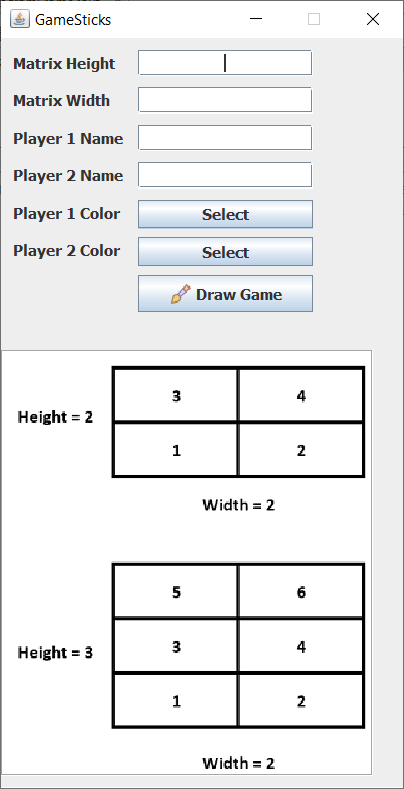
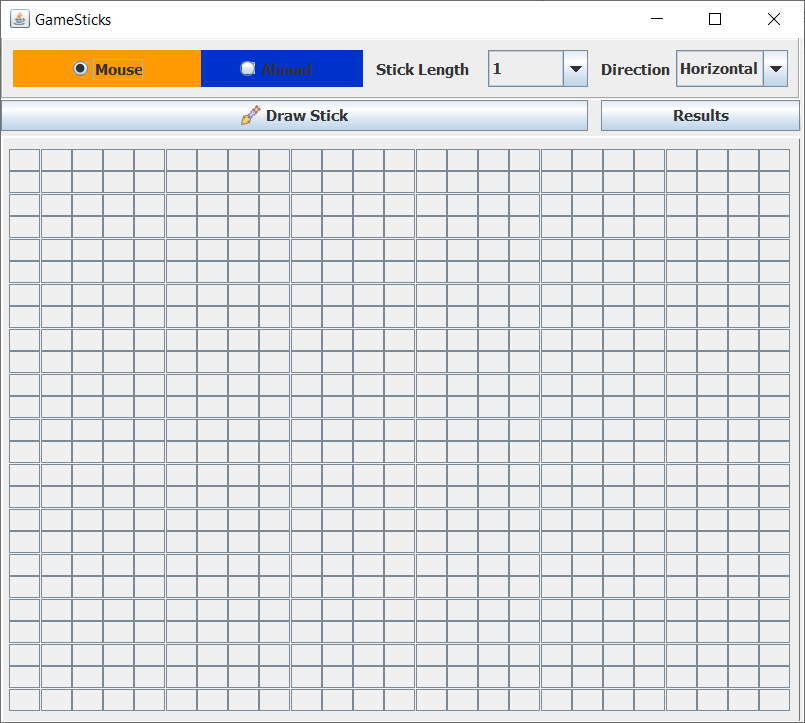
**Sticks Game Report**



**Prepared by :**

**Submitted to:**

**5/2020**

**Stick Game Idea:**

Two players draw stick vertical or horizental in matrix board. Each player can select the length of the stick befor drawing. Finally, the player who owns more cells in the board he is the winner.

**Project Java Classes**

1. **Players**

Class Declaration: *public class Players*

Class Variables:

* String playerName; (to name each player with his/her own name).
* int score; (to save number of squares has been accupaied by the player in the game).
* Color color; (to save the color for each player).

Java Code

import java.awt.Color;

public class Players {

private String playerName;

private int score;

private Color color;

/\*\*

\* @return the playerName

\*/

public String getPlayerName() {

return playerName;

}

/\*\*

\* @param playerName the playerName to set

\*/

public void setPlayerName(String playerName) {

this.playerName = playerName;

}

/\*\*

\* @return the score

\*/

public int getScore() {

return score;

}

/\*\*

\* @param score the score to set

\*/

public void setScore(int score) {

this.score += score;

}

/\*\*

\* @return the color

\*/

public Color getColor() {

return color;

}

/\*\*

\* @param color the c to set

\*/

public void setColor(Color color) {

this.color = color;

}

public void resetScore(){

score = 0;

}

}

1. **MatrixSize**

Class Declaration: *public class MatrixSize*

Class variables:

* int matrixHeight; (detect the height of the created matrix for the game).
* int matrixWidth; (detect the width of the created matrix for the game).
* int remainder; (stored the empty squares in the matrix).

Java Code

public class MatrixSize {

private int matrixHeight;

private int matrixWidth;

private int remainder;

/\*\*

\* @return the height

\*/

public int getMatrixHeight() {

return matrixHeight;

}

/\*\*

\* @param matrixHeight the matrixHeight to set

\*/

public void setMatrixHeight(int matrixHeight) {

this.matrixHeight = matrixHeight;

}

/\*\*

\* @return the matrixWidth

\*/

public int getMatrixWidth() {

return matrixWidth;

}

/\*\*

\* @param matrixWidth the width to set

\*/

public void setMatrixWidth(int matrixWidth) {

this.matrixWidth = matrixWidth;

}

/\*\*

\* @return the remainder

\*/

public int getRemainder() {

return remainder;

}

/\*\*

\* @param remainder the remainder to set

\*/

public void setRemainder(int remainder) {

this.remainder = remainder;

}

}

1. **MatrixFrame**

Class Declaration: *public class MatrixFrame extends javax.swing.JFrame*

1. **DrawMatrixFrame**

Class Declaration: *public class DrawMatrixFrame extends javax.swing.JFrame implements ActionListener*

Class variables:

* JButton[][] grid; // used to fill buttons into 2D matrix
* int buttonsOrder; // to count the number of drawn buttons
* GridLayout gridLayOut; // to fill the JPanel with the gridlayout size
* int playerStickCounter;// to count the number of sticks drawn by players every turn.
* int playerStickDirection; // to select the direction of sticks drawn by players every turn(vertical, Horizental).
* int xCoordinate;//to read the x coordinate of the first selection by players every turn.
* int yCoordinate; ;//to read the y coordinate of the first selection by players every turn.
* boolean player1Turn;// to flag the players 1 turn.
* boolean player2Turn;// to flag the players 2 turn.
* oolean firstSelect;// to detect the first selected done by players every turn
* int lastButtonXPosition;// the last x coordinate colored by the player
* int lastButtonYPosition;// the last y coordinate colored by the player
* MatrixSize matrixSize;// object of MatrixSize class
* Players p1, p2;// object of Players class

**Class Hierarchy**

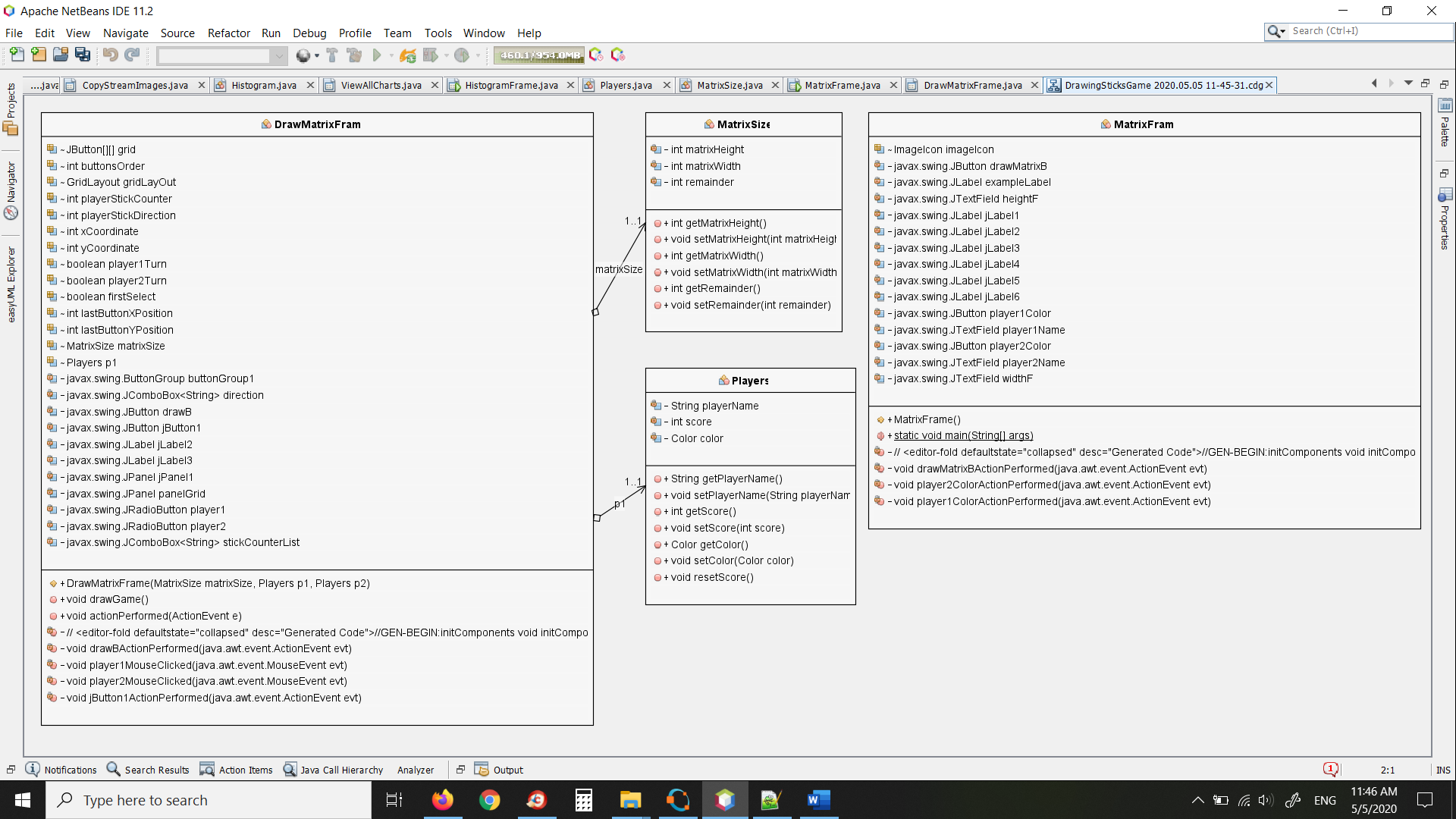
java.lang.[**Object**](https://docs.oracle.com/en/java/javase/13/docs/api/java.base/java/lang/Object.html?is-external=true)

* java.awt.[**Component**](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/Component.html?is-external=true) (implements java.awt.image.[ImageObserver](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/image/ImageObserver.html?is-external=true), java.awt.[MenuContainer](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/MenuContainer.html?is-external=true), java.io.[Serializable](https://docs.oracle.com/en/java/javase/13/docs/api/java.base/java/io/Serializable.html?is-external=true))
  + java.awt.[**Container**](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/Container.html?is-external=true)
    - java.awt.[**Window**](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/Window.html?is-external=true) (implements javax.accessibility.[Accessible](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/javax/accessibility/Accessible.html?is-external=true))
      * java.awt.[**Frame**](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/Frame.html?is-external=true) (implements java.awt.[MenuContainer](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/MenuContainer.html?is-external=true))
        + javax.swing.[**JFrame**](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/javax/swing/JFrame.html?is-external=true) (implements javax.accessibility.[Accessible](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/javax/accessibility/Accessible.html?is-external=true), javax.swing.[RootPaneContainer](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/javax/swing/RootPaneContainer.html?is-external=true), javax.swing.[WindowConstants](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/javax/swing/WindowConstants.html?is-external=true))

drawing.drawingsticksgame.[**DrawMatrixFrame**](file:///F:\FreeLancing\DrawingSticksGame\target\site\apidocs\drawing\drawingsticksgame\DrawMatrixFrame.html) (implements java.awt.event.[ActionListener](https://docs.oracle.com/en/java/javase/13/docs/api/java.desktop/java/awt/event/ActionListener.html?is-external=true))

drawing.drawingsticksgame.[**MatrixFrame**](file:///F:\FreeLancing\DrawingSticksGame\target\site\apidocs\drawing\drawingsticksgame\MatrixFrame.html)

* drawing.drawingsticksgame.[**MatrixSize**](file:///F:\FreeLancing\DrawingSticksGame\target\site\apidocs\drawing\drawingsticksgame\MatrixSize.html)
* drawing.drawingsticksgame.[**Players**](file:///F:\FreeLancing\DrawingSticksGame\target\site\apidocs\drawing\drawingsticksgame\Players.html)

**UML daigram**

**Stick Game GUI**

