Algonquin College Logo

# SCHOOL OF ADVANCED TECHNOLOGY

### ICT - Applications & Programming

### Computer Engineering Technology – Computing Science



A11

Game Interface

Team:

Isaac Proulx- Id:041007853

Game Proposal - Picross

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

**EXPLANATION**

This document describes the GUI that will be used for the picross game.

What the main GUI should look like:

* 1. **Defining the Components**

**List of components**

I’ll be using JavaFX for the GUI.

* Top Panel
  + Implemented as a VBox
  + Contains the list of numbers that specify how many squares in each column should be filled
    - The numbers are implemented as StackPane nodes containing Text nodes
  + The panel should automatically resize to work with any game size
  + The width of the panel should be the same as the width of the game panel
  + The height of the panel should be the same as the width of the left panel
* Left Panel
  + Implemented as an HBox
  + Contains the list of numbers that specify how many squares in each row should be filled
    - The numbers are implemented as StackPane nodes containing Text nodes
  + The panel should automatically resize to work with any game size
  + The height of the panel should be the same as the height of the game panel
  + The width of the panel should be the same as the height of the top panel
* Options Panel
  + Implemented as a StackPane
  + Contains a ChoiceBox that allows the user to select many different options
  + The width of the panel should be the same as the width of the left panel
  + The height of the panel should be the same as the height of the top panel
* Game Panel
  + Implemented as a TilePane
  + Contains a grid of squares
    - The squares will be StackPane nodes that can be filled (bg colour) or marked with an “x” (Text node)
* Control Panel
  + Implemented as a VBox
  + Contains
    - Points counter
      * StackPane containing a Text node
        + Shows the current number of points
    - History Area
      * VBox containing Text nodes
        + Logs each action performed by the user
    - Timer
      * StackPane containing a Text node
        + Times the user on how long they take to solve a puzzle
    - Reset Button
      * Button
        + Resets the current game
    - Mark Box
      * CheckBox
        + Switches between mark and fill mode

**Functionalities and Behaviors**

Functionalities:

* Options Menu (Options Panel)
  + Game Mode:
    - A submenu (ChoiceBox)
    - Options
      * Design
        + Switches to design mode so you can create your own puzzles
      * Play
        + Switch to solve mode so you can solve puzzles
  + New Game
    - A submenu (ChoiceBox)
    - Options
      * Board Dimensions (some sort of input field to specify rows and columns)
      * Generate button (Button) generates new game with specified dimensions
  + Load
    - Opens a file selector and loads a puzzle from the selected file
  + Save
    - Saves the current design to a file (only usable in design mode)
  + Language
    - A submenu (ChoiceBox)
    - Options
      * English
      * French
  + Reset
    - Resets the current game, this option is also accessible through the control panel
* Game Panel
  + Left clicking a square will fill it in
    - When in mark mode the square will be marked instead
  + Right clicking a square will mark it as empty
  + Making incorrect moves will result in a lower score
  + Marking does not count as making a move
* Control Panel
  + Reset Button
    - Resets the timer and the game board (but keeps the same solution)
  + Mark Option
    - Switches between marking mode and filling mode

Behaviors:

* The current points can be seen at all times in the control panel
* All actions performed by the user can be seen in the control panel
* The amount of time spent on a puzzle can be seen in the control panel

**Languages**

English and French

**Details**

Diagram

Description automatically generated

* 1. **User Manual**

**Basic cycle**

The game is played by filling in all of the correct tiles, the numbers above the board specify the length of each block of filled tiles in each column, the numbers to the left of the board specify the length of each block of filled tiles in each row, there must be at least 1 empty tile between each block.

Tiles can be filled by left clicking them while in the fill mode (this is the default mode). Tiles can be marked as empty by either right clicking them or by left clicking them while in mark mode. You can switch between fill mode and mark mode by checking or unchecking the mark check box in the control panel.

While playing you will be timed on how quickly you can solve the puzzle. You will also be able to view your current score in the control panel.

You can reset the game by clicking the reset button in the control panel, this will reset the current game (it will still have the same solution): all tiles will be cleared, the timer will be reset, all mistakes will be forgotten.

You can generate a new game by selecting the new game option in the options menu, you will be asked for the dimensions of the new game, and then a new game of the given dimensions will be generated.

You can change the language in the options menu. Supported languages are: English and French.

**References**

* <https://fxdocs.github.io/docs/html5>

Algonquin College

Winter, 2023