```java
/*--------------------------------------------------------------------------------------------------
 * Author:      Dan Cassidy
 * Date:        2015-07-18
 * Assignment:  HW4-1
 * Source File: GameController.java
 * Language:    Java
 * Course:      CSCI-C 490, Android Programming, MoWe 08:00
--------------------------------------------------------------------------------------------------*/
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

/**
 * Controller class for the Tic Tac Toe game. Bridges the model (Game class) and the view (GameView
 * class) and associates user generated events with model (data) actions.
 *
 * @author Dan Cassidy
 */
public class GameController
{
    private GameModel theGame;  // Model.
    private GameView theView;   // View.

    /**
     * 3-parameter constructor.
     *
     * @param numRows The number of rows the game board will have.
     * @param numColumns The number of columns the game board will have.
     * @param winLength The length of the sequence required to win the game.
     */
    public GameController(int numRows, int numColumns, int winLength)
    {
        // Set up the model and the view.
        theGame = new GameModel(numRows, numColumns, winLength);
        theView = new GameView(theGame.getRows(), theGame.getColumns());
        theView.setWinningConditionsLabelText(theGame.getWinLength() + " in a row wins");
        theView.setStatusLabelText(theGame.getStatusString());

        // Add listeners to the view.
        theView.addResetButtonActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent evt)
            {
                theGame.reset();
                theView.reset();
                theView.setStatusLabelText(theGame.getStatusString());
            }
        });
        theView.addBoardButtonActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent event)
            {
                int row = Integer.parseInt(event.getActionCommand()) / theGame.getColumns();
                int column = Integer.parseInt(event.getActionCommand()) % theGame.getColumns();

                theGame.playMove(row, column);
                theView.setBoardButtonText(row, column, theGame.getSpaceString(row, column));
                theView.setStatusLabelText(theGame.getStatusString());
                if (theGame.getStatus() != GameModel.Status.IN_PROGRESS)
                    theView.setBoardEnabled(false);
            }
        });
    }

    /**
     * Show the view.
     */
    public void showView()
    {
        theView.setVisible(true);
    }
}

```