Tic tac toe:

# Summary of work accomplished:

I've completed the assigned task, and I've attempted some of the extra credit questions.

The implementation uses AJAX. It only uses 2 files: one is an HTML file with some Javascript. The HTML file just sends AJAX requests to the back-end ASP, and receives JSON response data. The response data contains the current state of the grid. Basically, the HTML file is like a client making requests to the back-end logic.

The second file is an ASP file written in VBScript (which I regret, because VBScipt is a PAINFUL language to use). It contains all of the evaluation logic, and it spits out JSON. The HTML file sends requests to the ASP file, and the ASP file responds with JSON.

Pretty simple.

Below is a bullet list summary of which of the extra credit questions I completed, and also explanations for why I did not complete the ones I didn't.

* I've completed Extra credit #1. The code will display a message when someone wins, saying "Game over, X won!" or "Game over, O won!". But there is a bug, where the game doesn't properly detect when it's a "draw", and instead says that "X won!", when in fact it was a draw. I think it also sometimes states the wrong winner as well.
* I did **not** complete or attempt #2.
* I completed extra credit #3. My solution is AJAX based, but it requires the second player to click the "refresh" button to see the updated grid. However, the actual page **does not** need to be refreshed. The "refresh" button just re-draws the grid.
* I would have attempted #4, but I don't think I could have ported my previous AI from my first Tic-tac-toe (the one written in javascript) in just 3 days, so I focused on what I knew I could do. I suppose I could offer my previous implementation as an example of how I would go about implementing an AI, since it really does count as an example of how to do a Tic-Tac-Toe AI.

However, since I didn't complete the napkin-sketch **specifically** for this test, I guess it doesn't count.

* I **will not** try to do #5, because I can't really think of any way to improve the test -- it serves its purpose, as far as I can tell, which is to test developers for back-end knowledge, and potentially also test them for AJAX experience.
* I have uploaded my solution to github on my public github profile.

# Instructions for use:

## Required software:

* Microsoft IIS Server, with support for VBScript.NET ASP (aka, Classic ASP).
* Windows Operating System.
* Web browser.
* Jquery (Bundled with the project).
* Bootstrap (Bundled with the project).

## Installing Microsoft IIS Server.

**You can skip this section if you have Microsoft IIS installed, or you are an adept user of Microsoft IIS. This section just contains instructions on how to get IIS running on a windows machine.**

1. Install Microsoft IIS Server (though I suspect that persons at Medullan already have this working). To install IIS:
   1. Go into the Control Panel of a windows computer.
   2. Before installing, please make sure that no other HTTP servers are running on your computer.
      1. If you have any already-running servers (apache, node's http-server, etc), please either temporarily shut these down, or else configure them to listen on a port other than TCP port 80.
      2. You can also choose to just configure IIS to listen on a port other than 80. If you do so, I am assuming you already know how to do this, and **will not** be providing instructions for how to do so.
   3. Go to "Install or remove Programs"
   4. Select "Add or remove Windows Components".
   5. In the small window that pops up, you'll see a long list of checkboxes for different software and Windows services.
      1. Scroll down within the list until you see "Internet Information Services".
      2. Check this option, and expand it.
      3. When you expand it, you'll see many items beneath it, and each of these items will have child-items as well.
      4. Go through all of the child-items of all the items you see, and make sure that every single item is checked.
   6. Now that you have fully-selected all the components of IIS, you can click "Apply" or "Ok" to apply the changes and install IIS.
2. You now have Microsoft's IIS Web Server installed on your windows machine, and it should be running and listening on TCP port 80. If you chose to use a different configuration, I will assume that you have correctly configured that set of options.

## Running the Tic-Tac-Toe implementation:

* Just extract the ZIP into the IIS document root. This is **usually** found in the following location:
  + C:\inetpub\wwwroot
* If your IIS is configured differently, I assume you know how to find out where your document root is, or otherwise how to change it to something else you desire.
* There is nothing else to be done -- the solution is pretty simple, so now you need only open up your web browser and open up:
  + http://localhost/index.asp
* Enjoy playing Tic-tac-toe.
* Follow the error and warning messages that the program spits out at you. It will give you guidance as needed. Be sure to enter your **player name** and the **name of the game**.