Swing Features)*. [Online]. Available: https://web.mit.edu/6.005/www/sp14/psets/ps4/java-6-tutorial/components.html. [Accessed: 18-Sep-2022].

Algonquin College

Fall, 2022