**Design Decision Made**

**Milestone 4**

By: Adam Prins 100 879 683

|  |  |
| --- | --- |
| **Design Decision** | **Why it Was Made** |
| JumpIN GUI | The GUI was changed so that the creation and the action listeners are more compartmentalized. This was achieved with the use of the creation methods and the actionOn methods. This allows for butter understanding of the creation order for elements, and where to go to diagnose their specific implementation. This is very helpful for the following Design Decision. |
| Level Creator GUI | The Level creator GUI is a port of the original JumpIN GUI. It was decided that the GUI had the base of most of what we needed for the level builder. With the new creation methods, removing unused elements and configuring the existing ones was streamlined.  The goal was a simple selection toolkit for users to place the pieces they want onto a blank canvas.  Once the user is satisfied, they can select the save button to add the new puzzle to the XML of existing puzzles. There is a check for solvability. |
| Level Builder | The level builder takes static methods to determine if a game configuration is solvable, and if it is, the piece configuration is turned into an XML and appended to the end of the list of puzzles. |
| Save features | The save feature in the JumpIN GUI is able to save the current puzzle number, the position of the Pieces, and both the undo and redo stacks. You are then able to load the configuration from any other state. |
| Use of XML | XML was chosen because we had greater experience with the tools available for it. We believed that this would be advantageous and help us create a better project than with JSON. |