Assignment Specification School of Computer and Engineering Sciences			
Module Code CO7006	**Module Title** Web Systems	**Assessment No** 1 of 1	**Weighting** 100%
Title Practical Project & Critique		**In-Year Reassessment Offered** No	**Generative AI** Not Allowed
Summary To produce a website that allows users to search for universities and their web address, with the added benefit of having unique lists for themselves. A report critically evaluating web frameworks.		**Submission Date** 09/07/2025 at 13:00 7-day Submission Window Allowed	**Feedback Due** 21/08/2025

Instructions

Website Brief

Develop a website, consisting of up to 2 pages (For clarity a page is any file that outputs html).
The website should have the following features:

- The website should allow the user to search for a list of universities
- A user should be able to select a university and see the website link for that university.
- Authenticated users should be able to have a unique list of universities.
- Only valid universities should be presented in the list.
- The list should persist between user sessions, through database storage.
- A unique design

Your website must be created using HTML, CSS, JavaScript, and PHP that is written by yourself. The use of third-party libraries is expressly forbidden. Frameworks and automated authoring tools (Bootstrap, Angular, etc) is also expressly forbidden. Your website should be original and should not be built following a tutorial or template. Use the techniques you have learnt on this module and from your self-study.

A database for you to use in your assignment has been provided to you, in the file uniList.sql on Moodle. You should create a database from this backup file and create a new user for the database through cPanel. The website you develop should manipulate this database.

You can retrieve information about universities using the following api:
https://github.com/Hipo/university-domains-list-api
If you choose to implement security passwords as part of the authentication process you must use the CryptoAPI to ensure a secure process. You should assume that any passwords used are compromised.
You should not place the search function behind any authentication barrier.
For clarity secure persistence of user logins is not required. Requiring the user to provide login details for each authenticated action is sufficient.

For the avoidance of doubt, the inclusion of a data driven aspect requires you to write original code. Implementations which can be achieved by copying and pasting code ‘as-is’ such as the

integration of a YouTube video or mapping provider would not meet the requirements for this aspect.

Third party images, videos and copy may be used to provide content for the site, but these must all be attributed on the page where they are used (see https://codepen.io/andymuncey/pen/eozpZR for examples of how to do this). The reference for every item of third-party content must be provided in the footer of the page on which it is used, in APA format.

Critical Reflection Brief

You must also provide a 750-word written submission which critically evaluates two different technology stacks (such as LAMP, MEAN, the ASP.NET framework or a serverless stack) for implementing data driven websites. You should provide arguments to support a statement that you have regarding the framework. Your arguments should be backed by strong academic evidence. Citations must be in APA format and contain a reference list for them.

Assessment Criteria Weightings

- Website 85%
- Critical Reflection 15%

Mark Requirements

Website

Minimal Requirements

- All HTML pages are valid to W3C standards.
- All CSS is valid to W3C standards and provided in a separate stylesheet.
- All JS is provided in a separate file.
- Reasonable conformance to WCAG 2.1
- Users are to search for universities and a weblink to the university
- A list of universities is shown from the database to the user which they then can search for a weblink without typing.
- Appropriate comments included in the JavaScript files.

Additional Requirements

- Styling has been used to make the website look more appealing.
- Users can add or delete from their personal list of universities, and this is reflected in the database.
- Only valid universities are entered into the database.
- Authentication is secure
- The website handles incorrect data.

Critical Reflection

Minimal Requirements

- The critique includes a satisfactory evaluation of two technology stacks
- Arguments are supported with evidence.
- The critique is formatted sensibly, including headers and paragraphs.
- Citations reasonably conform to APA style.

Additional Requirements

- The critique demonstrates a high degree of analysis and evaluation and may provide new insights into knowledge in the domain of two technology stacks

- The critique demonstrates a convincing and sustained command of accepted critical positions regarding data driven web application technologies.
- Arguments are supported with strong evidence.
- Citations confirm to APA style.

Pre-submission checklist

This checklist is provided to help you avoid some of the most common errors found in previous submissions, it is not comprehensive, and you must still ensure you meet the criteria set out above.

Website

- ☐ All pages checked for valid HTML using the validator at https://validator.w3.org/
- ☐ All pages checked for valid CSS using validator at https://jigsaw.w3.org/css-validator/ ☐ All pages checked for accessibility issues using the WAVE tool at https://wave.webaim.org/ ☐ The layout is consistent e.g., menus appear in the same place (and work) on each page.
- ☐ All pages checked manually for accessibility issues, in particular ensuring alt text for images that accurately describes each image. Written component
- ☐ Both your critique, and your discussion on credential security, are supported by citations from a range of sources.
- ☐ Your critique is structured with headings and paragraphs.

Additional Information

Learning Outcomes Assessed

- Design and create structurally sound web sites with suitable stylistic features using a wide range of techniques appropriate for a given set of requirements.
- Implement scripting to enhance the functionality of a website.
- Utilise server-side applications in the process of the development of a data-driven web application.
- Critically evaluate different methodologies for implementing data driven websites

Assessment Support

Students can request help through the workshops, and can email the lecturer directly to arrange one-to-one support.

Submission Window, Exceptional Circumstances, and Assessment Regulations

You are expected to submit work by the submission date specified at the start of the assignment specification. Some assignments may support a 7-day window in which students can submit work late without penalty and this will be specified below the submission date at the start of this brief. Any work submitted outside of the submission date (or submission window where allowed) will be given a mark of zero.

The assessment for CO7006 Web Systems consists of a single assignment.
The assignment itself consists of two aspects, including the weighting in brackets and the learning outcomes as a sub list:

1. Website (65%)
    1. LO1: Design and create structurally sound web sites with suitable stylistic features using a wide range of techniques appropriate for a given set of requirements.
    2. LO2: Implement scripting to enhance the functionality of a website.
    3. LO3: Utilise server-side applications in the process of the development of a data-driven web application.
2. Critical Reflection (35%)
    1. LO4: Critically evaluate different methodologies for implementing data driven websites

Both parts of the assessment are due in together, see the Assignment Brief for information on deadlines.

Academic Conduct

The material you submit must be your own work. You must not collude with your peers on your work unless the brief explicitly allows this (such as in the case of group work). The penalties for breaching the academic conduct policy are severe. The minimum penalty is usually zero for that piece of work. Further information is available below: • Academic Conduct

- Excess Word Count Penalties
- Cite Them Right Online guidance

Generative AI

The use of generative AI tools where not permitted will be treated as a breach of the academic c**onduct** policy.

This assignment **does not** permit the use of any generative AI tools, including but not limited to ChatGPT, Gemini, Copilot, Midjourney, and others.

Referencing code

Code adapted from third parties must be clearly referenced using comments to denote the start and end of the adapted code. You must also include an APA format reference in the PDF file.

Example of referenced code

```
//code adapted from Thomson, 2012
if (someCharacter == 'z' || someCharacter == 'Z') {
    someCharacter -= 25;
}
//end of adapted code
```

Example of reference entry in PDF file

Thomson, C. (2012). *Rot-13 function in Java?.* Stackoverflow. Retrieved October 25, 2021, from http://stackoverflow.com/questions/8981296/rot-13-function-in-java

Submission Information

The critical evaluation has a word limit of 750 words

For projects that include programming code:

The TurnItIn submission box will have multiple parts. You must submit to the appropriate part

- A PDF file containing in order ○ A link to your website
  - A username and password (If applicable)
  - All code in a monospace font, with headings for each file name. This is on a white background with black text
  - Your written component and reference list in Arial Font
- A ZIP file containing the project

Both files must be named with your assessment (J number), e.g. J123456.pdf and J123456.zip. The name for each entry on TurnItIn must also be your assessment number.

Files submitted in an incorrect format will usually be marked as zero.

All components must be submitted to avoid receiving a mark of zero.

Any late work penalties for assignments will be calculated using the latest submission date/time.

Assessment Criteria Postgraduate

Assignment Task (LOs Covered)	**Fail** (<50%)	**Pass** (50-59%)	**Merit** (60-69%)	**Distinction** (>=70%)
Website • Design and create structurally sound web sites with suitable stylistic features using a wide range of techniques appropriate for a given set of requirements. • Implement scripting to enhance the functionality of a website. • Utilise server-side applications in the process of the development of a data-driven web application.	A website that does not retrieve a list of universities from the database. The website may not be able to search for a university and find a web link dynamically. The website does not follow w3c standard and does not meet web accessibility standards. Code may not follow standard practice.	A website that retrieves a list of universities from the database, which then can search dynamically for a website link. The user can also search for any university and find the corresponding weblink. The website meets w3c standards and most web accessibility standards. All code is developed following standard practice.	An insecure website that allows users to have a list of universities which they can then search for a website link dynamically. The website meets w3c standards. The website meets most web accessibility standards. All code is developed following standard practice	A secure website that allows users to have a unique list of universities which they then can search for the website link dynamically. The website conforms to w3c standards and meets all web accessibility standards. All code is developed following standard practice.

Critical Evaluation • Critically evaluate different methodologies for	An unsatisfactory evaluation of two technology stacks. Arguments are	The critique includes a satisfactory evaluation of two technology stacks Arguments are supported with evidence.	The critique demonstrates a convincing and sustained command of accepted critical positions	The critique demonstrates a high degree of analysis and evaluation and may provide new insights into
implementing data driven websites	unsupported or weak. Poor formatting in the report. Citations not recognisable as APA style	The critique is formatted sensibly, including headers and paragraphs. Citations reasonably conform to APA style.	regarding data driven web application technologies. Arguments are supported with good evidence. Citations conform to APA style.	knowledge in the domain of two technology stacks Arguments are supported with strong evidence. Citations conform to APA style.