University of Lincoln Assessment Framework
Assessment Briefing Template 2024-2025

1. Module code & title	**CMP9134 Software Engineering**
2. Assessed learning outcomes	• [LO1] Critically apply software engineering principles and techniques to software engineering problems, taking into account recent advances in the field. • [LO2] Analyse, develop and evaluate a software artefact from inception to deployment employing professional engineering approaches to developing software systems. • [LO3] Apply social, ethical and professional practices and critically analyse their applicability to current software engineering systems.
3. Assessment title	Assessment 1
4. Contribution to final module mark (%)	**100%**
5. Description of assessment task	This is Assessment 1 and is an **individual** assignment. For this assignment you need to design, develop, evaluate and document a comprehensive web application for searching open-license media. The project will demonstrate your ability to apply advanced software engineering principles and techniques to create a fully functional web-based software system. You will be evaluated on your ability to apply software engineering principles, create a functional and user-friendly application, and critically reflect on the technical and ethical aspects of the project. **Deliverables:** • A public GitHub repository with complete source code, documentation and evidence of required engineering practices • A detailed project report (PDF) • A 5-minute video demonstration of the artefact **The artefact:** The primary objective is to create a web application that:

1. Provides a robust user account management system with
    a. User registration and authentication
    b. Ability to manage (save, retrieve, delete, etc) recent searches in the application
    c. Secure user data handling
2. Implements a full-featured media search interface
    a. Integration with Openverse API (https://api.openverse.org/v1/) for searching and retrieving openly-licensed media.
    b. Graphical web interface for searching and browsing media
    c. Advanced search and filtering capabilities
    d. Ability to play and display media results
3. Demonstrates software engineering best practices
    a. Modular and scalable architecture using well established design patterns and OOP principles
    b. Containerisation with Docker
    c. Automated building strategy
    d. Automated testing strategy
    e. Clean, well-documented code
    f. Efficient API integration with Openverse and any other API services you would like to use (for example, for OAuth 2.0)
4. Provides a comprehensive documentation enabling re-use and extensions.

The artefact can be programmed in any OOP language, can be composed of a backed and frontend component or be a standalone application.

The report:

The report should be structured in the following sections.

1) <u>Introduction (Critical Reflection on Software Engineering Principles)</u>

When crafting the introduction, you should go beyond a mere description of the project. This section is your opportunity to demonstrate your understanding of software engineering principles. Begin by contextualising your web application within the broader landscape of current software development trends.

You are expected to:

- Introduce the project and the context behind it
- Critically analyse recent advances in software engineering
- Demonstrate understanding of development approaches
- Use relevant academic references to substantiate your choices
- Provide a clear rationale for the technological and methodological decisions

Aim to answer questions like:

- Why did you choose this specific approach to building the media search application?
- How do your chosen technologies reflect current best practices in software engineering?
- What recent academic research informed your design decisions?

2) Project Planning

This section is crucial for demonstrating your ability to apply software engineering principles systematically.

Key elements to include:

- Architectural diagrams showing system components and use of OOP, design and architectural patterns
- Sequence/interaction diagrams illustrating user interactions
- Detailed project timeline with realistic milestones
- Risk assessment and mitigation plans

3) Project Management and Development Process

This section, together with evidence from your GitHub repository, should provide a transparent view of your development workflow. **You should provide the link to your public GitHub repository here.**

Demonstrate your proficiency with professional software engineering practices illustrating:

- Development logs evidencing the progress over time and how you kept track of it
- Evidence of feature-driven development
- Documentation of development decisions and trade-offs
- Describe how you iteratively developed and improved the application
- Provide any additional information that is not evidenced by the GitHub repo. For example, diagrams, user stories, or screenshots of other tools utilised

4) Testing, Building, and Containerisation

Demonstrate your approach to ensuring software quality and reliability, using GitHub workflow functionalities and/or others tools.

What to include:

- Document your testing methodology
- Provide evidence of test cases and coverage
- Explain your containerisation approach

	• Showcase how your strategies ensure portability and scalability 5) <u>Social, Ethical, and Entrepreneurial Implications</u> In this section use your critical thinking to discuss the broader context of your software. Consider the impact of these factors both in the context of the application you have developed and in the current software development landscape: • Data privacy and security • Potential societal impacts of digital technologies • Entrepreneurial opportunities • Ethical considerations in software development 6) <u>References</u> Provide references of any academic work cited in the previous sections following the University of Lincoln Harvard style referencing guidelines: https://guides.library.lincoln.ac.uk/referencing/harvard **The video:** The video must briefly show how one can setup of your developed system and demonstrate its functioning. In addition to recording your screen for the demonstration, you should record yourself with video and audio and verbally describe what you are doing during the demo. The video must not be longer than 5 minutes in duration.
6. Assessment submission instructions	This assessment is composed of 3 components: 1. A report to be electronically submitted to the Turnitin upload area of this assessment. Your work must be submitted in a single **PDF format**. DO NOT include this briefing document with your submission. **2.** A **public** GitHub repository containing your artefact. **The link to the repository must be included in the report and clearly accessible**. **3.** A video demonstrating how to setup and start your artefact and demonstrating its features. The video file must be uploaded in the **Supporting Documentation upload area and compressed as zip file; the video must be in .mp4 or .avi format and have a max duration of 5 minutes.** All these components are assessed individually. Your work must be presented according to the University of Lincoln School of Engineering and Physical Sciences guidelines for the presentation of assessed written work.

7. Date for return of mark and feedback	Please see the **Hand In Dates.xls** spreadsheet. Note: *all marks awarded are provisional until confirmed by the Board of Examiners.*
8. Feedback format	Summative feedback will be provided on Blackboard according to CRG criteria (see CRG file).
9. Use of Artificial Intelligence (AI) in this assessment	You may **not use Artificial Intelligence (AI)** in this assignment. This means that you may not use any AI technologies including Grammarly, CoPilot, QuillBot and others. If you are not sure whether you should be using a particular tool then ask your module leader first.
10. Marking criteria for assessment	A Criterion Reference Grid (CRG) is used to evaluate your learning against a set of pre-defined criteria.
11. Additional information (support, advice, tips etc)	Students are encouraged to seek assistance from any member of the delivery team for support or tips on this assessment.
12. Important Information on Dishonesty, Plagiarism and AI Tools	University of Lincoln Regulations define plagiarism as *'the passing off of another person's thoughts, ideas, writings or images as one's own...'*. Examples of plagiarism include the unacknowledged use of another person's material whether in original or summary form. Plagiarism also includes the copying of another student's work'. Plagiarism is a serious offence and is treated by the University as a form of academic dishonesty. For more information on examples of Academic Offences, please see the **Academic Offence Guidance**. Please note, if you use AI tools in the production of assessment work **where it is not permitted**, then it will be classed as an academic offence and treated by the University as a form of academic dishonesty. Students are directed to the University Regulations for details of the procedures and penalties involved. For further information, see www.plagiarism.org