```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package mvctictactoe;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseListener;
import java.util.Observable;
import javax.imageio.ImageIO;
import javax.swing.Icon;
import javax.swing.ImageIcon;
import javax.swing.JButton;

/**
 *
 * @author Benjamin Chinwe 2016
 */
public class TicTacToeController extends Observable {

    private final TicTacToeView theView;
    private final TicTacToeModel theModel;

    // TicTacToeController class constructor
    public TicTacToeController(TicTacToeView ticTacToeView) {
        theView = ticTacToeView;
```

```
        theView.setPlayersName(); // Call setPlayersName method of TicTacToeView class
        theView.setPlayerSeed(); // Call setPlayerSeed method of TicTacToeView class

        //Instantiate the TicTacToeModel class
        theModel = new TicTacToeModel(theView.getGameButton(),
                theView.getPlayerOneName(), theView.getPlayerTwoName(),
                theView.getPlayerSeed());

        //Call the addGameButtonListener method of TicTacToeView class and
        //class a new class GameButtonListener()
        theView.addGameButtonListener(new GameButtonListener());

    }

    // Inner class GameButtonListener with ActionListener implimentation
    public class GameButtonListener extends MouseAdapter implements
            ActionListener, MouseListener {

        public GameButtonListener() { //Constructor
        }

        // ActionPerformed performs Buttoon click ActionEvent
        @Override
        public void actionPerformed(ActionEvent evt) {
            if (theModel.isGo()) {
                theModel.reset(); // Reset the game after a win or a draw

                //ReSet Player's ID_Seed after a Win or a Draw
```

```
                theView.setPlayerSeed(theModel.getPlayerSeed());
                //Get Resetted Game Button array after a Win or a Draw
                theView.setGameButton(theModel.getGameButton());
            } //game over, reset all flags

            if (theView.getPlayerSeed().equals("X")) {
                theView.getOutPutText().setText(theView.getPlayerTwoName()
                        + " 'Turn to play - TicTacToe");
            } else {
                theView.getOutPutText().setText(theView.getPlayerOneName()
                        + " 'Turn to play - TicTacToe");
            }

            //Call jButtonActionPerformed() to pass event and button array
            //to other function
            jButtonActionPerformed(evt, theView.getGameButton());
        }

        private void jButtonActionPerformed(ActionEvent evt,
                JButton[][] gameButton) {
            try {
                //if (evt.getActionCommand().equals(TicTacToeModel.Seed.EMPTY)) {
                for (int i = 0; i < 3; i++) {
                    for (int j = 0; j < 3; j++) {
                        // Determine event source
                        if ((JButton) evt.getSource() == gameButton[i][j]) {
                            theView.setPlayerSeed(theModel.getPlayerSeed());
                            if (theView.getPlayerSeed().equals("X")) {
                                Icon img = new ImageIcon(ImageIO.read(
```

```
                                    GameButtonListener.class.
                                            getResourceAsStream("image/cross.jpg")));
                            ((JButton) evt.getSource()).setIcon(img);

                            // Set enum value
                            theModel.setCurrentSeed(theModel.currentSeed.CROSS, i, j);
                            theModel.setPlayerSeed(theView.getPlayerSeed());
                        } else {
                            Icon img = new ImageIcon(ImageIO.read(
                                    GameButtonListener.class.
                                            getResourceAsStream("image/zero.jpg")));
                            ((JButton) evt.getSource()).setIcon((img));

                            // Set enum value NOUGHT
                            theModel.setCurrentSeed(theModel.currentSeed.NOUGHT, i, j);
                            theModel.setPlayerSeed(theView.getPlayerSeed());
                        }
                        gameButton[i][j].setEnabled(false);

                    }

                }
            }

            if (theModel.boardFull()) { // Determine if is a draw
                draw();
            }

            theView.setGameStatusLabel("");
```

```
                theView.getGameStatusLabel();
                theModel.whoWins(); // Determine if there is winnner

                // Swap the ID-letters 'X* for 'O'
                if (theView.getPlayerSeed().equals("X")) {
                    theView.setPlayerSeed("O");
                    theModel.setPlayerSeed(theView.getPlayerSeed());

                    // Swap the ID-letters 'O' for 'X'
                } else if (theView.getPlayerSeed().equals("O")) {
                    theView.setPlayerSeed("X");
                    theModel.setPlayerSeed(theView.getPlayerSeed());
                }

            } catch (Exception e) {
                theView.displayErrorMessage(e.getMessage()); // Handle error
            }
        }

        private void draw() { // Method to notify players of a draw game
            theModel.setGo(true);
            theView.displayErrorMessage("Draw Score X: " + theModel.getXTotal()
                    + "  O: " + theModel.getOTotal());
            theModel.reset();
        }

        @Override
        public void mouseClicked(java.awt.event.MouseEvent e) {
            addButtonMouseListener(e, theView.getGameButton());
```

```
        }

        @Override
        public void mousePressed(java.awt.event.MouseEvent e) {
        }

        @Override
        public void mouseReleased(java.awt.event.MouseEvent e) {
            theView.setGameStatusLabel("");
        }

        @Override
        public void mouseEntered(java.awt.event.MouseEvent e) {
        }

        @Override
        public void mouseExited(java.awt.event.MouseEvent e) {

        }

        private void addButtonMouseListener(java.awt.event.MouseEvent e,
                JButton[][] gameButton) {

            for (int i = 0; i < 3; i++) {
                for (int j = 0; j < 3; j++) {
                    if ((JButton) e.getSource() == gameButton[i][j]) {
                        if (!(gameButton[i][j].isEnabled())) {
                            //setChange and Notify the update method of TicTacToeView
                            setChanged();
```

```
                            notifyObservers("Taken, click elsewhere");
                        }
                    }
                }
                theView.getGameStatusLabel();
            }
        }
    }
}
```