FIT1040 Assignment 2

About This Application

This application is a minesweeper game implementation in Scribble, in which the primary objective of the player is to reveal all non-mine squares in a grid of NxN. The player wins if all squares except the mines has bee revealed at the end of the game, while the player loses immediately if a mine has been clicked on.

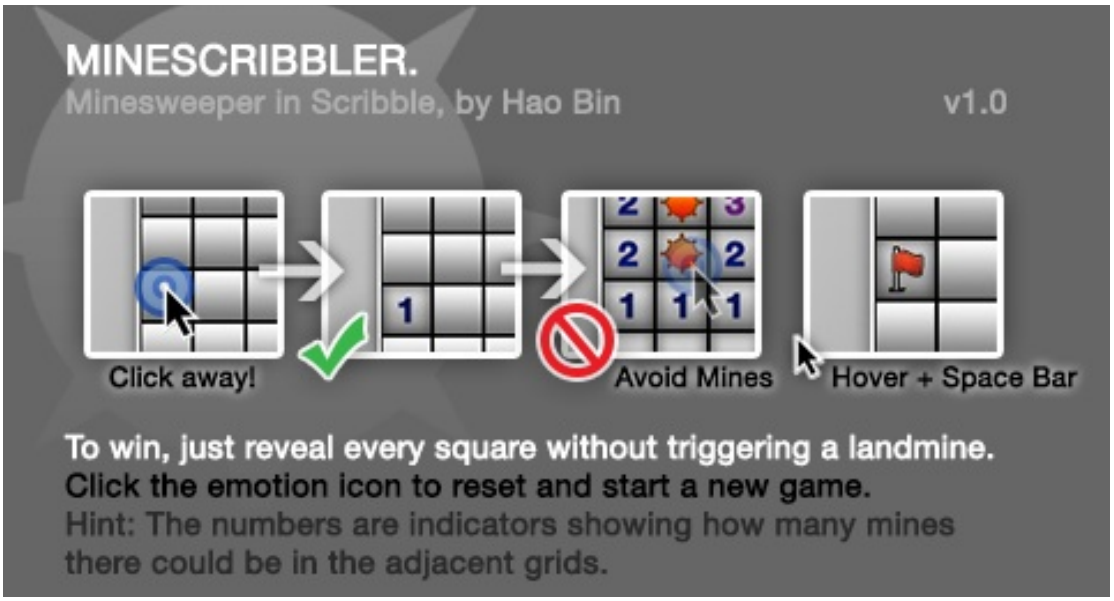
In a typical minesweeper game, when the player clicks on a square, he would expect to uncover one out of these three types of outcomes: a number, an empty square or a mine. The number revealed in the square are indicators of how many mines there could possibly be in the adjacent squares, all 8 directions of the current square. So a number square of 1 would mean there is at most one mine in the surrounding 8 squares, and a number square of 8 would mean all the surrounding 8 squares are mines.

In this game, when the green flag is clicked, a help screen will be displayed showing a brief guide on how to play this game, while the game loads. After the game loaded, the game will inform the player that it is ready to play, and allows the grid to be clicked on. The player can then click on any of the grids to begin the game. When the player won or lost the game by the condition stated above, a message will be displayed to the user, and asks the player whether does he want to play again.

At any point the player require hints on how to play this game, he can click on the question mark at the top right corner in order to display the help screen again.

If the player wants to restart the game, he can do so by clicking the smiley face icon at the left side of the screen.

There is also game statistics displayed at the left side of the screen, in which Wins, Loses, and the winning rate of the player is displayed. At the bottom of the screen is a status bar and progress bar showing the game status.



Overview of Workings

This application utilises 2 main lists to represent its game states. An additional temporary list ('tmp') is used at the beginning to initialise the game. The two main lists are Board, and mines.

Lists

Board represents the current state of the game grid, it is a list of lists which is made up of 100 index to represent each of the squares in the game (e.g Index 1 is Row 1 Column 1, and Index 100 represents Row 10 Column 10). Each index contains another list, which stores two types of data inside. The first index of the inner list stores the type of the square (i.e, a mine is represented by a 9, empty squares are represented by 0, and numbers are [1-8]), while the second index represents the status of the square (i.e, 0 represented unopened, 1 represents opened, 2 represents flagged).

The mines list store a list of mines existing in the current game board, and is used in determining the winning state of the game. One of the way to determine the winning state of minesweeper is if

{ NumberOfOpenedSquares = TotalNumberOfSquares – Number of Mines }

Initialising

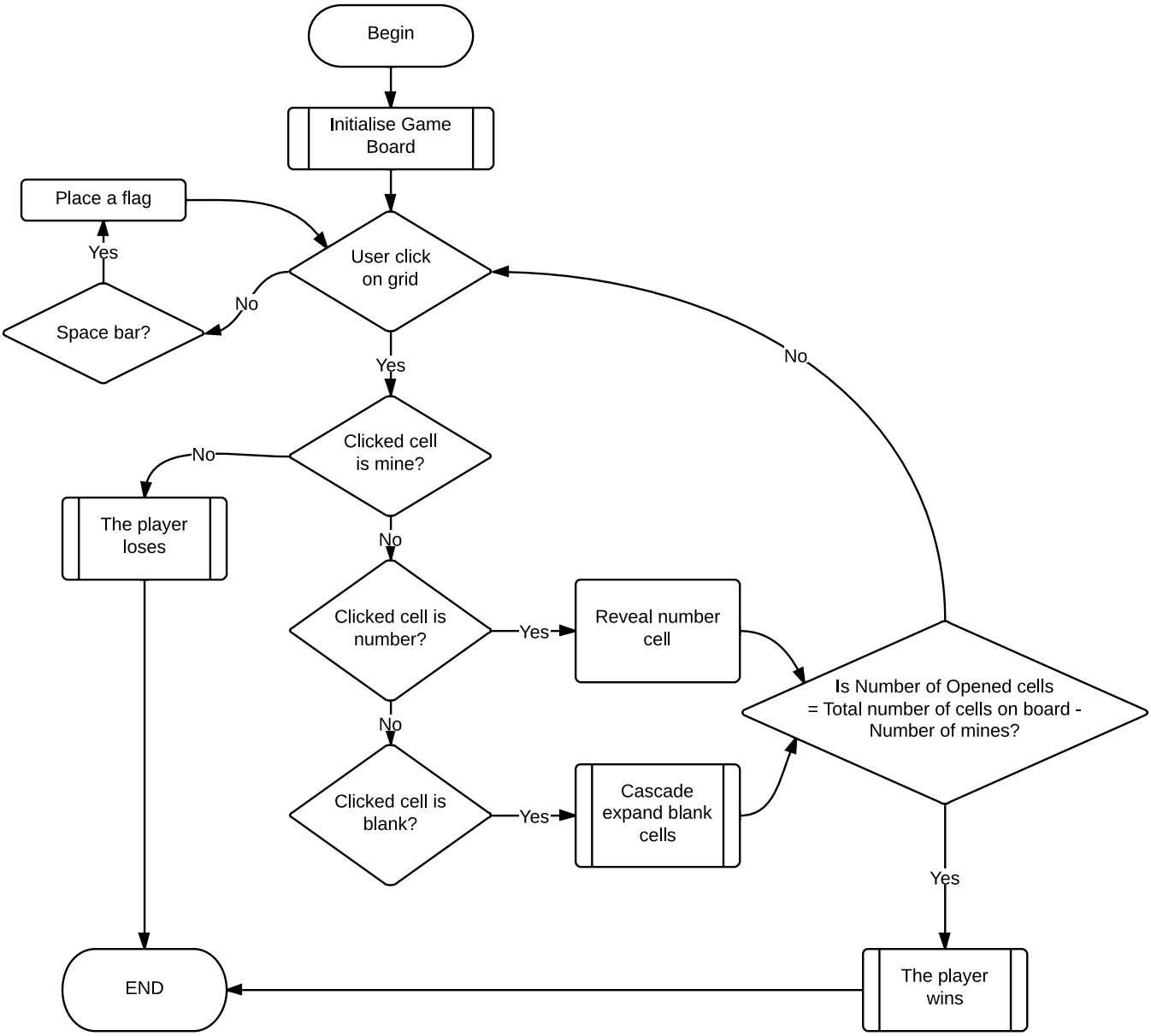
When the game is initialised, it resets the value of the variables such as *numOfMines* and *Opened*, clears the list of *Board* and *Mines*, then it starts by placing n mines at random positions of the board. Then we go through the *board* lists, and check if there is a mine in that index, we increment the adjacent squares (except adjacent mine squares) by 1. Then we set *NumOfMines* to the length of the *Mines* list. Once that is done, the game is ready.

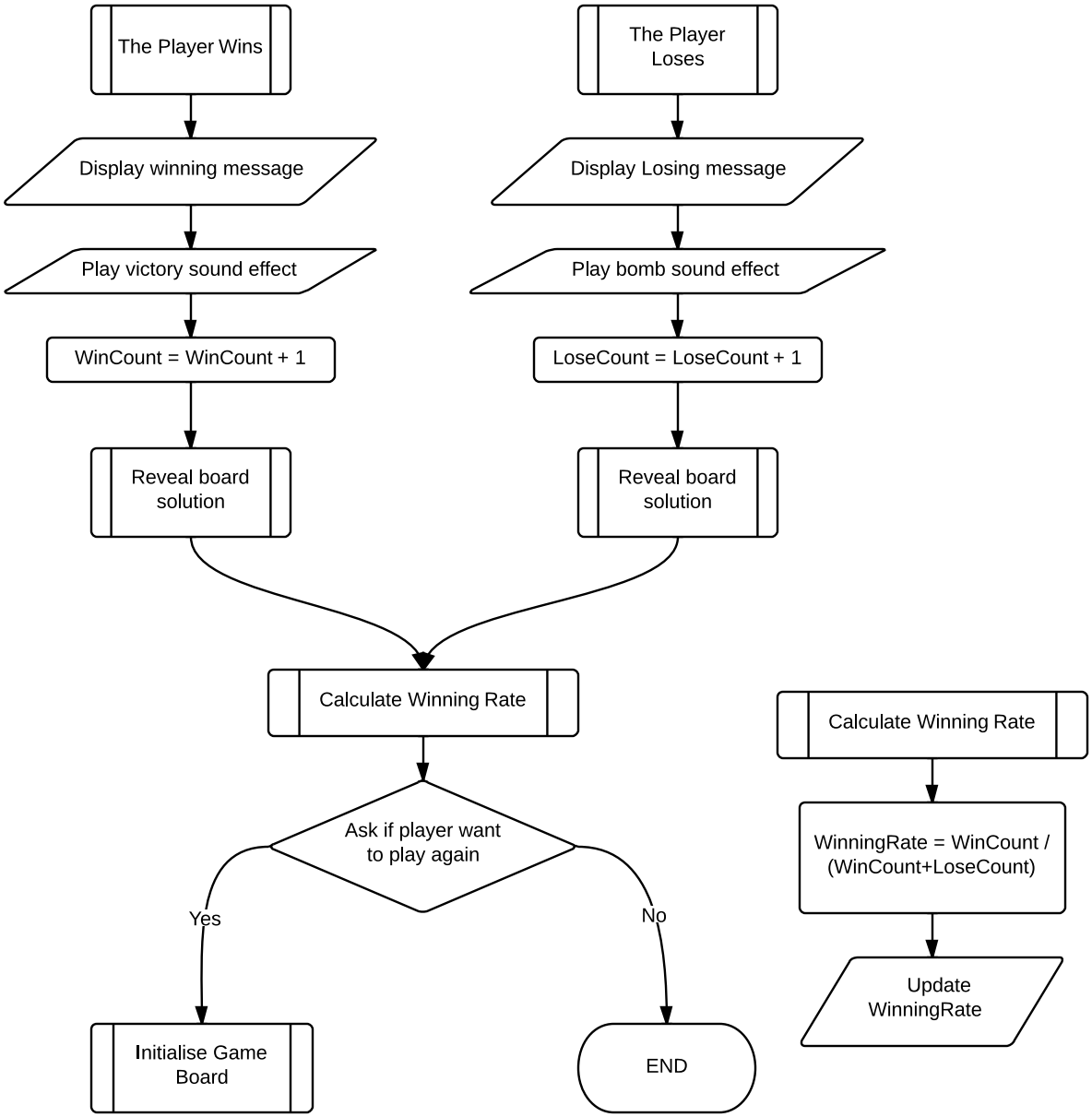
Gameplay

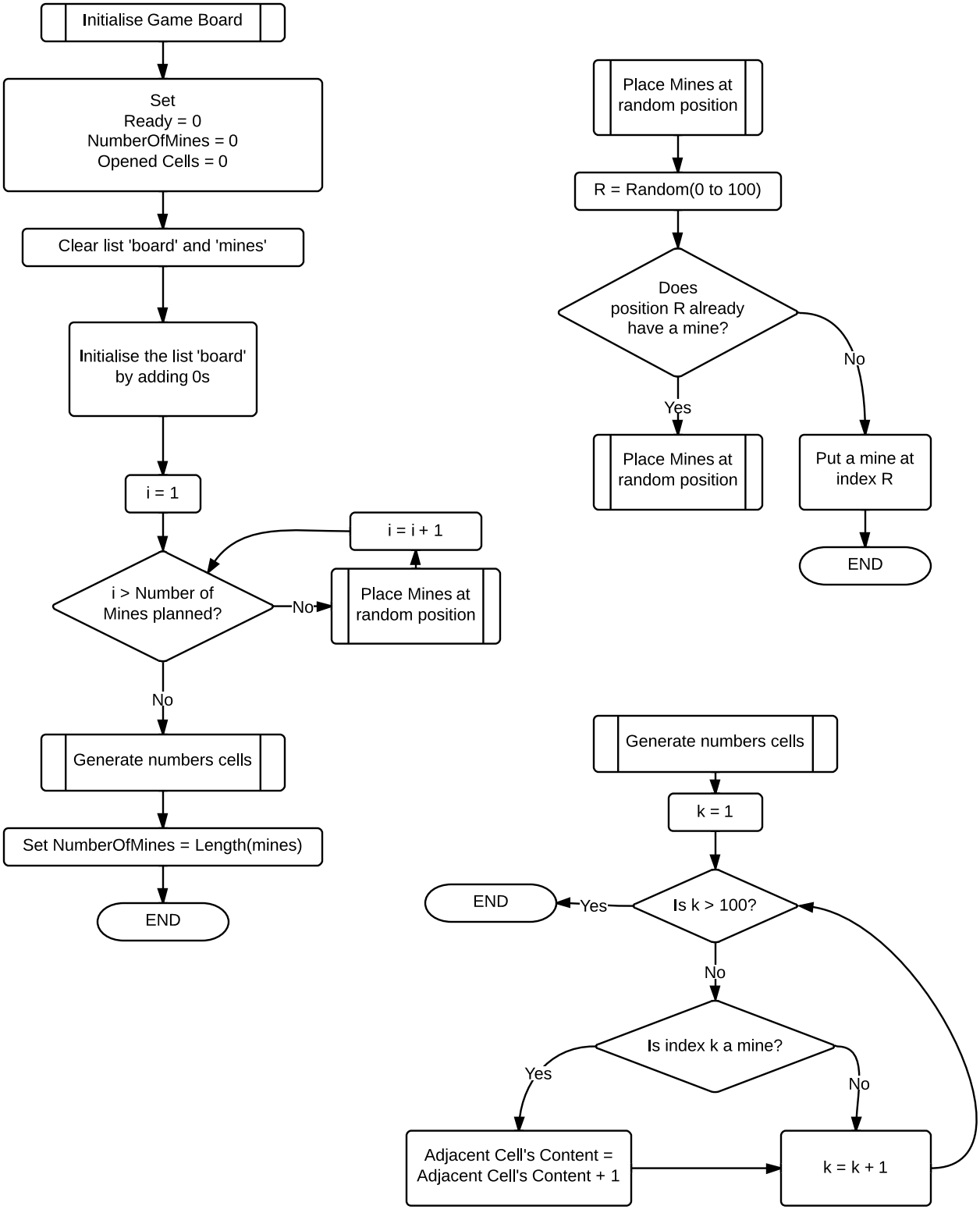
When the user clicks in the game board, the current x and y position is stored and calculated, and then we move the tiles stamper to the appropriate x and y position. We then calculate the *boardindex* value through formula (column+10*(row-1)). Then we check if the item at *boardindex* position is opened or not, if not, then we set it to opened, and change to the corresponding costume and stamp. Then we increment NumberOfOpenedSquares by 1. If the item at boardindex is a blank square, we then run the Cascade(boardindex) custome block, which recursively goes through adjacent squares and check whether are they empty or not, then reveals a contiguous area of empty squares.

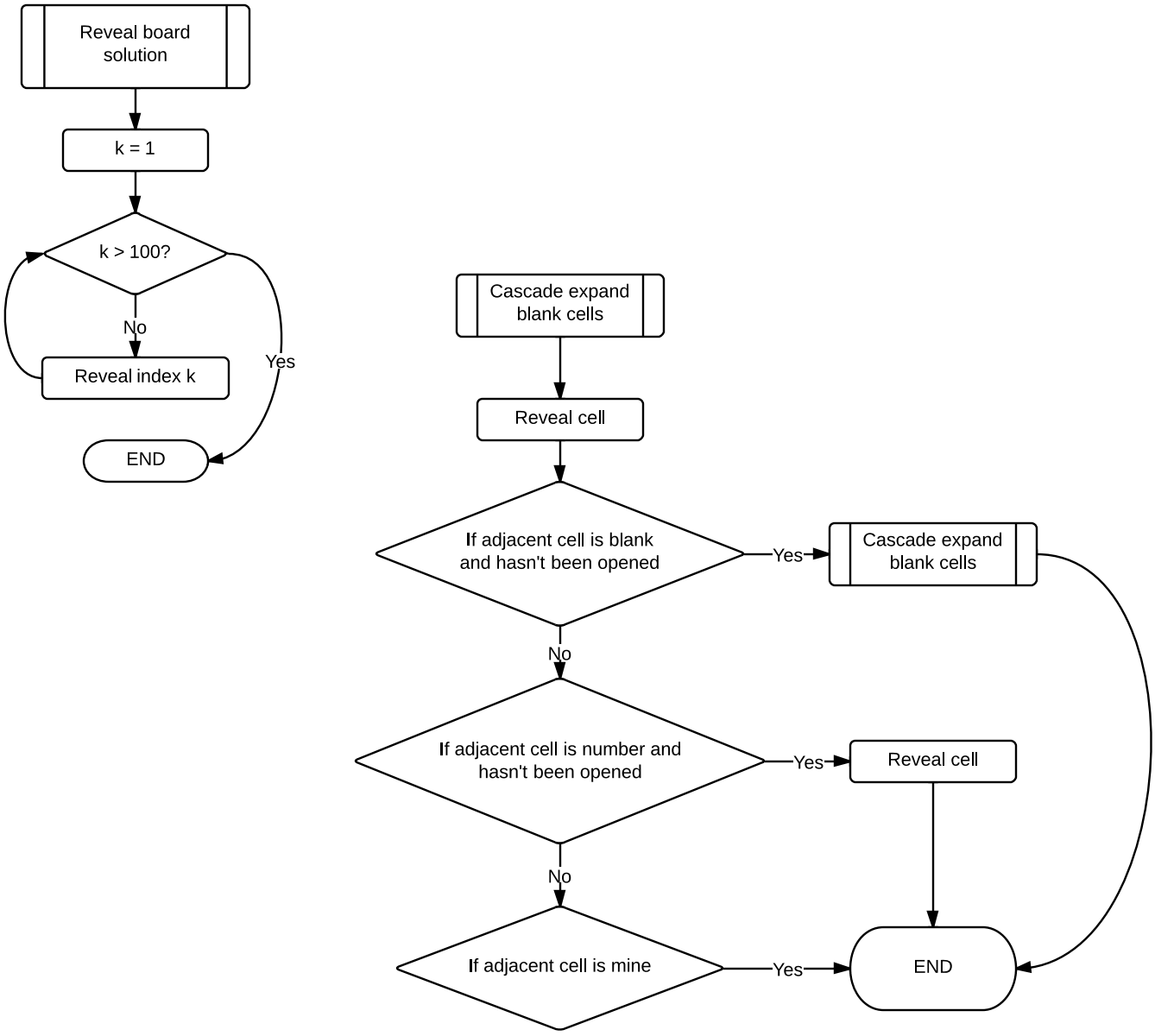
Now, if the item at boardindex is a mine, then we broadcast a message to initiate the Lost The Game sequence. A bomb sound effect would be played, and then the solutions would be revealed, and a message will be displayed. We also increment the lost count by 1 and recalculate the winning rate.

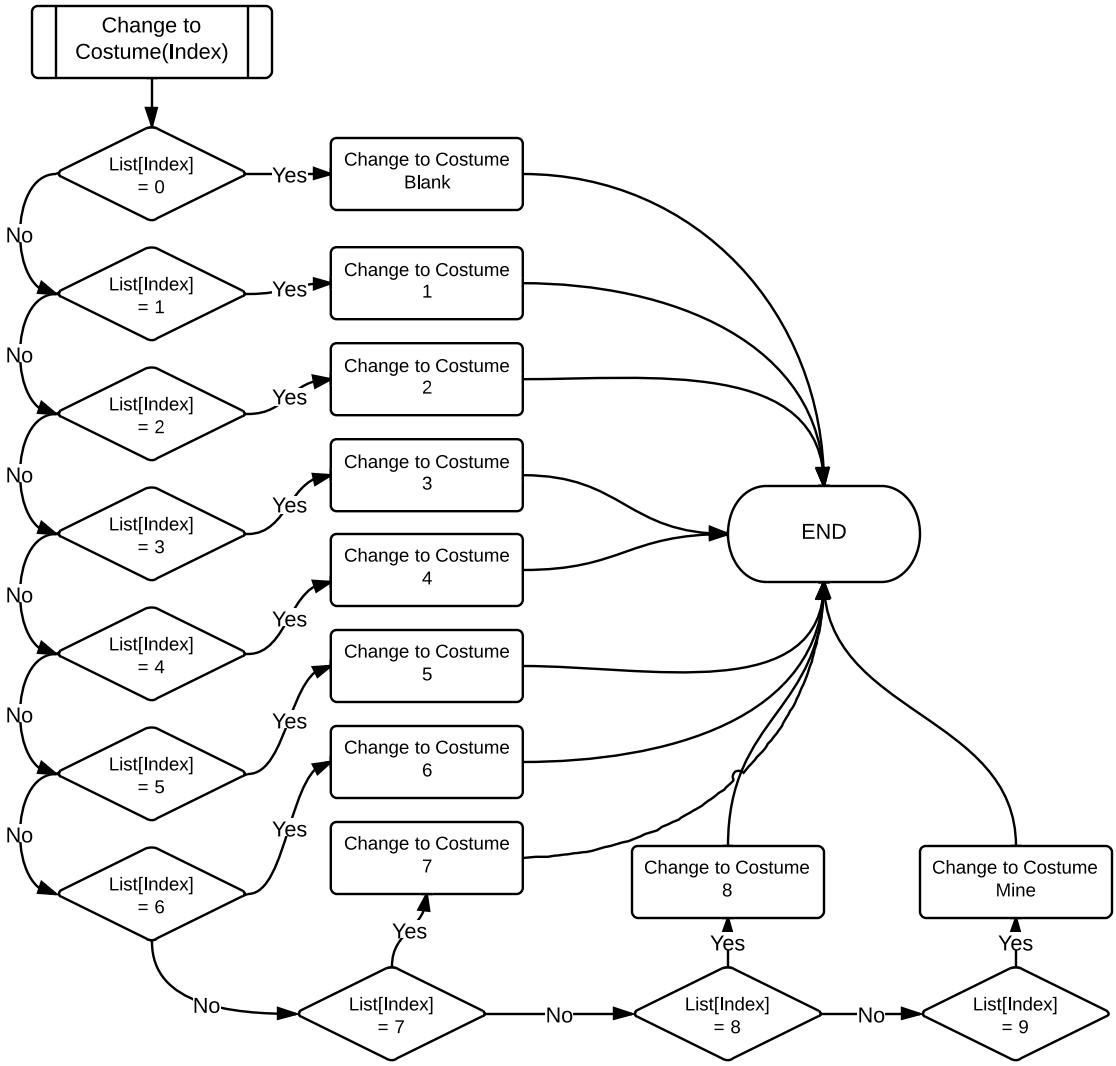
Level 0


Begin
Initialise Game Board
Place a flag
Yes
Space bar?
No
User click on grid
Yes
No
Clicked cell is mine?
No
The player loses
No
Clicked cell is number?
Yes
Reveal number cell
No
Clicked cell is blank?
Yes
Cascade expand blank cells
Is Number of Opened cells = Total number of cells on board - Number of mines?
Yes
END
The player wins


The Player Wins
Display winning message
Play victory sound effect
WinCount = WinCount + 1
Reveal board solution
The Player Loses
Display Losing message
Play bomb sound effect
LoseCount = LoseCount + 1
Reveal board solution
Calculate Winning Rate
Ask if player want to play again
Yes
No
Initialise Game Board
END
Calculate Winning Rate
WinningRate = WinCount / (WinCount+LoseCount)
Update WinningRate


Initialise Game Board
Set
Ready = 0
NumberOfMines = 0
Opened Cells = 0
Clear list 'board' and 'mines'
Initialise the list 'board'
by adding 0s
i = 1
i = i + 1
i > Number of
Mines planned?
No
Place Mines at
random position
No
Generate numbers cells
Set NumberOfMines = Length(mines)
END
Place Mines at
random position
R = Random(0 to 100)
Does
position R already
have a mine?
No
Yes
Place Mines at
random position
Put a mine at
index R
END
Generate numbers cells
k = 1
END
Yes
Is k > 100?
No
Is index k a mine?
Yes
No
Adjacent Cell's Content =
Adjacent Cell's Content + 1
k = k + 1


Reveal board solution
k = 1
k > 100?
No
Yes
Reveal index k
END
Cascade expand blank cells
Reveal cell
If adjacent cell is blank and hasn't been opened
Yes
Cascade expand blank cells
No
If adjacent cell is number and hasn't been opened
Yes
Reveal cell
No
If adjacent cell is mine
Yes
END

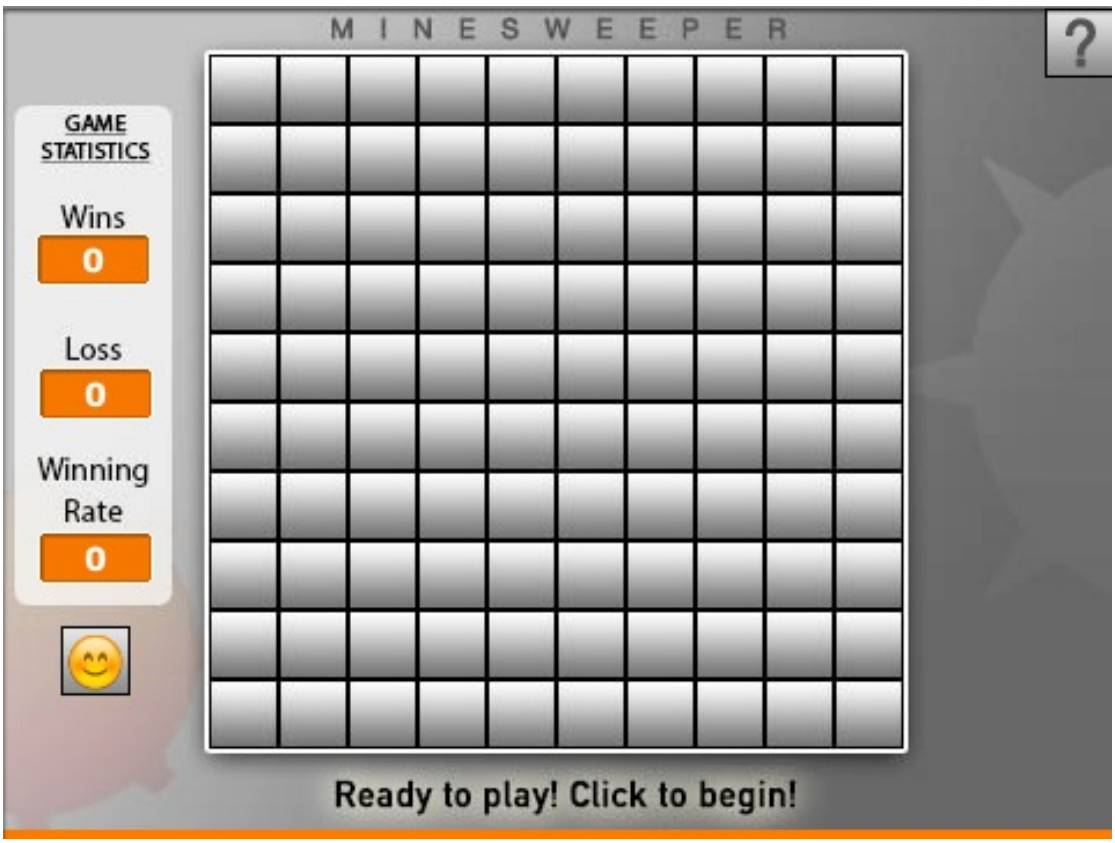

Change to Costume(Index)
List[Index] = 0
Yes
Change to Costume Blank
No
List[Index] = 1
Yes
Change to Costume 1
No
List[Index] = 2
Yes
Change to Costume 2
No
List[Index] = 3
Yes
Change to Costume 3
No
List[Index] = 4
Yes
Change to Costume 4
No
List[Index] = 5
Yes
Change to Costume 5
No
List[Index] = 6
Yes
Change to Costume 6
No
List[Index] = 7
Yes
Change to Costume 7
No
List[Index] = 8
Yes
Change to Costume 8
No
List[Index] = 9
Yes
Change to Costume Mine
END

Test Plan

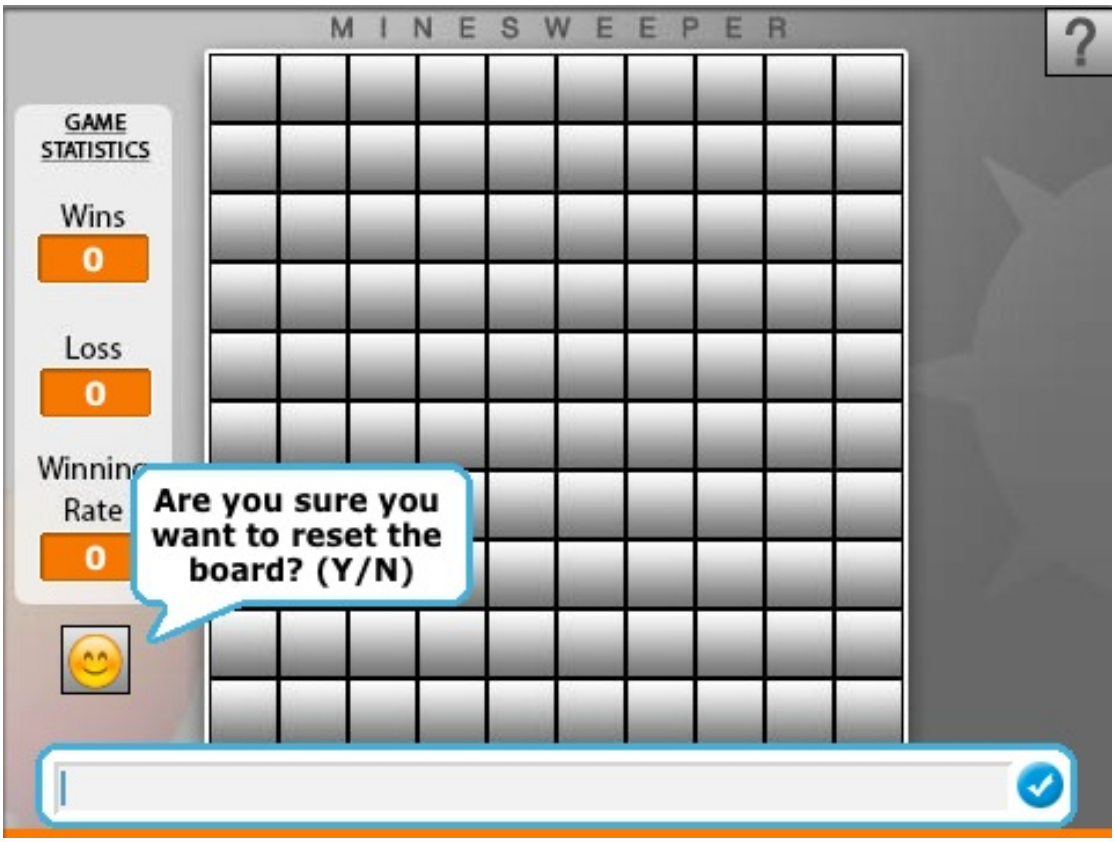
#	Description	Expected Result	Pass
1	Clicking the green flag to start the game	• A help screen/splash screen should slide down from the top of the canvas. • The status message should display that the game is loading, with an orange progress bar. • After the progress bar finishes loading, the status should change to "Ready to Play! Click to begin."	✓
2	Clicking the OK button in the help screen/splash screen.	The help screen should slide up and reveal the game grid, while the question mark button slides down from the top.	✓
3	Clicking on any of the squares in the game grid.	A 'pop' sound should be played, the smiley face icon should change to a horrified expression, and followed by one of the following 3 outcomes: • Display a number • Display a mine • Display empty square(s)	✓
4	Clicking on a number square	It should play a pop sound, the smiley face icon changed to a horrified expression, and then a number will be displayed. Then the smiley face will return to smiling.	✓
5	Clicking on a mine square	It should play a bomb sound, the smiley face icon changed to a horrified expression, and then a message saying LOSE should appear, then the game solutions will be revealed. The smiley face should show a sad face. Then the status should change to "You Lost! Play Again?" The Loss count at the sidebar should be incremented by 1, and the winning rate be calculated accordingly.	✓
6	Clicking on an empty square	It should play a pop sound, the smiley face icon changed to a horrified expression, then adjacent empty squares will be revealed as well (if there are any), and the numbers squares at the edges of this contiguous empty square region will be revealed.	✓
7	Clicking on smiley face	It should ask the player whether does he want to reset the game.	✓
8	Answering "Y" following Test #7	The game should be resetted, and be reinitialised. The status would change to Loading. The win and loss count should retain.	✓
9	Answering "N" following Test #7	Nothing should happen	✓

10	Answers other than "N" or "Y" in Test #7	It should ask the user to put in only Y or N.	✓
11	Revealing all the non-mine squares	A "WIN" message should be displayed to the user, and victory music played. A flag should appear in the background, and status be changed to "Congratulations, you win!". The smiley face should be changed to a face with sunglasses. The Win count at sidebar should be incremented by 1, and winning rate be updated accordingly.	✓
12	Clicking the "Click here to Play again" in WIN or LOSE messages	It should reset the game board, and reinitialise the game. The status should change to loading. The win and loss count should be retained.	✓
13	Putting the mouse cursor on top of an unrevealed square, and then tapping the space bar	The square should change into a flag, showing that it's marked.	✓
14	Following Test #13, put the mouse cursor on the same square and press space bar again.	The marked square should change back into an unrevealed square.	✓
15	After revealing a square, clicking the square again	Nothing should happen. The square should stay intact.	✓
16	Clicking the Question Mark icon at top right	The help screen should slide down, and clicking on the help screen should not cause a move in the game grid.	✓
17	Calculating Winning Rate	The winning rate should reflect the calculation of { Win Count/(Win Count+Lose Count) }	✓
18	Clicking outside the game grid	Nothing should happen.	✓
19	After winning or losing a game, press the green flag again	Following #1, a help screen should appear, and then the game be resetted and reinitialised accordingly, with no abnormalies happening resulting from previous game.	✓

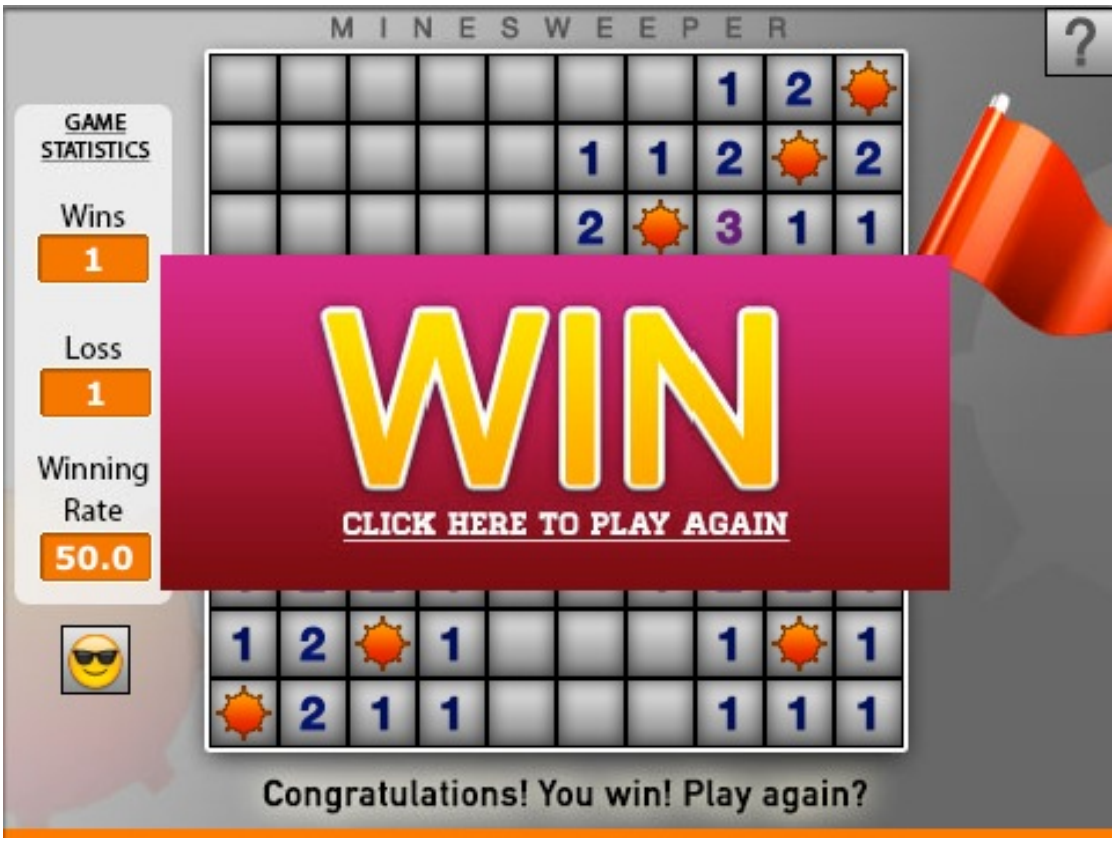
Screenshots



MINESWEEPER
GAME STATISTICS
Wins
1
Loss
5
Winning Rate
16.7
Ready to play! Click to begin!


MINESWEEPER
GAME STATISTICS
Wins
0
Loss
0
Rate
0
Are you sure you want to reset the board? (Y/N)

MINESWEEPER
Stop everything
GAME STATISTICS
Wins
0
Loss
1
Winning Rate
0
LOSE
CLICK HERE TO PLAY AGAIN
Too bad, you lost! Play again?


MINESWEEPER
GAME STATISTICS
Wins
1
Loss
1
Winning Rate
50.0
WIN
CLICK HERE TO PLAY AGAIN
Congratulations! You win! Play again?