**Requirements:**

Write up a design document (describe how your game will work) and Implement in VB a Bingo Card Game.  See video if you are not sure how to play the game.

Make sure to customize this bingo game to make it your own.

Grading for Final Project:

45 pts: Bingo Board with RANDOM numbers

5 pts: CORRECT random numbers on board and NO duplicates:

B: 1-15, I:16-30, N:31-45, G:46-60, O:61-75

5 pts: Button or other way to control game … “Next Bingo Ball Number”

5 pts: Random number generated for “next bingo ball” – NO duplicates

5 pts: Select Bingo number called on board (by you or the computer)

5 pts: NEW game feature to start over after win or in middle of game

20 pts: Check if user wins (horizontally, vertically, or diagonally)

5 pts: Original Work

5 pts: Good Design

**Design:**

NOTE: a mock-up for each interface was created and they may be found in the Vision file called “GUI's Mock-ups”.

The application will be made from three Window forms. The main form that will host the menu and the game. The Game Settings form that will allow the user to set the conditions for the game which in this case are the username, number of player and number of boards per player. And the Board form which will provide an illustration of the state of the board upon which the user has clicked. The main menu will host the “Play” button that calls the Game Settings form before proceeding to play and the “How to play” button which brings up the game instructions.

NOTE: Click on the “BINGO” label to change the game’s theme. You may also play around with the bingo board that changes position every 2 seconds by clicking it. Every click will increase the speed at which it changes its current position.

Once the player has called clicked the “Play” button and entered the game settings, the application will shift its view to what I like to call “The Battlefield”. There the user will be able to start playing against other player(s). The battlefield view hosts three buttons. A “Back” button, in case the user wants to go back to the menu. A “Restart” button, in case the user wants to re-start the game and choose different game conditions. And a “Start” button that will begin the game by showing off the first ball number (Once this button is clicked by the first time its text will change to “Next”).

Every time the user calls a new bingo ball the corresponding number will be automatically marked on the opponents’ boards. The user on the other hand will have to mark the numbers that correspond to the bingo ball’s number on his/her own. The user will not be allowed to call another ball until he/she has marked the corresponding number on the corresponding board if there’s any.

NOTE: The user may take a look at any board on the table throughout the whole game.

The game continue in this manner until a winner is found. Once the game has concluded and the winner is selected, the application will let the user know of this event and will allow the user to take a look at the board that brought about the winner. At this point the user may either close the game or start a new game. If the user does not do anything and tries to call another board, he/she will be prompted for an answer as to whether he/she want to play another game.

**Brief Look at Classes, Forms and Utilities:**

**Forms:**

* MainForm.vb: The main form of the application that will host the game events.
* GameSettings.vb: This form will host the values such as the username, number of player and number of boards per player that the user must provide before proceeding with the game.
* Board.vb: This form will illustrate the state of the board upon which the user has clicked.

**Classes:**

* Resizer.vb: This class will hold the application’s controls’ proportions in percentages. When the user resizes the window the controls’ proportions will change accordingly with the help of this class’ methods.
* SetterAndAnnouncer.vb: This class will be responsible for determining the numbers on the boards, tracking the marks made on each board throughout the game, announcing the next ball number and announcing the winner.

**Modules (Utility):**

* CommonFunctions.vb: This module will have utility procedures (e.g CenterForm to centralize the forms position on the screen before it is generated) that do not required a class instance so they can be used by all forms.