Assessment Specification

Course / Programme:	**Computing / Computing & Website Development (Level HE5)**
Module Name:	**Web Programming**
Module Code:	**CPU5001**
Module Leader:	**Martin Stanhope**
E-mail address:	**m.stanhope@bolton.ac.uk**
Assignment Number & Title:	**001 – Bingo Game (Client-side scripting)**
Issue Date:	**Monday 10th October 2016**
Submission Deadline:	1. In-class demonstrations: In each weekly class session up to and including the week beginning **Monday 14th Nov. 2016.** 2. Submission of the required zip file into Turnitin (for viewing by the External Examiner) by: **23:00, Friday 18th Nov. 2016**

Aim

To create a browser-based bingo game using HTML5 canvas and JavaScript.

Background

Bingo is a game in which 90 uniquely numbered balls are spun in an enclosure to allow one number at a time to randomly fall. That number is then called out by the bingo caller to allow the players to mark the number on their bingo card. Prizes are normally awarded for the first bingo player to get a full horizontal line of numbers (a full-line) and all of the numbers on their card (a full-house).

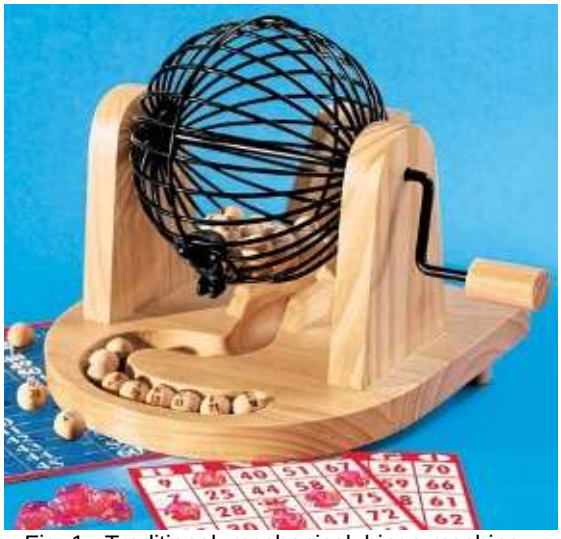

Fig. 1 Traditional, mechanical, bingo machine.

Fig. 2 Traditional, paper, bingo cards.

Bingo is very popular as an entertaining past-time in care homes for the elderly in which the traditional, mechanical machine and paper bingo cards are used.

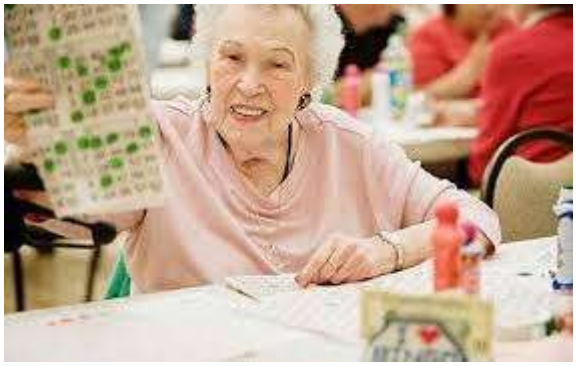

Fig. 3 Bingo in care homes for the elderly.

A local care home for the elderly has made a request for the development of a browser-based, bingo number generating game that the home can use instead of the mechanical machine. A laptop will be connected to a large, wall-mounted display and be operated by a care assistant. The bingo players will continue to use paper bingo cards and felt-tip marker pens.

The residents have asked if the bingo caller, in addition to reading out the displayed number, would also call out the associated British bingo nickname for each number, as outlined in the table below:

Number	Nickname
1	Kelly's Eye
2	One little duck
3	Cup of tea
4	Knock at the door
5	Man alive
6	Half a dozen
7	Lucky

Table 1 - Bingo nicknames. (Full list available: https://en.wikipedia.org/wiki/List_of_British_bingo_nicknames

Instructions

1. Using HTML5 Canvas, CSS and JavaScript, create a browser-based bingo game which allows the bingo caller to control the following features:

a. Display a large graphic (almost full screen) of a ball showing a unique, random number in the range 1 – 90 together with the nickname of the number.

b. Display a matrix of balls (1 – 90) to indicate which numbers have been called during a game. For example, the display could show 90 balls number 1 – 90 all displayed in red and when a random number is generated the ball that has that number should turn to a different colour, such as green.

c. Display the number of numbers that have been called at any time.

d. Display a list of the numbers that have been called in the order they have been generated. This is often useful for checking if a player has correctly called a full-line or full-house on their bingo card.

e. Display the current day, date, year and time.

2. You are allowed to include any other features that you consider to be useful and would improve the operation of the game.

Submission Requirements

1. Using the Turnitin submission link provided at the top of the CPU5001 Moodle page, submit a zip file containing all of your bingo related HTML, CSS and JavaScript files in additional to the weekly skills development exercises and your work log spreadsheet. Name the zip file using the following format:

SURNAME-Firstname-StudentIDNumber-username-BINGO.zip

for example:

SMITH-John-1412345-js99crt-BINGO.zip

The home page of your site should be called: **index.html**

2. Give the tutor weekly demonstrations (up to and including the final submission week) of your development work during your timetabled class. You will be asked to explain the operation of any line of HTML, CSS and JavaScript code so make sure you have written all of the code yourself. The mark you receive will be based on the work you can demonstrate and explain the operation of.

Marking guidelines

(Please remember that these are only guidelines and the tutor's awarded mark is final).

Grade D (40-49%)

a. The completion of the weekly skills development exercises issued in weeks 1 and 2 based on the use of the HTML6 canvas element and modular programming .

b. A basic bingo game which allows the bingo caller to carry out the following:

i. Clear the canvas area
ii. Draw a large, easy to see, graphic of a uniquely numbered bingo ball.
iii. Display the bingo number's nickname.

Grade C (50-59%)

a. The completion of all of the weekly skills development exercises issued in week 3 based on the use button control using the onclick event and the JavaScript Date object.

b. As for grade D, plus the following features:

i. Display the number of numbers that have been called in the current game.
ii. Display a list of the numbers that have been called in the current game.
iii. Display the day, date, year and the current time in hour and minutes that is updated continuously throughout the current game.

Grade B (60-69%)

As for grade C, plus the following features:

i. Display a matrix of all balls (1 to 90) and indicate, by the use of colours, which numbers have and haven't been generated at any stage within the current game.
ii. Well structured, modular code, using local variables and parameter passing.
iii. Extensive use of programme comments.

Grade A (70-100%)

a. The completion of all of the weekly skills development exercises issued in week 4 based on the use of JavaScript timers.

b. As for grade B, plus the following features:

i. The use of advanced JavaScript features, such as timers, to make the game more visually appealing by the use of simple animation.

ii. Any game feature of your own design that the nursing home has not specified and that you consider to be an improvement of the game specification.

Learning Outcomes (LO's)

The following table shows the 4 Learning Outcomes for this module and indicates those assessed by this assignment.

Learning Outcome	Description	Assessed by:
1	Critically understand the concepts and principles of client-side scripting used to create dynamic web sites.	Assignment 1
2	Evaluate the appropriateness of open-source software libraries to provide enhanced, client-side functionality.	Assignment 1
3	Critically understand the limits of client-side scripting and the need for server-side scripting to allow web sites to interact with remote data resources.	Assignment 2
4	Understand and use a range of programming techniques to provide operational resilience of web sites.	Assignment 2