ASSIGNMENT DETAILS

This is an individual project; the weighting of the project towards the overall module mark will vary depending on your current academic standing. Please contact the module lecturer if you have any questions around this.

Students are reminded that there are strict rules around plagiarism and collusion. University guidance on plagiarism can be found here. Students are also asked to be mindful of copyright and intellectual property issues.

This assignment should be submitted to Canvas by 5pm on Friday 9th August 2024 as per the instructions provided.

SCENARIO

You are currently applying for the position of a client-side web developer. For the interview you have been asked to develop a word-based game inspired* by the TV show 'Countdown', which you will be asked to present. Specifically, you have been asked to focus your development on the implementation of a single player word round from the game show.

*** The emphasis of this assignment is to create a word-based game using Countdown as inspiration only. So long as you create a word-based game that meets the features and requirements outlined below, you will have met the brief.**

WHAT IS THE COUNTDOWN TV SHOW?

A long standing popular TV game show consisting of various number and word challenges, further information on this show may be found at the following Wikipedia page, https://en.wikipedia.org/wiki/Countdown_(game_show). Episodes of the TV show may also be freely viewed online, for example, via YouTube.

You have been asked to focus on the development of a single word-based round inspired by this TV show. These rounds challenge the player to find the longest valid word (as defined by a dictionary, e.g. Oxford English Dictionary) from a set of nine randomly drawn letters. Generally, in the Countdown TV Show, the process is as follows for a word round:

- At the start of the round, the nine letters to be used are drawn. To do this the player will state whether they would like a vowel (A, E, I, O, U) or consonant (any other letter), which is randomly drawn from the appropriate set of letters. The player will see the drawn letter and repeat the process until all nine letters are drawn.
- Once the letters have been selected, the player normally has 30 seconds to identify the longest valid word they can from the letters previously drawn.
- At the end of the round, the longest word is presented and, if valid, points are awarded, usually 1 per letter of the word.

Remember, the brief is to be inspired by the Countdown game, you do not need to adhere rigidly to this format.

REQUIREMENTS

Broadly, it is anticipated that most implementations will consist of:

- A game landing where the concept and key instructions are provided to the player.
- Selection of letters to be used in the round.
- A time limited period where players record the words they have identified.
- The results of the game where the players performance (score) is calculated and presented to them.

It is expected that you will use a broad range of HTML, CSS and JavaScript skills in the development of this game. As this assignment aims to assess your ability with these technologies, the use of templates or frameworks is not expected, and their use may reduce, rather than enhance your coursework mark.

KEY FEATURES

While a great variety in the style of game produced is expected, all should incorporate and clearly evidence the following features as part of the final submission.

- Players shall be able to select a set number of random consonant or vowel letters. Furthermore, they shall not be able to select more than the defined number of letters and shall not be able to enter any words until the last letter is drawn.

- Once all letters have been selected, players shall have a limited amount of time to identify as many valid words as they can from the set of letters drawn. Once the time available has expired, the game shall automatically end, and no further words may be entered into the game.
- Please note there is no requirement to check the words entered against an official dictionary (e.g. Oxford). it is acceptable to consider a word valid so long as it is composed of letters that have been drawn in the current round of the game.
- Once the game timer has expired, a score should be automatically calculated and displayed based on the words entered by the player. The list of words entered by the player during the round shall also be displayed.
- The specific scoring mechanism to be used is up to you, however, here are a few tips to help you get started:
    - Valid words should be composed of the letters drawn at the start of the round.
    - Each letter drawn may only be used once per word. For example, if the letter 'A' is drawn once at the start of the round, then each word submitted may only use the letter 'A' once.
    - The total score may be calculated in different ways. This could be based on the longest valid word, the total number of valid words identified, the difficulty of the word based on the letters used or a combination of these and other scoring mechanisms.
    - As mentioned, it is up to you to determine the most appropriate scoring method, however, it must be logical and provide a fair score based on the player's performance.
- Once the game timer has expired, a score should be automatically calculated and displayed based on the words entered by the player. Additionally, the list of words entered by the player during the round shall be displayed.
- The integrity and reliability of the game should be considered as part of the development. Common error conditions such as trying to submit an empty field should be handled. Likewise, trying to submit a word consisting of invalid characters or one that cannot be formed from the given letters should be identified to the user and appropriate action taken.
- A range of media such as images, video, audio effects/music, animations and other techniques should be incorporated to enhance the player's experience.
- Remember, be mindful of copyright and intellectual property. Ensure you use your own or 'free to use' content. If 'free to use' content has been incorporated within your project, ensure it is suitably referenced in the code and documentation.
- The game should embed a range of suitable accessibility features to ensure that the game may be enjoyed by those with additional needs.

Please note, the submitted game should be inspired by the Countdown TV Shows word rounds, however, not be a direct clone of it. Within the scope of the above requirements, you are encouraged to apply your own creativity to meet the brief.

ASSESSMENT CRITERIA

For this assignment, a score (percentage) out of 100, rounded to the nearest integer will be returned based on the following marking criteria. The weighting of each criterion within the assignment has also been listed.

Meeting the Assessment Brief: Production of a '*Word-based quiz inspired by the TV show Countdown*' (15% Weighting)

1st (>= 70%)	2.1 (60% - 69%)	2.2 (50% - 59%)	3rd (40% - 49%)	Marginal Fail (35% - 39%)	Fail (<35%)
An excellent, refined, novel, engaging game, true to the brief has been produced. Good practice from existing similar games has been evaluated and adopted.	A very good, refined, novel, engaging game, true to the brief has been produced. Some consideration of existing practice and adoption into game is evidenced.	A game reminiscent of a word-based quiz has been produced. Weak evidence of considering existing approaches and incorporating good practice into the game.	A game reminiscent of a word-based quiz has been produced. Game experience may lack some cohesion, be lacking in content or omit other elements.	A game reminiscent of a word-based quiz has been produced. Game experience significantly lacks cohesion, has insufficient content or omits other key parts.	Game not reminiscent of a word-based quiz. Severely lacks in cohesion, content or other key elements.

Quality of User Interface and User Experience (20% Weighting)

1st (>= 70%)	2.1 (60% - 69%)	2.2 (50% - 59%)	3rd (40% - 49%)	Marginal Fail (35% - 39%)	Fail (<35%)
An intuitive, interactive game, that creates an immersive environment through a range of well curated and diverse techniques and media.	An intuitive, interactive game, that creates a suitable environment through a range of techniques and media.	An intuitive, interactive game presented in a suitable environment. Key UX elements have been considered, albeit with weaknesses in their implementation.	A somewhat intuitive and interactive game. Significant weaknesses in UX elements, but none that prevent player engagement with the game.	Gameplay is not intuitive, and interactivity is limited. Game is playable, but with significant barriers to user engagement and completion.	Gameplay is not intuitive, and interactivity is limited. Game is broadly not playable.

Implementation of Key Technical Features (30% Weighting, up to 5% per feature)

	1st (>= 70%)	2.1 (60% - 69%)	2.2 (50% - 59%)	3rd (40% - 49%)	Marginal Fail (35% - 39%)	Fail (<35%)
Selection of Letters	Complete, suitable implementation and free of all but very trivial defects.	Suitable implementation and mostly free of defects.	Clear attempt at implementation, has some functionality, may have noticeable minor defects	Clear attempt at implementation, has some functionality, may have significant major defects.	Some attempt at implementation, but broadly non-operational.	Largely missing or non-operational.
JavaScript Timed Events						
Appropriate Use of Media						
Suitable Game Scoring						
Game Accessibility						
Integrity & Reliability						

Technical Complexity of Implementation (20% Weighting)

1st (>= 70%)	2.1 (60% - 69%)	2.2 (50% - 59%)	3rd (40% - 49%)	Marginal Fail (35% - 39%)	Fail (<35%)
Clear evidence that learning beyond core module material has taken place through implementation of suitable advanced technical features; normally using HTML, CSS and/or JS; these should be broadly defect free.	Broadly defect free use of HTML, CSS and JS to implement game. This implementation should utilise a broad range of the technical skills developed within the scope of the modules content.	Use of HTML, CSS and JS to implement game. This implementation should utilise a broad range of the technical skills developed within the scope of the modules content. There may be some minor code defects.	Use of HTML, CSS and JS to implement game. This implementation should utilise a broad range of the technical skills developed within the scope of the modules content. There may be some major code defects.	Use of HTML, CSS and JS to implement game. This implementation utilises a limited range of the technical skills developed within the scope of the modules content. There may be major code defects significantly limiting game playability.	Use of HTML, CSS and JS to implement game. This implementation utilises a very limited range of the technical skills developed within the scope of the modules content. There may be major code defects rendering the game unplayable.

Presentation of Completed Game (including documentation) (15% Weighting)

1st (>= 70%)	2.1 (60% - 69%)	2.2 (50% - 59%)	3rd (40% - 49%)	Marginal Fail (35% - 39%)	Fail (<35%)
An excellent, cohesive presentation clearly communicating core and additional game features. Excellent PPT supporting presentation delivery. All team member contributions are clearly explained.	A very good, cohesive presentation clearly communicating core and additional game features. Very good PPT supporting presentation delivery. All team member contributions are clearly explained.	A good presentation communicating core and additional game features. Good PPT supporting presentation delivery. All team member contributions are clearly explained.	Acceptable presentation communicating core and additional game features. Reasonable PPT supporting presentation delivery. Team member contributions explained.	Limited presentation. Core and additional game features poorly communicated. Poor PPT supporting presentation delivery. Team member contributions not fully explained.	Poor presentation. Core and additional game features not sufficiently discussed. Poor/No PPT aiding presentation delivery. Team member contributions not fully explained.

COURSEWORK SUPPORT

Students will have access to the following support during the project:

- Online resources including this assignment spec and additional guidance material (including video) available via Canvas.
- General student support via email, Canvas and Microsoft Teams. Students are encouraged to contact the lecturer by email in the first instance.

SUMMATIVE FEEDBACK

After assignment work has been submitted, summative feedback is available via:

- An individual mark broken down by the criteria provided above.
- The opportunity to receive further individual feedback by appointment with the lecturer.

Please note that feedback will be returned as soon as possible, subject to marker availability and other constraints.

PROJECT DELIVERABLES

1. Website, to include all files and folders needed to view your game (HTML, CSS, JS, Images, Sound, Video, etc.)
2. PowerPoint slides summarising your game, guidance follows below.
3. Recorded video demonstration of game in operation, guidance follows below.

In deliverables 2 and 3, it is important that you clearly highlight where you have implemented HTML, CSS and JavaScript.

POWERPOINT SLIDES GUIDANCE

The aim of the PowerPoint is to provide a clear reference guide for the key deliverables of your project and should have the following structure.

- Cover slide showing project title, your name and student number.
- A slide with an annotated screenshot showing the welcome/landing screen for your game.
- A slide with an annotated screenshot showing how the letters are selected for the game, this may include annotated screenshots of both the rendered page in the browser and corresponding code.
- A slide illustrating the operation of the JavaScript timed event, this may include annotated screenshots of both the rendered page in the browser and corresponding code.
- A slide illustrating how the player submits words in the game, this may include annotated screenshots of both the rendered page in the browser and corresponding code.
- A slide showing a screenshot of the scoring for the player. Annotations briefly explaining how the score was derived should be included.
- A slide showing how you have protected against user input error conditions (e.g. empty fields) and ensured accurate scoring of the players submitted words (e.g. only valid words get scored). This may include annotated screenshots of both the rendered page in the browser and corresponding code.
- A slide with annotated screenshot(s) showing the operation of accessibility features that have been incorporated in the game and the issues they are designed to mitigate.
- A slide briefly outlining, as you deem appropriate, the top two or three features of the game in terms of technical complexity and/or quality of user experience. Annotations and code snippets should be included.
- A slide summarising where game images have been sourced and confirming their use is in line with copyright/IP rules.

For the above, a maximum of TWO slides per point is permissible. For clarity, your complete PowerPoint should be no longer than 20 slides.

VIDEO DEMONSTRATION GUIDANCE

The video demonstration should cover the points highlighted in the PowerPoint submission. Specifically, this should be a practical demonstration of the game in action, the slides are not expected to be used. Additionally, the focus should be on the operation of the game in the browser, any inclusion of the underlying code is expected to be by exception and minimal.

The video shall last no more than 10 minutes (though this is a limit not a target, for most videos 6-8 minutes will be fine - the absolute max is 11 minutes), should consist of screen recording with a corresponding student narrative, and will normally have the following structure:

1. Video opening should briefly introduce and contextualise the game (max 1 minute).
2. At least one complete playthrough of the game, however, more may be needed to ensure that you can demonstrate all of the key features. Ensure you take enough time so that the environment created through the game can be appreciated and graded accordingly. Rushing over parts may impact your grade.
3. The video should conclude with a summary of 'best bits' in terms of technical complexity and quality of experience.

Important Notes:

- Within Points 2-3, all of the areas described in the PowerPoint presentation must be covered.
- It is your responsibility to ensure the quality of the video is sufficient to clearly see and hear all elements of the demonstration. A poor-quality video will likely have the corresponding effect on final marks!

RECORDING THE VIDEO

Firstly, before recording the video, you are strongly advised to plan what you will show and talk about. For example, by making a series of bullet points with approximate timings beside each. Use this outline to make sure you cover all the required areas and show off the best of your project within the 10-minute limit.

The video may be recorded using any suitable software, however, OBS Studio (https://obsproject.com/) is freeware that works well for screen recordings etc. and has been used successfully for similar work in other modules. Note, on Macs, high resolution screens have been known to cause some problems with OBS, in this case, using QuickTime Player's screen recording function works well.

Again, please bear in mind that the quality of your video recordings is important to ensure that your work can be accurately marked. To this end, it is recommended that you use a minimum recording resolution of 720p (ideally 1080p) and a video format of mp4. Note that if you are using QuickTime, this records in .mov by default, but you can export to mp4.

SUBMISSION INSTRUCTIONS

Please follow these instructions carefully for submission of the CSC1030 coursework

1. Firstly, review all assignment guidance and make sure that you have completed all the required work.

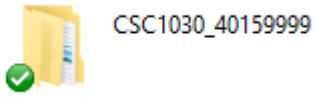
2. Compile your work into the following folder/file structure.

- CSC1030_XXXXXXXX (Folder)
  - Code (Folder)
    - *All website files go here – HTML, CSS, JS, Images, Video, Sound, etc.*
  - Documentation (Folder)
    - PowerPoint Presentation (File)
    - Video Demonstration (File, mp4 recommended)

Note: XXXXXXXX should be your student number.

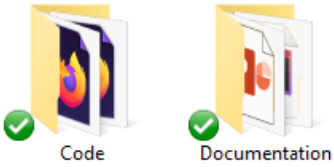
Screenshots of Example Submission for student 40159999:

Ensure you follow the same naming conventions!



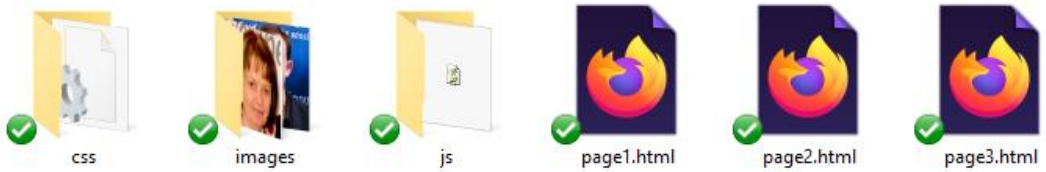
All material should be submitted in a parent folder as shown above.

Contents of 'CSC1030_40159999' Folder



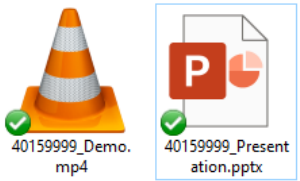
The submission will have two sub folders, one for the website code and other necessary resources, such as images. The second contains the PowerPoint and demonstration video.

Contents of 'Code' Folder



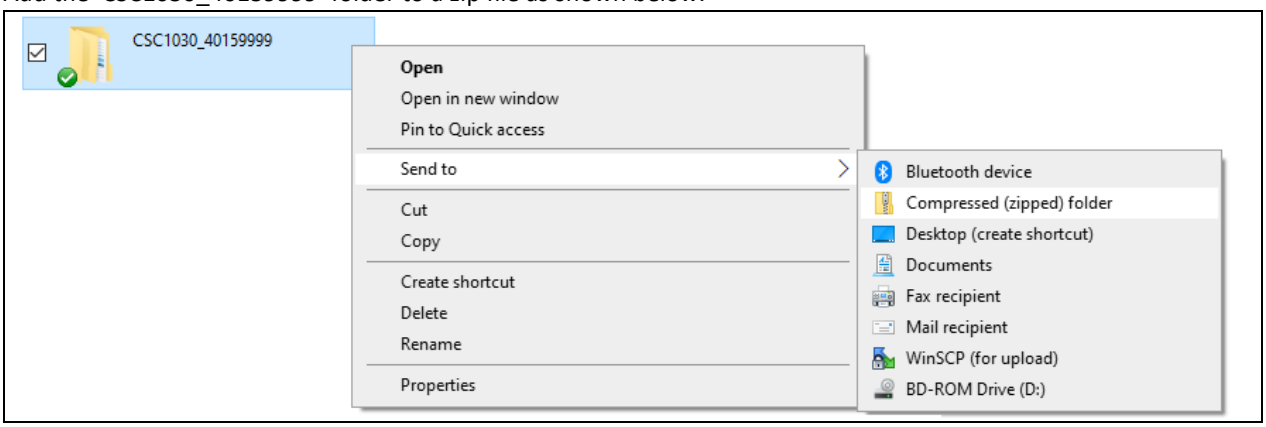
This is an example of what may be contained within the 'Code' folder. Every student's website structure will likely be different. It is essential that you check the website works as expected when everything is copied into this location.

Contents of the 'Documentation' Folder



The presentation and video demonstration will be placed here. Please ensure that you follow the naming conventions provided. '40159999' should be replaced with your student number.

3. Add the ‘CSC1030_40159999’ folder to a zip file as shown below.



4. Upload the ‘CSC1030_40159999.zip’ file to the ‘CSC1030 Summer Coursework’ assignment area on Canvas. Note:
    - Please do not leave uploading the assignment to the last minute, be mindful that submissions are likely to have a substantial file size. The submission time is based on the completion of a submission, not the start.

It is your responsibility to ensure the coursework is uploaded in time, presented in line with the instructions above and is not corrupted in any way. You should leave enough time to upload, then verify your submission for your own piece of mind and to avoid any loss of marks.