MCAST

ASSESSMENT AND INTERNAL VERIFICATION FRONT SHEET (Individual Criteria)

(Note: This version is to be used for an assignment brief issued to students via Classter)

Course Title	**Bachelor of Arts (Honours) in Interactive Digital Media**		**Lecturer Name & Surname**	**Matthew Cumbo**	
Unit Number & Title	**CAIDM-506-2206 | PHP & Databases**				
Assignment Number, Title / Type	**Back-End Shenanigans**				
Date Set	14/10/2025	**Deadline Date**	**30/01/2026**		
Student Name		**ID Number**		**Class / Group**	

Assessment Criteria	Maximum Mark
KU1 Construct basic web pages using HTML, CSS, and JS for use in a PHP project.	10
KU2 Indicate the requirements of a dynamic website using Sitemaps, UML Data Models, and IPO Charts.	10
KU3 Describe own use of programming practices in PHP and MySQL in a technical document.	10
AA1 Use basic programming practices to display content dynamically using variables, conditions, and loops.	10
AA2 Use Session Management, Cookies, and HTTP Requests and Responses to personalize website content.	10
AA3 Devise an appropriate database structure using correct data types and attributes.	10
AA4 Use PHP functions and SQL operations to create, read, update, and delete records from a database.	10
SE1 Assemble web pages from defined templates using appropriate PHP rendering techniques.	10
SE2 Manage table relationships in an SQL database using appropriate SQL operations.	10
SE3 Design a testing strategy to ensure project functionality and completeness.	10
Total Mark	100

Notes to Students:

- This assignment brief has been approved and released by the Internal Verifier through Classter.
- Assessment marks and feedback by the lecturer will be available online via Classter (http://mcast.classter.com) following release by the Internal Verifier
- Students submitting their assignment on Moodle/Turnitin will be requested to confirm online the following statements:

Student's declaration prior to handing-in of assignment

- ❖ I certify that the work submitted for this assignment is my own and that I have read and understood the respective Plagiarism Policy

Student's declaration on assessment special arrangements

- ❖ I certify that adequate support was given to me during the assignment through the Institute and/or the Inclusive Education Unit.
- ❖ I declare that I refused the special support offered by the Institute.

MCAST

Purpose and Aims

The purpose of this Unit is to teach learners the fundamental theory behind server-side web development, and consolidate knowledge by employing a hands-on approach to integrating a Server-Side programming language (PHP) with a Database using Structured Query Language (SQL). This Unit draws on the practice covered in previous units and builds on the knowledge and skills gained in implementing web design.

Scenario

Using backend Programming Languages like PHP in conjunction with Databases to store all our data, we can develop complex web applications or websites with highly developed functionalities. This unit will push you towards building a fully-fledged web application, including saving, reading, updating, and deleting data on a database through the UI designed for the user to interact with.

For this unit, you are tasked with building a School Management System. You are given a set of basic requirements to implement but will need to come up with more ideas to create more features that will make the system more usable. Along with solid implementation of all the specified features, you are also required to come up with a good design for your web application.

N.B. You are also required to use **GitHub** periodically, showing incremental progress on your project, which will also serve as your backup for your work.

MCAST - Institute for the Creative Arts
Misraħ l-Għonoq, Targa Gap, Mosta, MST 1735
+356 2398 7753 adi@mcast.edu.mt

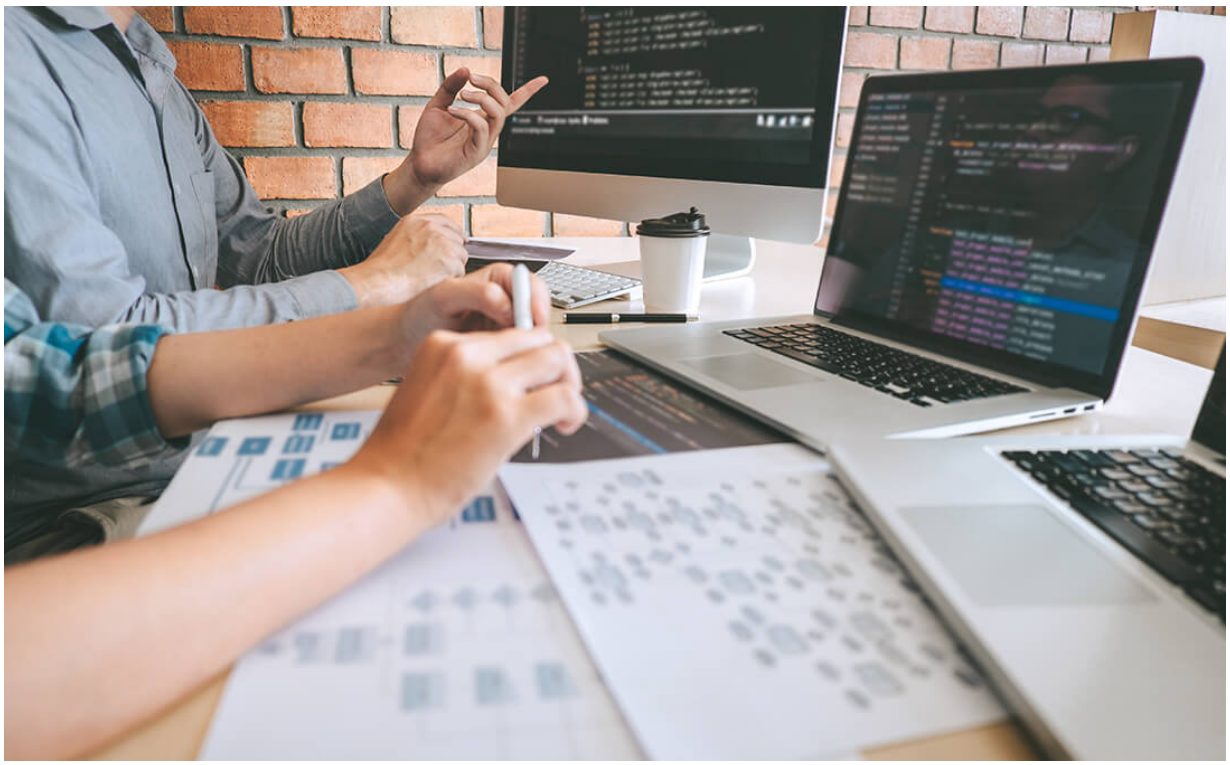
Task 1 - Planning

Before starting any work on developing an application, one should always start off with a strategy. For this task, you are to establish your planning for your web application. It is extremely important that this work is done to a high level of detail so as to have some solid groundwork for the following tasks.

Before starting any code-related work, you are to generate a **Planning Document**. This document should include the following:

1. A detailed description of your project
    a. This must include details about each feature that will feature in your project
2. A Sitemap showcasing the Navigational Structure of your web application
3. An ERD showing your Database Structure, including data types, relationships, etc
4. An IPO Chart, setting up your testing strategy to be followed once development has started
    a. Each sub-system within your project should have its own IPO Chart

Note: Although not required, you are encouraged to also set up a Style Guide and construct Wireframes for your web pages, as was done in previous units, as these will help you later on during this project.



MCAST

You web application **must** eventually, at minimum, include the following features:

1. Account Management
    a. Admin users can register new Admins, Lecturers, and Students
    b. Admin users can deregister other users
    c. Login feature
    d. Edit Account details
2. Course Management
    a. Admins will be able to create, update, and delete courses
3. Unit Management
    a. Admins will be able to create, update, and delete units
    b. Admins will be able to link a unit to a course
    c. Admins will be able to enroll students to a unit
    d. Admins will be able to assign a unit to one or more lecturers
4. Assignment Management
    a. Lecturers will be able to create, update, and delete one or more assignments linked to a particular unit
    b. This includes the ability to upload a file (the assignment brief)
    c. This file must be visible to the enrolled students
5. Submission Management
    a. Students will be able to upload their work (this can be multiple files)
    b. These files must be visible to the lecturer who is assigned this unit
6. Grading
    a. Lecturers will be able to give a mark to a student's work
    b. This mark will be visible to the student whose work is being graded

IMPORTANT: Your database must be composed of **at least 15 tables**, including intermediary tables.

Submission: 1 PDF Document via Moodle

File Name: "Planning Document - [your full name]"

Deadline: 14th November 2025

Grading Criteria: *KU2*

MCAST - Institute for the Creative Arts
Misraħ l-Għonoq, Targa Gap, Mosta, MST 1735
+356 2398 7753 adi@mcast.edu.mt

MCAST

Task 2 - Project

For this task, you will be building your web application based on the planning outlined in your Planning Document. Your project is to be hosted on GitHub for backup and version control purposes.

Part 1

For the first part of this task, you are to build a database following the **ERD** you submitted for the previous task. Using correct implementations, the database must include the following:

1. Tables
2. Fields
3. Data Types
4. Constraints
5. Relationships

Part 2

Once the database is set up, you need to **develop your web application**, making sure to:

1. Connect to the database using the correct PHP functions
2. Execute SQL CRUD Operations from within your PHP files
3. Use basic PHP syntax to render your pages
4. Use Session Management to implement a Login feature
5. Use Cookies and Requests where necessary to personalize the content of your web application
6. Use PHP Templating system to reduce unnecessary duplicated code

Important Notes:

- Make sure that your HTML and CSS is valid throughout your project. Bootstrap can be used to aid development of this project. JavaScript and jQuery can also be used to enhance the UX of this project.
- Make sure to include inline comments in your project, especially to explain the more complex parts.
- You are required to back up your work using GitHub. It is recommended that you use branches to define new functionalities and merge them into the main branch when they are done.
- When pushing Commits to your repository, make sure to add a proper description to explain the work done within that Commit.
- Make sure to include a detailed README.md file documenting your project.
- Make sure that an export of your database is included in your repository.

Submission: Link to Public GitHub Repository via Moodle

Deadline: 23rd January 2026

Grading Criteria: KU1, AA1, AA2, AA3, AA4, SE1, SE2

MCAST - Institute for the Creative Arts
Misraħ l-Għonoq, Targa Gap, Mosta, MST 1735
+356 2398 7753 adi@mcast.edu.mt

MCAST

Task 3 – Evaluation and Testing

For your final task, you are to write up a **Technical Document** which will represent your experience in developing your web application.

Without using any screenshots of your code, make sure to:

1. Explain how a virtual server was set up locally on your device to mimic a live server
2. Explain how your database was set up
3. Explain techniques used to build a dynamic web application
4. Explain techniques used to manipulate data in your database through your web application
5. Include Test Cases following the process detailed in your IPO Chart in Task 1
    a. Make sure this includes all interactive elements in your project, and that any failed tests are redone until they pass
    b. This testing process should start during Task 2 and is finalised at this point

Submission: 1 PDF Document added to the same GitHub repository hosting your web application

File Name: "Technical Document – [your full name]"

Deadline: 30th January 2026

Grading Criteria: *KU3, SE3*

MCAST

Minimum Evidence List

1	**Task 1:** 1 PDF document submitted via Moodle	☐
2	**Task 2:** Link to Public GitHub Repository submitted via Moodle	☐
3	**Task 2:** All project files added to the submitted repository	☐
4	**Task 3:** 1 PDF Document submitted via Moodle	☐

Print this page and hand in with your assignment on final hand in date.