

2022-2023 F21GA Assessed Coursework

TV show materials to showcase your modelling, animation, and interaction.

3D Graphics and Animation

The course assessment consists of 100% coursework (no exam). There are **three** deadlines for a total of 100 marks (equivalent to 100% of your overall mark). Please note, that for the assessment of this course, you will be expected to demonstrate incremental work.

Overview of the Coursework

This coursework is an **individual** project, and you will be asked to produce, explain and showcase your work using your own code, screenshots and videos. You will need to submit your original solution for this coursework, including models, written code, and markdown explanation files, using GitHub.

The primary learning task of this coursework is to **create materials for a TV show intro**. The main motivation behind this task is that you will be able to experience creating various models, adding animation and coding interaction to your scene. Also, the different outputs from this coursework could be used as a showcase of your skills when transitioning to industry.

For the coursework, you will need to create a total of three component outputs:

1. A **rendered image** of a scene with various modelled objects showcasing the theme of your show.
2. A **video sequence animation** of your scene with a fly through camera.
3. An **interactive application** of your scene for your show.



By completing this coursework, you will investigate several core concepts in computer graphics theory, use industry-standard tools, develop advanced programming skills in 3D graphics and animation, and have a visual output for your university portfolio to show to future employers.

Tools, Libraries, and Environment

The following software technologies *are available* for the development of your coursework:

- Visual Studio Code (<https://code.visualstudio.com>)
- Blender 3.0+ (<https://www.blender.org>)
- Unreal Engine 5 (<https://www.unrealengine.com/en-US>)
- GitHub Desktop (<https://desktop.github.com>)

The following open-source libraries *are recommended* to complete the interactive component section of the coursework:

- GLFW Library (<https://www.glfw.org>)
- GLAD Library (<https://github.com/Dav1dde/glad>)
- Dear ImGui (<https://github.com/ocornut/imgui>)
- GLM Library (<http://glm.g-truc.net>)
- Tiny GLTF (<https://github.com/syoyo/tinygtf>)

The following open-source programming interfaces *are required* for this coursework:

- OpenGL core version 4.1 or later (<https://www.opengl.org/>).

These tools are available on the Edinburgh (GRID DL) and Dubai lab machines. If you want to develop the coursework on your laptop, you will be required to install some or all of the above software. Please follow the links above for download, installation and usage instructions.

Modern OpenGL is a requirement for the interactive section; therefore, coursework running in other interfaces (e.g., Unity, DirectX, WebGL, or legacy OpenGL) will receive no marks. Please note, you will need to extend the provided code template, and code submitted not using the code template will receive no marks. Finally, screenshots, videos and the code of your work will need to be submitted to GitHub, and you will be required to show your incremental work during the semester.

Please note, markers reserve the right to ask additional questions or request you to demonstrate your outputs after submission. This request can be on campus or as a remote viva session if the marker is in doubt a section of your coursework. No marks will be given for sections if you fail to explain or present your work on request.

Coursework Component Deliverables

The assessed coursework consists of understanding, designing and implementing 3D graphics using different tools and a graphical API. This coursework will allow you to demonstrate the various concepts learned throughout the course. The groundwork necessary to complete the assessment will be possible with lecture materials, lab materials and support from the lab sessions during weeks 2 to 11.

There are three coursework deliverables:


1. **A high-quality offline rendering of your TV show idea.** This deliverable is a rendered image of your scene including **six different models**, plus any necessary lights, materials and scene structure items.
2. **A video sequence animation of your scene.** Export your models (and animation if created in Blender) from the modelling tool into Unreal Engine. Replace and fix any pipeline problems (e.g. light sources, textures, positions) **and animate a video sequence** using the Engine, Blueprints and the Sequencer. Save this animation sequence as a video.
3. **An interactive visualiser.** Export your objects from your modelling package into the template graphics API code, and create an interactive animation to show an 'alive' scene. You will need to add an **exploration camera** (moves with keys or has a defined path – not static), simple **materials, lights, a framebuffer effect** and **keystroke interactions** to make your scene alive (e.g. add moving objects, loop interactions, changes in lights, wind, etc.).

You are encouraged to design a simple, well-defined intro that is creative, fit the coursework requirements, can be implemented in the allowed time, and demonstrates your work. Campus lab sessions in weeks 3 - 9 can be used for **formative assessment** of your outputs if you are doubtful of your idea or struggling to scope it.

The submission of the deliverables is using a **GitHub repository**. You will need to create the correct file and folder structure as described below to show and explain your work. Please use Markdown pages (.md) for explanations, Images and Videos for outputs, and your versioned code.

Please note GitHub has file limits. We are expecting model files, videos and code files to be less than 25MB. If files are larger than these limits, then please reduce the geometry of your models, compress your video using a modern format, or simplify your output. Extremely large repositories which can't be easily downloaded **will not be marked**.

Main repository structure:

 Readme.md (Explain main idea or theme for your coursework)






 **Render**

 **Animation**




 **Interactive**

 **Appendices**




Render Folder Structure:

-  Readme.md (Explain the modelling process for each object (e.g. reference, modelling and materials), plus how you set up the scene (e.g. lights and objects), and the render output (e.g. settings)).
-  Blender_Project_Files_V1 (include your models. Versioning might not work, so include various snapshots of your files).
-  Blender_Project_Files_V2
-  Blender_Project_Files_V3
- ...
-  Final_Render.jpg (the final output image from Blender – jpg or png).

Animation folder structure:

-  Readme.md (Explain how the objects were imported to Unreal, any pipeline fixes and issues sorted, and how you created your animation (e.g. moving objects, moving camera, other animations)).
-  Unreal_Project (include Unreal Project – if too large, please optimise, remove large assets – use gitignore).
-  Video.mp4 (video output from Unreal Engine – mp4 – needs to be small in file size).

Interaction folder structure:

-  Readme.md (Explain in detail how the models are loaded, how the scene was built, the shaders and materials used, how the lights work, and how the interaction works).
-  Code_Project (include project files – if too large, please optimise – use gitignore).
-  Video.mp4 (2-3 minute screen capture video showing application running plus interactions).

Appendices folder structure:

 Readme.md (Anything extra that you might want to explain – use folders if needed).

Important: Any attempts to conceal or fake the fundamental structure of the repository (e.g. dates, rebase, or other) will be considered cheating and you will be reported to the School's disciplinary committee.

Please see below the marking rubrics for the coursework, so you know which areas you need to cover in the explanation files and outputs.

Please note for this coursework, it is expected from you to code and visually demonstrate your work instead of writing large volumes of text, so please keep the explanations more visual and the text relatively concise. When explaining sections of your modelling or code, please use images and links to the code in the repository. As a rough guideline, we are not expecting you to exceed more than 4000 words (around eight pages of full text), not including code for all the explanations.

As a **Masters' level student**, you are also required to **include some critical analysis and reflect** on your explanations. This include the critical assessment and reflection of your methods, code, procedures, designs and development. For example: why did you use certain method to create your model? What other techniques you could have used? Why you choose certain procedures? What went right and what can be improved? Can you add some recommendations?

If needed, you can use an appendix folder to explain parts further. This appendix section will not count for your maximum page limit.

Submission

The deadlines for submitting the deliverables are:

- (D1) GitHub repo setup and share with instructor - **Wednesday 12th of October 2022 (23:59)** (Week 5) (10%),
- (D2) Render image and game engine animation video - **Wednesday 9th of November 2022 (23:59)** (Week 9) (45%),
- (D3) Real-time interactive application - **Wednesday 30th of November 2022 (23:59)** (week 12) (45%).

You will need to submit your repository link to Canvas. Please note, late submissions will be subject to the standard penalties as defined below.

Please make sure your repository is accessible to S.Padilla@hw.ac.uk (Edinburgh) or R.Soobhany@hw.ac.uk (Dubai). **Please do not make your repository public.**

Notes

- Any legacy code (immediate mode or OpenGL GL_ARB_compatibility extension) and external code or libraries will receive no credit towards your overall mark.
- Coursework reports must be written in your own words, and any code in your coursework must be your code. If some text or code in the coursework has been taken from other sources, these sources must be referenced appropriately and should be open for use.
- Failure to reference work that has been obtained from other sources or to copy the words and code of another student is plagiarism, and if detected, this will be reported to the School's Discipline Committee. If a student is found guilty of plagiarism, the penalty could involve voiding the course.
- You must never give hard or soft copies of your coursework reports or code to another student. You must always refuse any request from another student for a copy of your report and code.
- Sharing a coursework report and code with another student is collusion, and if detected, this will be reported to the School's Discipline Committee. If found guilty of collusion, the penalty could involve voiding the course.

Marking Guidelines

Marking rubrics for the coursework is attached at the end. The marker can use detailed information from the projects and code to reinforce the coursework mark if needed.

Late Submissions

The University recognises that, on occasion, students may be unable to submit coursework on the submission date or be unable to present their work on the submission date. In these cases, the University's Submission of Coursework Policy outlines are:

- No individual extensions are permitted under any circumstances.
- Standard 30% deduction from the mark awarded (maximum of five working days).

- In the case where a student submits coursework up to five working days late, and the student has valid mitigating circumstances, the mitigating circumstances policy will apply, and appropriate mitigation will be applied.
- Any coursework submitted after five working days of the set submission date shall be automatically awarded a no grade with no formative feedback provided.

Marks and Feedback

You will receive formal feedback during the regular tutorial/lab sessions. Summative feedback and marks will be delivered to you shortly after the deadline and no more than three teaching weeks after the deadline (in line with the Universities Feedback on Coursework Policy). Feedback will be given using the marking rubrics ranking and criteria for the sections, but if needed, you can request further feedback from the marker on a live session/meeting.

Any problems?

Please do let us know if you have any problems. Or please contact your Personal Tutor or Year Coordinator if you are unable to meet the deadlines or need information for Temporal Suspensions of Studies (TSS) or Mitigating Circumstances (MC).

If you have any questions or queries about the assessment, please do not hesitate to contact S.Padilla@hw.ac.uk (Edinburgh) or R.Soobhany@hw.ac.uk (Dubai).

Coursework Marking Rubrics

| # | Assessment Output and skills | Ranking and Criteria | | | |
|---|---|--|--|--|--|
| 1 | (D2) Offline Render Modelling quality | (0.0) There is no evidence of modelling or editing of the models for the scene. How to improve? Need to explain how your models were created or cite models source and changes. | (0.0-4.0) There is some evidence of modelling and/or editing work for your models in the scene. How to improve? The modelling can do with more work. | (4.0-7.0) There is good evidence of modelling or editing effort for the models in your scene including explanations (inc criticality and reflections) and incremental work. How to improve? You were needed to add more, or more explanations for your modelling or editing of your models. | (7.0-10.0) There is significant evidence of modelling or editing effort for the models in your scene, including good explanations of the work, and good evidence of incremental work (2-3 weeks). How to improve? No need – well done. |
| 2 | (D2) Offline Render Materials quality | (0.0) There is no evidence of materials in the models of the scene. How to improve? Need to explain how your materials were created or cite sources and changes. | (0.0-2.0) There is some evidence of materials and/or editing work for your materials in the scene. How to improve? The materials can do with more work. | (2.0-4.0) There is good evidence of materials or editing effort in your scene, and good explanations (inc criticality and reflections) and incremental work. How to improve? You were needed to add more work, explanations for your materials or editing of your materials. | (4.0-6.0) There is significant evidence of materials or editing effort for the materials in your scene, including good explanations of the work, and good evidence of incremental work (2-3 weeks).. How to improve? No need – well done. |
| 3 | (D2) Offline Render Scene quality | (0.0) There is no evidence of a scene composition or lights in the scene. How to improve? You needed to create a concise scene and add lights. | (0.0-1.0) There is some evidence of compositions and lights in the scene. How to improve? Composition and lights can do with more work. | (1.0-2.0) There is good evidence of compositions and lights in your scene, explanations (inc criticality and reflections) and incremental work. How to improve? You are needed to add more effort here. | (2.0-4.0) There is significant evidence of composition and lights, including good explanations of the work, and good evidence of incremental work (2-3 weeks). How to improve? No need – well done. |
| 4 | (D2) Offline Render Scene rendering | (0.0) There is no evidence of a rendered image. How to improve? Need to create a rendered output. | (0.0-1.0) There is some evidence of a rendered image. How to improve? Output can be improved and explained. | (1.0-2.0) There is a good rendered image, and good explanation (inc criticality and reflections) of how it was created (e.g. camera, render settings). How to improve? You are needed to add more explanations for your scene. | (2.0-5.0) Great output image and explanation and creativity. How to improve? No need – well done. |
| | | | | | |

| | | | | | |
|---|---|---|--|---|---|
| 5 | (D2) Engine Animation Workflow Quality | (0.0) There is no evidence of work in exporting and exporting the scene from modeller to Engine. How to improve? Need to explain your work – a video was not provided or provided work without evidence. | (0.0-3.0) There is some evidence of work building the scene in the Engine, but there are issues. How to improve? You can add more work to improve your loading of models in the Engine (say fix textures). | (3.0-6.0) There is good evidence of effort importing the models in your scene, explanation (inc criticality and reflections) and incremental work. How to improve? You needed to add more work or explanation on how you did things and what workflow issues you resolved. | (6.0-8.0) There is significant evidence of importing and workflow effort for the scene, including good explanations of the work, and good evidence of incremental work (2-3 weeks). How to improve? No need – well done. |
| 6 | (D2) Engine Animation Animation quality | (0.0) There is no evidence of work in animating the scene in the Engine. How to improve? Need to explain your work – a video was not provided or provided work without evidence. | (0.0-3.0) There is some evidence of work animating the scene in the Engine, but there are issues. How to improve? You can add more work to improve your animation of models in the Engine (say, add more moving objects or change in light conditions or camera). | (3.0-6.0) There is good evidence of animation in your scene, explanation (inc criticality and reflections) and incremental work. How to improve? You were needed to add mor effort, or explanation on how you did things and what animation you created. | (6.0-8.0) There is significant evidence of animation effort for the scene, including good explanations of the work, and good evidence of incremental work (2-3 weeks). How to improve? No need – well done. |
| 7 | (D2) Engine Animation Output quality | (0.0) There is no evidence of output from the Engine. How to improve? You needed to compile a video. | (0.0-1.0) There is a video, but it has issues. How to improve? You can check the output format and improve on the settings. | (1.0-2.0) There is a good rendered video, but there is no explanation (inc criticality and reflections) of how it was created (e.g. camera, render settings). How to improve? You are needed to add more explanations for your scene. | (2.0-4.0) There is good video output and running video files in the repository, plus it is creative. How to improve? No need – well done. |
| | | | | | |

| | | | | | |
|----|---|---|---|---|--|
| | | | | | |
| 8 | (D3) Interactive Output Importing and Drawing | (0.0) There is no evidence of importing object files into the Graphical API. How to improve? Need to explain your work – a video was not provided or provided work without evidence (code). | (0.0-4.0) There is some evidence of work importing the objects, but there are issues. How to improve? You can add more work to improve your loading of models and drawing (e.g. not placed in the correct location or broken geometry). | (4.0-10.0) There is good evidence of work importing objects from your scene, explanation (inc criticality and reflections) and incremental work. How to improve? You were needed to explain how you did things and how everything was loaded and placed. | (10.0-15.0) There is significant evidence of work importing objects from the scene, including good explanations of the work, and good evidence of incremental work (3-4 weeks). How to improve? No need – well done. |
| 9 | (D3) Interactive Output Materials, Light and Shading | (0.0) There is no evidence of materials or lights in the scene using the Graphical API. How to improve? Need to explain your work – a video was not provided or provided work without evidence (code). | (0.0-4.0) There is some evidence of work materials and lights the scene using the Graphical API, but there are issues. How to improve? You can add more work to improve your materials and lights like loading their textures. | (4.0-10.0) There is good evidence of work in lights and materials in the scene, explanation (inc criticality and reflections) and incremental work. How to improve? You were needed to add more effort or explain how you did things and how everything was loaded and placed. | (10.0-15.0) There is significant evidence of work in light and materials in the scene, including good explanations of the work, and good evidence of incremental work (3-4 weeks). How to improve? No need – well done. |
| 10 | (D3) Interactive Output Animation, Interaction and FrameBuffer effect | (0.0) There is no evidence of animation and interaction in the scene using the Graphical API. How to improve? Need to explain your work – a video was not provided or provided work without evidence (code). | (0.0-4.0) There is some evidence of work animating and adding interaction to the scene using the Graphical API, but there are issues. How to improve? You can add more work to improve your interaction and animation (say, adding more moving objects or changing light conditions or using the camera when pressing keys). | (4.0-10.0) There is good evidence of work in interaction and animation in the scene, explanation (inc criticality and reflections) and incremental work. Plus a framebuffer effect. How to improve? You were needed to add more effort or explanation on how you did things and how everything was done. | (10.0-15.0) There is significant evidence of work in interaction and animation in the scene, including good explanations of the work, good evidence of incremental work (3-4 weeks) and it was creative. How to improve? No need – well done. |
| | | | | | |

| | | | | | |
|----|--|---|--|--|---|
| | | | | | |
| 11 | <p>(D1) Repository</p> <p>Technical Quality, Versioning and Clearness</p> | <p>(0.0)</p> <p>There is no evidence of the repository, or not shared on time.</p> <p>How to improve?</p> <p>No access to the repository.</p> | <p>(0.0-2.0)</p> <p>There is some work in the repository, but it is confusing or too large.</p> <p>How to improve?</p> <p>Quality can be improved, including versioning, conciseness, structure and clearness.</p> | <p>(2.0-4.0)</p> <p>There is work in the repository and some structure.</p> <p>How to improve?</p> <p>Quality can be improved, including versioning, conciseness, structure and clearness.</p> | <p>(4.0-10.0)</p> <p>There is good work in the repository including good structure, appropriately sized files, and demonstration of incremental work.</p> <p>How to improve?</p> <p>No need, well done.</p> |