159.352 Computer Work I:
Web based Cross and Naught (Tic-Tac-Toe)

Weight: 25%

This is a multi-part assessment with different intermediate deadlines.

Deadline: April 22nd 2018 (for part B & C)

PART A - Weekly tutorial exercises (5%)

Tutorial exercises reinforce the concepts introduced in lectures, and provide a foundation for this assessment. Therefore it is important to make steady progress and submit the tutorials on time.

Tutorial exercise	Due
Week 2	March 11
Week 3	March 18
Week 4	March 25
Week 5	April 1
Week 6	April 8

PART B – Web-based cross and naught (15%)

In this assessment you will implement the game of cross and naught to be played between a web client and a server using TCP/IP sockets. The client can be any web browser, however you will have to write code to implement the server in any programming language, which supports raw sockets. You may use the server code in the tutorial exercises as a starting point.

The outline of the play is as follows. First the server starts and listens on some chosen port number. Then, client connects to the server at this port and requests for the game page (eg. game.html). The game page can be a simple html document consisting of the cross and naught board layout, a textbox for user input, and submit & reset buttons. You are free to design more interactive version of this page.

___1___|___2___|___3___

___4___ |___5___|___6___

7 | 8 | 9

The user claims a board position e.g. by typing a number between 1 and 9 on the keyboard. This move is sent to the server (using HTTP POST method), which updates its own representation of the board by marking this position. The server, as the second player in this game then uses random strategy to mark its own position. The revised board layout as part of the modified game page is then sent to the client.

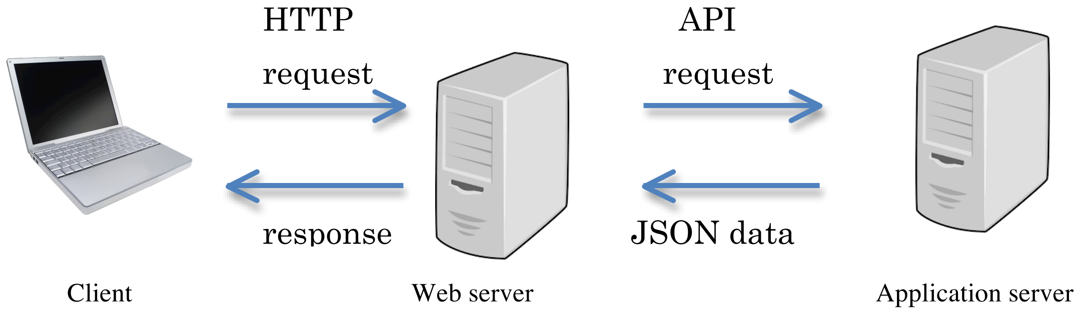
In this manner the game continues and each side takes turn until either:

1. One of the players (either the user or the server) has claimed three board positions in a row, horizontally or vertically, or diagonally, in which case this player wins, or
2. The board is filled in and neither player has won, in which case the game is a draw.

The result of the game is displayed to the user when the game ends. The reset button simply allows the user to reinitialize the board to its initial state, and play the game again. Note that the user may choose to play on a position that has already been claimed. In this case, the player's move is ignored.

Let’s think more about the architectural design. If you get your server to deal with the HTTP client request and handle the game logic, it is performing the role of an application server. Suppose if you want to implement more games like chess, you will end up overloading your web server with application logic.

A three-tiered architecture is a more scalable approach. Here the web server is oblivious to the game play and only focuses on serving standard HTTP requests. The backend application server implements the game logic and maintains state information (e.g. board position). In this way you can scale your web services by having a separate application server for each game. For the purpose of this assessment, the web server and application server can be running on the same host, and you can limit the game play to one client at a time.



Submission: A well-documented source code, complete with usage and test report via Stream site

PART C – Web security (5%)

In this section you are to examine your web system developed in Part B, and identify a security vulnerability affecting it. Describe in sufficient detail how this vulnerability can be exploited. Finally make recommendations for policies or control mechanisms, which addresses the vulnerabilities and counters the threat. Should it happen that after exhausting all options you are unable to find any vulnerability in your system, then you need to describe the mechanisms, which make the system secure.

Summarize your findings in a short report (around one page length) and include any code and references you have used to demonstrate/support your discussion. Marks will be awarded for the depth of ideas presented so it is imperative that you do a through research.