Sorry! Wellesley Edition

CS230 Fall ‘13

Final Project

Jazlyn Akaka and Alice Wong

Jazlyn Akaka and Alice Wong

CS 230 Fall ‘13

Final Project: Sorry! Wellesley Edition

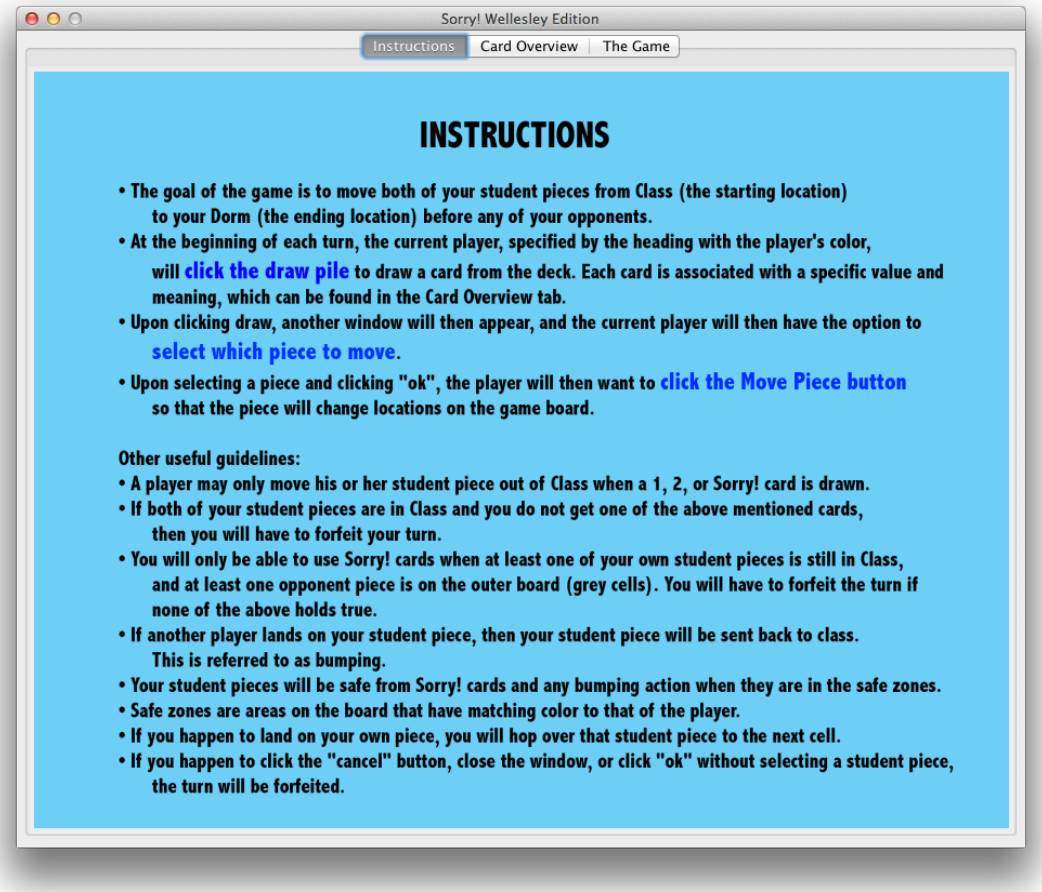
User Manual

Objective: Move all your pieces from CLASS to DORM. The first player to do this wins.

Players decide which color they would like to be from the choices of Yellow, Green, Red, and Purple, which represent the four class colors. Yellow is the first player to go, and is followed by Green, Red, and finally Purple.

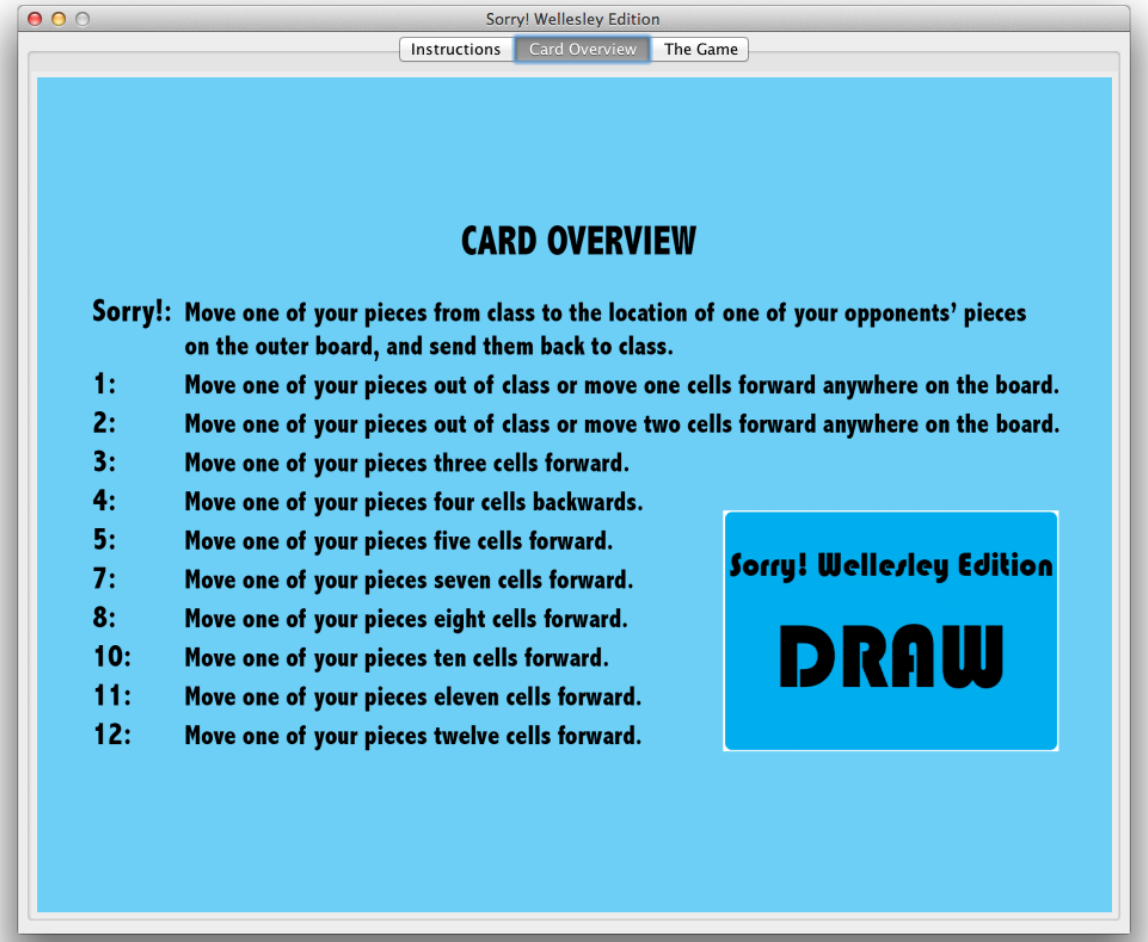
Instructions:

When the game is opened, the first window to appear is the Instructions panel of the SorryGUI. This panel includes the rules of the games, how to play, and any other guidelines that we would like the players to know.



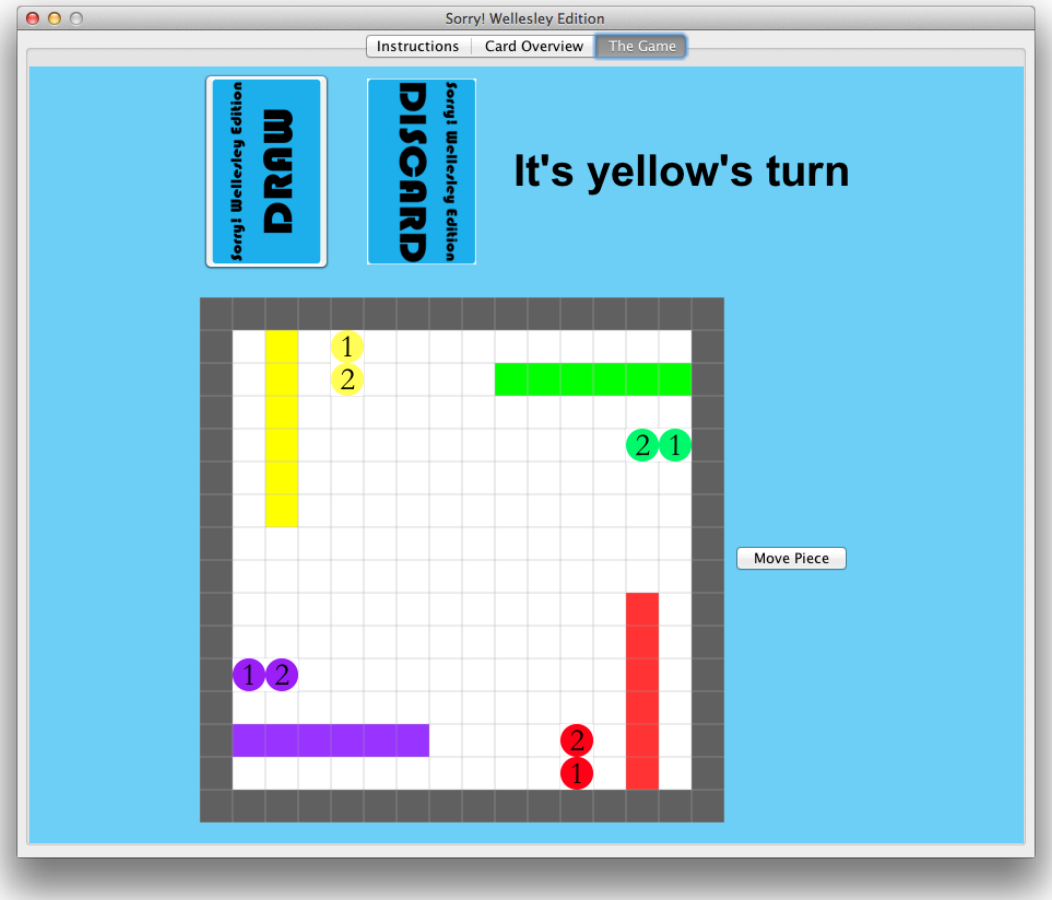
Card Overview:

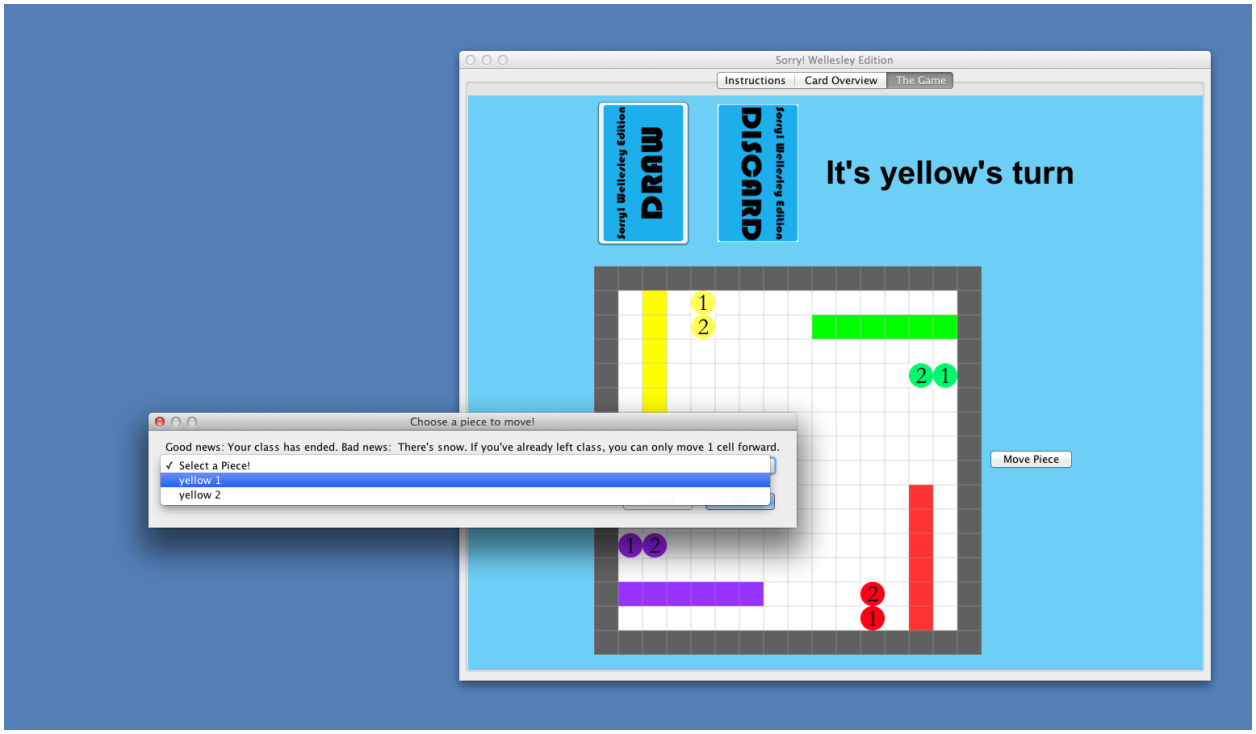
The next window that is available for the players to look at is the Card Overview tab. In this panel of the SorryGUI, players can look up what happens when they draw a certain card.

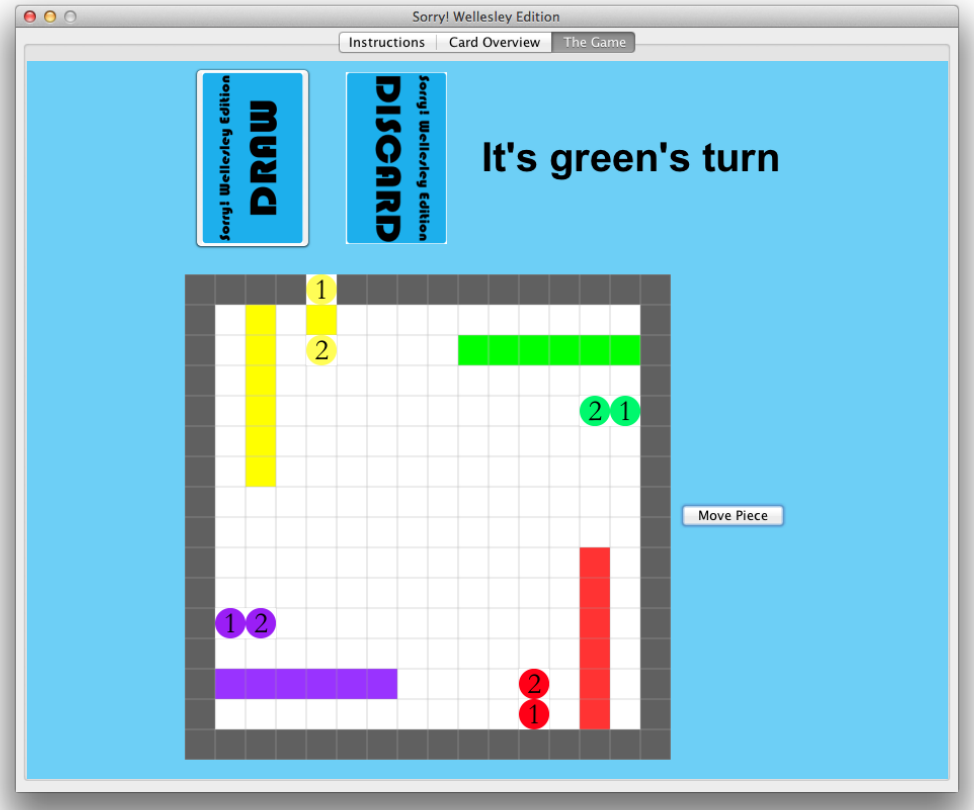


The Game:

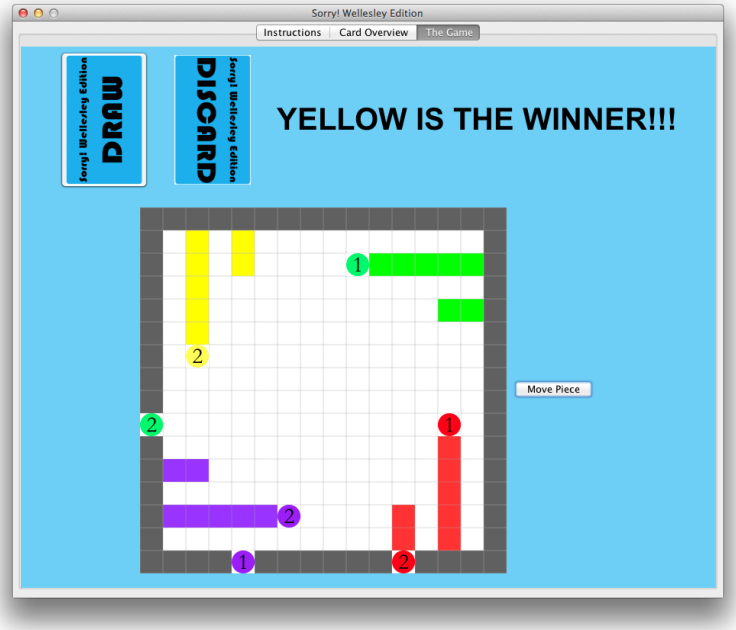
The third, and last tab, is The Game tab, where players get to play the game. At the beginning of each turn, a player will draw a card by clicking the draw pile. When they do this, a dialog box will open asking the player to select a piece to move. If no pieces are available to move, then the player is forced to forfeit their turn. After a player selects a piece, they will click the Move Piece button so that their piece will move to a new cell on the board (if applicable). It will then be the next players turn.




Sorry! Wellesley Edition
Instructions
Card Overview
The Game
DRAW
DISCARD
It's yellow's turn
Move Piece
Choose a piece to move!
Good news: Your class has ended. Bad news: There's snow. If you've already left class, you can only move 1 cell forward.
Select a Piece!
yellow 1
yellow 2


Sorry! Wellesley Edition
Instructions
Card Overview
The Game
DRAW
DISCARD
It's green's turn
Move Piece

At the end of the game when there is a winner, the color of the player who won will show up at the top in the announcement of who the winner is. When Move Piece is clicked for the final time, only the second piece to get to a player's Dorm will appear as the piece that arrived first will be underneath that piece.



Jazlyn Akaka and Alice Wong

CS 230 Fall '13

Final Project: Sorry! Wellesley Edition

Technical Report

ADTs

1. Stack: The collection of unused cards and used cards are stored in a stack. When a player draws a card, it is popped off the top so that no one can pick up the exact same card until all cards have been used. We decided to use a stack because we want to take off the top card from the collection (last in, first out).

2. LinkedList: There are a few different instances where we used LinkedLists. The first is in the SorryDeck class where we store the full deck in a LinkedList. This is so that we can randomly order the cards and then add them back to the stack of unused cards. The second instance where we used a LinkedList is in the SorryGame class. We used a LinkedList to help make the options for the drop down menu in the dialog box (which we stored in an array), and to make the corresponding array to the piece that is selected.

3. Graph: We use a graph to implement the board. Each cell on the board is a vertex, and has a cellNumber and a color associated with it. All cells have at least one arc. The cells that make up the outer board all have predecessors and successors and assorted arcs connect the vertices, making it a connected graph. We decided to use a graph because this will enable us to effectively move pieces around the board. A piece can only enter its own Dorm if the cell's color matches that of it's own.

Classes

SorryCard: Each card object is representative of its value and meaning. Appropriate values are 1, 2, 3, 4, 5, 7, 8, 10, 11, 12 and Sorry!, and the meanings associated with each card are either to move forward, backward, or bump. In order to make the game more "Wellesleyfied" we attached meanings that were scenarios that could happen around the Wellesley campus. This class includes methods for getting the value and meaning of the card.

SorryDeck: A collection of 45 cards total. There are five cards with value 1 and four each of the remaining values. We shuffle() the deck, which rearranges the order of cards in the deck, and stored in a stack. There is a method called draw(), which pops the card off the stack. When the stack becomes empty, draw() invokes shuffle(), and the stack becomes full, but this time with a rearranged order of

cards.

SorryPiece: Each piece object is representative of its color in a string, and its locations (oldLocation and current location). There are getters and setters to get and set colors and locations.

SorryPlayer: Each player object has two pieces of the same color. There are getters to get a player's piece.

SorryCell: Each cell has a string representing its color, so that a piece may reach its respective Dorm instead of continuing around the board. We have a getter to get the color of the cell to see if it is the current player's Dorm.

SorryBoard: The board is made up of sixty outer cells, eight Class cells, and twenty four Dorm cells (each Dorm has six cells, which is part of a player's safe zone). The board will use AdjMatGraphPlus.java/GraphPlus.java.

SorryGame: Consists of one board object, four player objects and one deck object. There are multiple methods that go into moving a piece around the board to Dorm, some of which include, forward(), backward(), movePiece(), playOneTurn(), etc. In playOneTurn(), a player draws a card, selects a piece, and moves accordingly. We also check to see if there is a winner after each turn.

RulesPanel: A panel with the instructions of the game. Users can read this before or during the game without disrupting a game in progress.

SorryCardOverviewPanel: A panel with the specification of each card, and the possible movements of player's piece depending on the value of the card.

Grid Panel: A panel that creates the visual representation of the board, and includes a button to move the current player's piece.

SorryGamePanel: Visual representation of the board with the pieces and deck of cards. Users interact with this panel to move pieces around the board and draw cards by clicking the draw button. A dialog box appears once the draw button has been clicked so that a player may select which piece to move.

SorryGUI: Puts together all the panels using three tabbed panes.