Coursework Assignment Brief
Assessment - Undergraduate

Academic Year 2024-25

Module Title:	Database and Web Application Development
Module Code:	DIG5127 SUNWAY S2 B and D
Assessment Title:	Design and development of a database-driven web application
Assessment Type:	CWRK Weighting: 100%
College:	College of Computing
Module Co-ordinator:	Sujan Shrestha (Sunway) Emmett Cooper (Birmingham City University)
Hand in details:	Formative Draft: see Assessment schedule on Moodle for due date - Application, exported SQL file and ERM all zipped and submitted through Moodle. - Individual reflective questionnaire filled out and submitted via Moodle. Summative Final: see Assessment schedule on Moodle for due date - Application, exported SQL file and ERM all zipped and submitted through Moodle. - Individual reflective questionnaire filled out and submitted via Moodle.
Return of Feedback date and format	20 working days from date of submission (see Moodle for details).
Re-assessment hand in deadline date:	See resit within Assessment schedule on Moodle for due date Note: the reassessment work may be different.
Support available for students required to submit a re-assessment:	Timetabled support sessions will be arranged for the period immediately preceding the hand-in date.
NOTE:	At the first assessment attempt, the full range of marks is available. At the re-assessment attempt the mark is capped and the maximum mark that can be achieved is 40%.

BIRMINGHAM CITY UNIVERSITY
FACULTY OF COMPUTING ENGINEERING AND THE BUILT ENVIRONMENT
COURSEWORK ASSIGNMENT BRIEF
DIG5127 Database and Web Application Development

Assessment Summary	In groups of two to three students[1] you will create a web application that provides a useful real world service which utilises a MySQL/MariaDB based database to handle data using CRUD operations (Create, Read, Update, Delete). The application should be accessible through a web browser and ideally be able to adapt to different web browsers and screen sizes.
	An ERM diagram of the database design should be produced as part of this process, to demonstrate you and your team's understanding of how to plan out a basic relational database structure.
	Your progress and completion of milestones will be assessed at specific points during the module, which will be indicated on Moodle.

MPORTANT STATEMENTS

Undergraduate Regulations

Your studies will be governed by the BCU Academic Regulations on Assessment, Progression and Awards. Copies of regulations can be found at https://www.bcu.ac.uk/student-info/student-contract

For courses accredited by professional bodies such as the IET (Institution of Engineering and Technology) there are some derogations from the standard regulations, and these are detailed in the academic regulations.

Cheating and Plagiarism

Both cheating and plagiarism are totally unacceptable, and the University maintains a strict policy against them. It is YOUR responsibility to be aware of this policy and to act accordingly. Please refer to the Academic Registry Guidance at https://icity.bcu.ac.uk/Academic-Services/Information-for-Students/Assessment/Avoiding-Allegations-of-Cheating

The basic principles are:
- Don't pass off anyone else's work as your own, including work from "essay banks". This is plagiarism and is viewed extremely seriously by the University.
- Don't submit a piece of work in whole or in part that has already been submitted for assessment elsewhere. This is called duplication and, like plagiarism, is viewed extremely seriously by the University.
- Always acknowledge all of the sources that you have used in your coursework assignment or project.
- If you are using the exact words of another person, always put them in quotation marks.

---

[1] Students with support summaries are encouraged to get in touch with the module leader to discuss their arrangements

- Check that you know whether the coursework is to be produced individually or whether you can work with others.
- If you are doing group work, be sure about what you are supposed to do on your own.
- Never make up or falsify data to prove your point.
- Never allow others to copy your work.
- Never lend disks, memory sticks or copies of your coursework to any other student in the University; this may lead you being accused of collusion.
- AI tools cannot be used to write assignments as these have to be your own work. Please refer to: FAQ link

By submitting coursework, either physically or electronically, you are confirming that it is your own work (or, in the case of a group submission, that it is the result of joint work undertaken by members of the group that you represent) and that you have read and understand the University's guidance on plagiarism and cheating.

You should be aware that coursework may be submitted to an electronic detection system to help ascertain if any plagiarised material is present. You may check your own work prior to submission using Turnitin at the Formative Moodle Site. If you have queries about what constitutes plagiarism, please speak to your module tutor or the Centre for Academic Success.

Electronic Submission of Work

It is your responsibility to ensure that work submitted in electronic format can be opened on a faculty computer and to check that any electronic submissions have been successfully uploaded. If it cannot be opened it will not be marked. Any required file formats will be specified in the assignment brief and failure to comply with these submission requirements will result in work not being marked. You must retain a copy of all electronic work you have submitted and re-submit if requested.

Learning Outcomes to be Assessed:

1   Identify, document and model the data requirements of a given system using an Entity-Relationship Model (ERM).

2   Implement a database using SQL on a host DBMS.

3   Demonstrate technical skills in client and server-side Web scripting languages.

4   Apply User experience / User Interaction principle to Web application development.

 Assessment Details:

Title: Web application that utilises a back-end database

Style: Exported web application and database files along with the ERM diagram must be zipped and uploaded onto the correct Moodle submission point.

Rationale:

According to Graduate Prospects, "the role of a web developer entails building and maintaining websites and web applications." A web developer should have the essential knowledge and technical skills to meet client requirements, which can be used for a variety of firm initiatives. A web developer's major goal is to create dependable and high-performance apps that can be accessed via various internet-connected devices such as PCs, tablets, and mobile phones. Developing excellent problem-solving skills is also essential for success in online and software development industries.

Description:
In this assessment, you and your group have the freedom to choose the type of application you want to develop, ensuring it aligns with the module's requirements. Each member of your group should be able to demonstrate how they have contributed to the project through demonstrating their skills and knowledge with the four learning outcomes. As such, there should not be a situation where, for example, one student designs and implements the database while another builds the web application, instead you will want to share ideas together and contribute to each major aspect of the project – thereby allowing each of you to demonstrate skills and knowledge across all the learning outcomes for the module.

Your web application should be built using PHP (and Perl where/when appropriate)[2] and MySQL/MariaDB, two powerful tools that will enable you to create dynamic and data-driven experiences. But it's not just about functionality; your application should be engaging and visually appealing, following the principles of User Experience (UX) and User Interactions (UI). You have the opportunity to create a professional-looking application that users will love to engage with.
One crucial aspect to consider is responsiveness. Your application should adapt seamlessly to various screen sizes, ensuring a consistent and enjoyable user experience across devices. This focus on accessibility will make your application accessible to a wider audience, enhancing its impact.

To store and retrieve data, you'll be utilising a MySQL database. This means you will need to design and implement appropriate tables and queries to fetch the desired information. The ability to retrieve and display data from multiple tables will be key to creating relevant and informative pages within your application.

The application can be of your choosing, but this will need to be agreed with the tutor before you start developing the application. Here are some suggestions but you should look to develop your own idea:

- Charity website - spreading awareness for a noble cause.
- Product catalogue website - where users can browse and explore various offerings.
- Sales or rental website – users can view products to purchase or rent (you are not required to create a working eCommerce site, but users can add products into a basket)
- Holiday booking website – search and view package holidays according to destination
- Online community - forum or blog where like-minded individuals can connect and share ideas.

---

[2] If you and your team wishes to investigate using another server side scripting language, you must present evidence that you are all in unanimous agreement and then discuss this with, and receive permission from, the module coordinator **before** the second milestone hand-in.

- Online resource management system - organising and accessing valuable information.

Remember, these are just suggestions, but you are free to develop your own unique idea **within the agreed scope** as long as it appropriately meets all the requirements for the assessment[3].

You may use the Bootstrap framework **(Starter Example Templates only https://getbootstrap.com/docs/5.0/examples/)** to design and construct your application however you are not allowed to use existing themes widely available on the internet and if suspected you may fail the assessment.

You will also be graded on the originality of your code so, although you may use some existing code snippets/scripts, you should attempt to develop and code the application yourself using known methods and techniques.
Your application should also satisfy the following:

- Use PHP[4] as the server-side scripting language
  - Why?
    - We need to assess the skills and knowledge you have gained for server-side web development / programming
- Use MySQL/MariaDB to store your data
  - Why?
    - We need to assess the skills and knowledge you have gained for planning out and creating relational database systems.
- The application should be able to read and write data to a database, as well as edit and delete it (CRUD).
  - Why?
    - We need to assess the skills and knowledge you have gained in utilising and maintaining relational database systems.
- When building your database structure, adopt the relational database model (Referential Integrity).
- Be responsive to different screen sizes
  - Why?
    - We need to assess the skills and knowledge you have gained in making your system software system flexible when dealing with different target platforms – one part of this is considering the screen size of the target platforms 😉

-

The assessment will follow a normal design and development process;

---

[3] If in doubt ask your lab lecturer and the module coordinator
[4] Again, any plans to diverge from this should be approved by the module coordinator and based on a unanimous team decision **before** the second milestone submission

Planning   Design   Development   Testing   Delivery
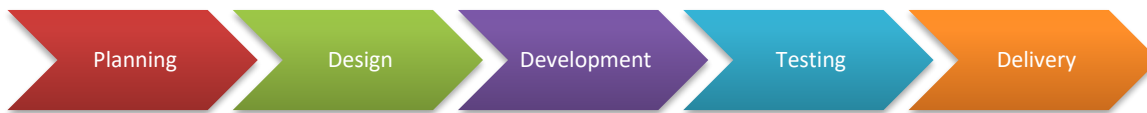
However, we recommend that you follow these stages **iteratively.**

What does this mean?
Well, considering everything up front and identifying potential pitfalls with your implementation can be very difficult for novice Engineers / Developers, so we recommend that you gradually build up your system through a cycle of repeated:

- PLANNING:
  - Produce Initial plans that set out a clear end goal of the system you intend to build
    - Imagine you want to plan out a route from Location A to Location G, you need to know
      - Your starting point (Location A)
      - Your desired destination (Location G)
  - Gradual refining these plans during each iteration
    - As you get a clearer and clearer picture of how to design and build your system you can think about the specific route you are taking – how exactly are you getting from A to G?
- DESIGN: Initial designs / Refined redesigns
- DEVELOPMENT: Initial small scale development prototyping / Refined larger scale web system
- TESTING: Testing how well your system meets the identified requirements throughout each development cycle

Planning: This is where you formulate your ideas, requirements and collect data from the review of existing applications. You should consider the audience or the purpose of the website. Write a list of interactions you propose to incorporate in your application.

Design:
From the information gathered you should first determine what information is to be stored in the database. This will then be the basis of the type of data being stored in the database. The Entity Relationship Diagram will be developed according to your specific requirements and detail a map of the database tables along with the field names and how they are linked together using primary and secondary keys. You should aim to design your database to minimise redundancy, i.e. avoid repeating data in different rows of a table or in different tables in the database.
You should also design basic layouts of how the information is to be presented. Think about the user interactions and the user journey as they navigate through your application. The user should be the centre of your design and as such consideration of the user experience

(UX) is important. You should also consider and design for different screen sizes and ensure the experience for the user is the same.

Development:

This is the main element of the assessment and where you will develop the design ideas into an application. The proposed interactions will need to be integrated into your application and how the database will be used to extract information and enable your site to be dynamic. This is an important **requirement** as it provides users with the ability to create, read, update and delete data (CRUD) in the database by the use of PHP webforms rather than direct manipulation using the database server.

Testing

In this stage, you will test for any issues and defects. You should look to fix any issues until the product meets the original specifications. You should ensure that the application reaches a quality standard. This stage may require sufficient time and resources and therefore you should include this in your plan.

Delivery:

This is the final stage of an application development and should happen after you have conducted a rigorous testing and review phase before launch (submission).

Milestones checkpoint (20% of the overall mark)

Through face-to-face practical lessons you will demonstrate your progress through milestone checks.

- Week 5: Demonstrate in class that you have set up XAMPP on a USB device, via a network drive, or on your own computer and execute basic PHP scripts from the exercise in weeks 1 and 2.
- Week 8: Provide an initial, early draft Entity Relationship Diagram which represents you and your teams' initial attempts at planning the structure of the database which will underpin your web application (to be submitted on Moodle).
- Week 10: Demonstrate in class that you have developed your initial database version and show how JOIN queries can be used to link some of these tables together appropriately. .
- Week 12: Demonstrate the progress of your prototype application so far, followed by submission of a draft copy of your work on 13th December 2023.
  - The features expected in the final product will be partly dependent on the kind of problem your are trying to solve, but by this stage you should try to demonstrate that your system can:
    - Navigate between the different major web-pages that your site consists of
    - **Read** data from the database and use it to dynamically generate some of the web page content using server-side scripting (PHP)
    - Ideally show some type of Creation of data, Updating of records or Deletion of records as appropriate based on user interactions through the web interface.

Additional information:

Use only your own code

The code and script used with your application should be you and your group's own work, and not directly copied from the Internet or a book without clearly referencing the original work. You are permitted to use the Bootstrap framework (**Starter example templates only**) to build the application however using a fully constructed template/theme/website is not permitted and if found to be copied, it will be deemed a failure.Any code or script techniques adapted or applied from the internet, a book, or other external resource (even lab notes) must be clearly acknowledged.

Make sure your application is working

Make sure you test your artefact before submission; if it does not load and work on a standard PC it cannot be assessed and will fail (see Format below and checklist)

Back up your work

Computer systems can and do fail. You should ensure that you backup all your work to an external storage device. If you lose your work, we will have no evidence that you did it.

This assessment should demonstrate the application of the various issues discussed in tutorials.

For advice on writing style, referencing and academic skills, please make use of the Centre for Academic Success: Centre for Academic Success - student support | Birmingham City University (bcu.ac.uk)

Workload:
This assessment workload is roughly equivalent to 4000 words and a typical student would be expected to spend a **minimum** of 40 hours of focused, dedicated work throughout the semester to pass this assessment.

Transferable skills:
Problem-Solving Skills
Verbal Communication
Time Keeping
Project management
Researching
Documenting

Marking Criteria:

Table of Assessment Criteria and Associated Grading Criteria

Learning Outcomes	1 Identify, document and model the data requirements of a given system using an Entity-Relationship Model (ERM).	2 Implement a database using SQL on a host DBMS.	3 Demonstrate technical skills in client and server-side Web scripting languages.	4 Apply User experience / User Interaction principle to Web application development.	1-4 Completion of milestones.
Assessment Criteria →					
Weighting:	15%	15%	40%	10%	20%
Grading Criteria **0 – 29%** **F**	No ERM model included as part of the application design or where it has been included it has not followed the correct conventions. Questionnaire questions related to the ERM design process not present or answered to an unacceptable standard.	No database or little to suggest a database has been implemented. Where it has been included, there is no indication it is being utilised. Questionnaire questions related to the Database implementation not present or answered to an unacceptable standard.	Little or no code included. Only a handful of poorly coded pages. The code is overtly large and shows lack of consideration. Nearly all of your code has been taken from various sources without any modifications or acknowledgement. Numerous errors in the code. Questionnaire questions related to the web application implementation not present or answered to an unacceptable standard.	No visual or functional design to the application. Little or no UX/UI principles applied. Even the basics has not been considered. Questionnaire questions related to the user interface design not present or answered to an unacceptable standard.	Little or no engagement with the set milestones.

30 – 39% **E**	Little or no attempt to identify and document the requirements for the database using the correct ERM conventions. Questionnaire questions related to the ERM design process not present or answered to an unacceptable standard.	A database has been included but little correlation to the ERM requirements. Where the script is connecting to the database, little to suggest the application is utilising it to extract data or CRUD functionally has been implemented. Questionnaire questions related to the database implementation not present or answered to an unacceptable standard.	The code is fairly large and appears to be patched together with little thought. Lack of logical structure to the code and repetitive. The majority of your code has been taken from various sources without any modifications or acknowledgement. Poor use of basic coding skills. Questionnaire questions related to the web application implementation not present or answered to an unacceptable standard.	Lacking in visual design and not sufficient consideration for functionality. Lacking in nearly all areas of basic UX/UI principles. Questionnaire questions related to the user interface design not present or answered to an unacceptable standard.	Some milestones have been met but to a poor standard. Not sufficient engagement.
40 – 49% **D**	Some attempt to identify and document the requirements for the database using ERM. Some of the entities, attributes and relationships have been shown but not following the correct conventions. Questionnaire questions related to the ERM design process answered to an acceptable standard, various improvements possible.	A database has been included with the application and some indication that it is connecting and extracting some data using basic queries. Not all of CRUD has been implemented. Questionnaire questions related to the database implementation answered to an acceptable standard, various improvements possible.	The code is fairly efficient but contains many errors within the construction of the application. Better use of coding structure would have been useful. A high portion of the code has been taken from various sources without any modifications or acknowledgement. Questionnaire questions related to the web application implementation answered to an acceptable standard, various improvements possible.	Adequate design of application with generally sufficient functionality. Evidence of some UX/UI principles considered and applied. Questionnaire questions related to the user interface design answered to an acceptable standard, various improvements possible.	Some indication milestones have been satisfied and to an adequate level.

50 – 59% **C**	Fair attempt to identify and document the requirements for the database using ERM. Most of the entities, attributes and relationships have been shown but further work required. Questionnaire questions related to the ERM design process answered to a reasonable standard, various improvements possible.	Reasonable implementation of a database as part of the application. Adequate use of queries but this could have been developed further. Proposed ERM not implemented entirely. CRUD may not have been implemented in its entirety. Questionnaire questions related to the database implementation answered to a reasonable standard, various improvements possible.	The code is efficient, readable and appropriately understood. Organised work with reasonable use of coding conventions. Some of the code has been taken from various sources with minimal modifications or acknowledgments. Some errors in the code. Questionnaire questions related to the web application implementation answered to a reasonable standard, various improvements possible.	Reasonable design to the application. Some aspects of UX/UI has been considered and applied but lacking in some areas. Questionnaire questions related to the user interface design answered to a reasonable standard, various improvements possible.	Milestones have been met to a reasonable level. Fair engagement with given tasks.
60 – 69% **B**	Good attempt to identify and document the requirements for the database using ERM. Entities, attributes and relationships have been shown but some information missing. Questionnaire questions related to the ERM design process answered to a good standard, various improvements possible.	Good use of queries to join at least two tables together and extract appropriate data. Implementation of the ERM proposal has been followed with some minor errors. CRUD functionality has been implemented well. Questionnaire questions related to the database implementation answered to a good standard, various improvements possible.	The code is efficient and well-organised. Logical structure that allows readability. Majority of the code is your own with some being taken from other sources and appropriately acknowledged. Some minor errors. Questionnaire questions related to the web application implementation answered to a good standard, various improvements possible.	Good attention to the detailed design of the application and careful consideration of UX/UI principles is evident. Some areas required further development. Questionnaire questions related to the user interface design answered to a good standard, various improvements possible.	Good engagement and milestones have been met to a good level.

70 – 79% **A**	Excellent demonstration to identify and document the requirements for the database using ERM. Entities, attributes and relationships have been shown correctly in great detail. Questionnaire questions related to the ERM design process answered to a very good standard, some improvements possible.	Excellent implementation of the proposed ERM. Only some minor omissions. Advanced use of SQL to join more than two tables together and extract appropriate data. CRUD functionality has been implemented skilfully. Questionnaire questions related to the database implementation answered to a very good standard, some improvements possible.	The code is very efficient without sacrificing readability and understanding. Easy to follow and interpret. Nearly all of your work is your own and where you have used known scripts, it has been appropriately acknowledged. Little or no errors, various improvements possible. Questionnaire questions related to the web application implementation answered to a very good standard, some improvements possible.	Excellent design of the application. Implementation of functionality to enhance UX/UI is clearly evident. Questionnaire questions related to the user interface design answered to a very good standard, some improvements possible.	Excellent engagement and milestones have been met to a high level.
80 – 89% **A+**	Outstanding demonstration to identify and document the requirements for the database using ERM. Very detailed information regarding entities, attributes, and relationships. Questionnaire questions related to the ERM design process answered to a great standard, some minor improvements possible.	Outstanding implementation of the proposed ERM. Little or no errors. Advanced use of SQL to join three or more tables together and extract suitable data. Outstanding implementation of CRUD. Questionnaire questions related to the database implementation answered to a great standard, some minor improvements possible.	The code is exceptionally well organised, structured, and logically planned. The application has been built from scratch with little or no use of existing scripts/code (other than the Bootstrap starter template). No errors in the code, though various improvements still possible Questionnaire questions related to the web application implementation answered to a great standard, some minor improvements possible.	Outstanding design that has allowed a refined design. The implemented functions have been completed to a high standard and UX/UI has clearly been thought through. Questionnaire questions related to the user interface design answered to a great standard, some minor improvements possible.	Outstanding engagement and milestones have been met to an exceptional level.

90 – 100% A*	Faultless demonstration to identify and document the requirements for the database using ERM. Very high level of knowledge and understanding. Impeccably informed regarding entities, attributes and relationships. Questionnaire questions related to the ERM design process answered to an excellent standard, though some very minor improvements possible.	Exemplar use of database SQL to extract data from number of joined tables. Database has been created precisely to the proposed professionally designed ERM. A very high quality implementation in all areas, some improvements may still be possible, but they are regarded as minor at this level of study. Questionnaire questions related to the database implementation answered to an excellent standard, though some very minor improvements possible.	The code structure and organisation are of a professional standard. The application has been written collaboratively together by you and your group other than the standard bootstrap starter template (if used). Some improvements may still be possible, but they are regarded as minor at this level of study. Questionnaire questions related to the web application implementation answered to an excellent standard, though some very minor improvements possible.	Impeccable design of the application. The functionality of the application is free of all major issues that might be expected at this level with little improvement required at this level of study Questionnaire questions related to the user interface design answered to an excellent standard, though some very minor improvements possible.	Very proficient in all levels. Demonstrates an exceptional, highly organised and professional attitude to completion of milestones.	

Submission Details:

Format:

The ERM should be submitted as a PDF file along with a zip archive of your final web application.

The final application files must be submitted on Moodle in a zip file using a logical and clear file structure:

- You need to submit a ZIP archive containing __only__ the files needed for your web application.
- The database should be exported as a **.sql** file and included with your site files in the zip.
- The main page of the web application <u>must</u> be named **index.php.**
- Any other resources (stylesheets, scripts, images, etc.) **must** be included within the sub-directories.
- Any of these resources should be linked using **relative URL paths** and the application must load and work properly when it is downloaded and executed on a machine with a clean version of XAMPP installed[5]

The work will be primarily assessed on standard specification PC running Windows 11 and using Google Chrome version 90 and above, Apache version 2.4.56, PHP version 8.2.4, and MySQL (MariaDB) server version 10.4.28

Make sure you test your application before submission; if the application is not executable on a standard specification Windows PC it cannot be assessed and will result in a fail. It is your responsibility to make sure your application works.

Regulations:

- The minimum pass mark for a module is 40%.
- Re-sit marks are capped at 40%

Full academic regulations are available for download using the link provided above in the IMPORTANT STATEMENTS section.

Late Penalties
If you submit an assessment late at the first attempt, then you will be subject to one of the following penalties:

- if the submission is made **between 1 and 24 hours** after the published deadline the original mark awarded will be reduced by **5%**. For example, a mark of 60% will be reduced by 3% so that the mark that the student will receive is 57%.
- if the submission is made between **24 hours** and **one week (5 working days)** after the published deadline the original mark awarded will be reduced by 10%. For example, a mark of 60% will be reduced by 6% so that the mark the student will receive is 54%.

---

[5] We want to avoid something like this happening:
https://www.reddit.com/r/ProgrammerHumor/comments/d4lenu/works_on_my_machine/
😉

- **if the submission is made after 5 days following the deadline, your work will be deemed as a fail and returned to you unmarked.**

The reduction in the mark will not be applied in the following two cases:

- the mark is below the pass mark for the assessment. In this case the mark achieved by the student will stand.

- where a deduction will reduce the mark from a pass to a fail. In this case the mark awarded will be the threshold (i.e.,40%).

Please note:
- **If you submit a <u>re-assessment</u> late, then it will be deemed as a fail and returned to you unmarked.**

Feedback:

Marks and Feedback on your work will normally be provided within 20 working days of its submission deadline via Moodle.

Where to get help:

Time will be provided in tutorials to seek assistance from the tutor, please refer to the teaching schedule for more information.

The contact details for the module coordinator are:

Emmett.Cooper@bcu.ac.uk

Students can get additional support from the library for searching for information and finding academic sources. See their iCity page for more information: http://libanswers.bcu.ac.uk/

The Centre for Academic Success offers 1:1 advice and feedback on academic writing, referencing, study skills and maths/statistics/computing. See their iCity page for more information: https://icity.bcu.ac.uk/celt/centre-for-academic-success

Additional assignment advice can be found here: https://libguides.bcu.ac.uk/MA

Fit to Submit:

Are you ready to submit your assignment – review this assignment brief and consider whether you have met the criteria. Use any checklists provided to ensure that you have done everything needed.

Assignment Checklist

Run through this simple tick list before submitting your work!

ERM

Well prepared materials make your work look more professional and easy to understand.

Item	Action	Done?
1	I have used the spellchecker and proofread the ERM correcting errors several times.	
2	I have checked that all material is directly related to the assignment tasks.	
3	I have checked that all the required information has been included in the ERM such as all Entities, Attributes and Relationships.	
4	The ERM is professionally presented using consistent headings, fonts and layout.	
5	All questionnaire questions related to the ERM design of the database have been answered individually, from my perspective, and in thorough detail.	

Application

Is your work complete? Have you included all the required elements?

Item	Action	Done?
1	You have included your final ERM design as part of your submission	
2	Your database should be identified as combination of your student IDs separated by underscores, i.e. 21123345_2312345_	
3	You have exported your database as a .sql file. It is your responsibility to check the exported sql file is correct	
4	You have included an index.php page as your homepage	
5	Your folder structure is appropriate for your application, i.e. all images in an images folder, all scripts in a scripts folder etc	
6	All your styling should be contained in your own CSS file and linked to your PHP pages using the correct link reference in your <head> tag	
7	You have used relative paths within your code rather than absolute paths	

8	Only include the files relevant to your application. You will be penalised for including any unnecessary files.	
10	It is recommended you use <?php include ….. ?> to simplify your code. For example, the database connection script should be in its own php page and the application then calls that script when it requires the database	
11	If you have a login page and you are using encrypted passwords, ensure you include a text file with the login information for testing purposes	
12	You have zipped up your application including the sql file as a single package. We will only accept a .zip file	
13	We will test your application using a standard PC running Windows with software stack version details specifies above in this document. If you have used any other localserver to build your application, it is recommended you test on our recommended set up or something as close to it as possible.	
14	We will assess your application using a standard PC running Windows and using Google Chrome version 90 and above. We will also use Firefox 118 to check for some standards compliance between browsers.	
15	It is your responsibility to check the exported application works on different computers. We cannot assess any application that fails to export correctly and it would be deemed a failure.	
16	All questionnaire questions related to the database implementation have been answered individually, from my perspective, and in thorough detail.	
17	All questionnaire questions related to the web application implementation have been answered individually, from my perspective, and in thorough detail.	
18	All questionnaire questions related to the user interface design have been answered individually, from my perspective, and in thorough detail.	