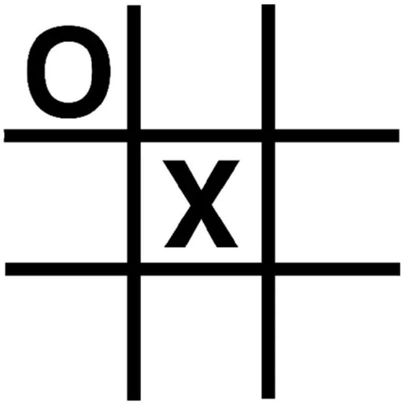
Project Proposal
Object Oriented Programming

Group Members :

1.Muhamad Farhan Rais Bin Noor Mohamed, ID : 20001518
2.Muhammad Khairun Nazrin Bin Ahmad Anuar, ID : 20001444

TIC TAC TOE

Overview



We chose Tic Tac Toe as our OOP Project. The game looks very simple but has interesting features like 2 players, multiple shapes, graphics and many more. This allows us to implement OOP fundamentals while programming it.

In real life, firstly, there will be two players. The players will choose if they want “X” or “O”. Then, there is a system to decide who starts first such as playing rock paper scissors. The first player will put his shape in one of the nine tiles that are drawn on a paper or board. Then, the next player makes his move. They do this until one of the players manages to make a pattern from his shape, of which the patterns are diagonal, vertical or horizontal. This will end the game with the player that got the pattern as winner.

Now that the concept of the game in real life is understood. We can deduce a programming flow from it. First, the system will ask the details of the first player, such as name and desired Shape, whether “X” or “O”. Then, the program will decide the first player by using a random function. The first player will use the mouse to click on the tile that they intend to put a shape on. The program will register the coordinate in a specific array. Then the program will call a function to highlight the tile. Now that the tile is highlighted, the current player can press “X” if their shape is “X” or “O” if their shape is “O”. This will draw their shape in the tile. Then, the next player can do the same. Once any player has drawn three shapes, the program will start checking for patterns. If there is a pattern found, the current player is the winner. There will be a score that will increment by one for the player. If there is no pattern found until all tiles are drawn on it, the game will be tied and both players will get 1 mark. The program will then ask if the players want a rematch or end the game. The player with the higher mark will be declared as game winner. In addition, there will also be a timer function for each player, so the player can think for only a specific time before time finishes and loses the round.

1. What are the concepts/things/objects in the game ... list them out

I. Player
II. Game Board
III. Time
IV. Score
V. Draw
VI. Shapes

2. For each concept/thing/object/noun .. what are the attributes/data?

I. GameBoard

+ size : int
+ backgroundColor : Color
+ tiles[9]:Shapes
- highlightCoordinate[]:int
- drawTool : Draw
- twoPlayers[2] : Player

II. Player

- name : String
- chosenShape : Shape
- score : Score
- time : Time
- shapeCoordinate[]:void
+ numberOfShapesDrawn : int

III. Time

-value : Time
-xpos : int
-ypos : int

IV. Score

-score : int
-position : String

V. Draw

-g : Graphics
-shapes[] : Shape
+numberOfShape : int

VI. Shapes

- name : String
- color : Color
- font : Font
- word : String
- size : int
- xpos : int
- ypos : int

3. For each concept/thing/object/noun, what are the related operations?

I. GameBoard

+ checkKeyPressed():boolean
+ checkMouseClicked():boolean
+ changeHighlightCoor():void
+ refreshBoard():void
+ checkWhosTurn():Player
+ addPlayer(Player):void
+ startGame():void
+ endGame():void
+ pauseGame():void

II. Player

+ changeName(String):void
+ choseShape():Shape
+ changeScore():void
+ changeTime():void
+ addShapeCoordinate(int,int):void
+ clickTile():void
+ putShape():void

III. Time

+countDown(int):void
+startTime():void
+changeValue(Time):void
+changePos(int,int):void

IV. Score

+changeScore(int):void
+changePosition(String):void

V. Draw

+makeAllShapes():void
+paint() : void
+paintTitle():void
+paintXO():void
+XShape():void
+drawOShape():void
+drawSquareTiles():void
+letters():void
+numbers():void
+highlightSquareTile():void
+drawShape(Shape): void
+drawTime(Time):void
+drawScore(Score):void

VI. Shapes

+changePos(int,int):void
+changeName(String):void
+changeColor(Color):void
+changeFont(Font):void
+changeWord(String):void
+changeSIze(int):void