**Southampton Solent University**

**Assessment Brief**

**Assessment Details**

|  |  |
| --- | --- |
| Unit Title: | Advanced Games Programming |
| Unit Code: | CGP600 |
| Unit Leader: | Nick Thomas |
| Level: | 6 |
| Assessment Title: | AE2 – Individual Project |
| Assessment Number: | AE2 |
| Assessment Type: | Individual Project Submission |
| Restrictions on Time/Length : | 1000 words (report should be less than 10 pages, should also include source code, video, executable and assets as evidence) |
| Individual/Group: | Individual |
| Assessment Weighting: | 75% |
| Issue Date: | September 2017 |
| Hand In Date: | 05th January 2018 |
| Planned Feedback Date: | 4 weeks after submission |
| Mode of Submission: | Online electronic submission |
| Mode of Marking: | Face-to-Face |
| Mode of Feedback: | Face-to-Face |
| Number of copies to be submitted: | 1 (single digital copy uploaded) |
| Anonymous Marking | This assessment willbe exempt from anonymous marking as it falls within an exempt category under the University’s Anonymous Marking Policy. |

# Assessment Task

See below

# Assessment criteria

See below

## **Learning Outcomes**

This assessment will enable students to demonstrate in full or in part the learning outcomes identified in the unit descriptors.

## **Late Submissions**

Students are reminded that:

If this assessment is submitted late i.e. within 5 working days of the submission deadline, the mark will be capped at 40% if a pass mark is achieved;

If this assessment is submitted later than 5 working days after the submission deadline, the work will be regarded as a non-submission and will be awarded a zero;

If this assessment is being submitted as a referred piece of work (second or third attempt) then it must be submitted by the deadline date; any Refer assessment submitted late will be regarded as a non-submission and will be awarded a zero.

<http://portal.solent.ac.uk/documents/academic-services/academic-handbook/section-2/2o-assessment-policy-annex-1-assessment-regulations.pdf?t=1411116004479>

## **Extenuating Circumstances**

The University’s Extenuating Circumstances procedure is in place if there are genuine circumstances that may prevent a student submitting an assessment. If students are not 'fit to study’, they can either request an extension to the submission deadline of 5 working days or they can request to submit the assessment at the next opportunity (Defer). In both instances students must submit an EC application with relevant evidence. If accepted by the EC Panel there will be no academic penalty for late submission or non-submission dependent on what is requested. Students are reminded that EC covers only short term issues (20 working days) and that if they experience longer term matters that impact on learning then they must contact a Student Achievement Officer for advice.

A summary of guidance notes for students is given below:

<http://portal.solent.ac.uk/support/official-documents/extenuating-circumstances/extenuating-circumstances.aspx>

## **Academic Misconduct**

Any submission must be students’ own work and, where facts or ideas have been used from other sources, these sources must be appropriately referenced. The University’s Academic Handbook includes the definitions of all practices that will be deemed to constitute academic misconduct. Students should check this link before submitting their work.

Procedures relating to student academic misconduct are given below:

<http://portal.solent.ac.uk/support/official-documents/complaints-conduct/student-academic-misconduct.aspx>

## Ethics Policy

The work being carried out by students must be in compliance with the Ethics Policy. Where there is an ethical issue, as specified within the Ethics Policy, then students will need an ethics release or an ethical approval prior to the start of the project.

The Ethics Policy is contained within Section 2S of the Academic Handbook:

<http://portal.solent.ac.uk/documents/academic-services/academic-handbook/section-2/2s-university-ethics-policy.pdf>

## **Anonymous Marking**

A copy of the University’s Policy on Anonymous Marking, process details and student guidance on submission sheet completion can be found on the following links, which are also uploaded on the Student Portal.

### Fact Sheet:

<http://portal.solent.ac.uk/documents/academic-services/policies-procedures-guidelines/anonymous-marking-fact-sheet.pdf>

### Process:

<http://portal.solent.ac.uk/documents/academic-services/policies-procedures-guidelines/anonymous-marking-process.pdf>

## **Grade marking**

The University uses a letter grade scale for the marking of assessments. Unless students have been specifically informed otherwise their marked assignment will be awarded a letter grade. More detailed information on grade marking and the grade scale can be found on myCourse.

### Policy:

<http://portal.solent.ac.uk/documents/academic-services/academic-handbook/section-2/2o-assessment-policy.pdf>

### Fact sheet:

<http://portal.solent.ac.uk/documents/academic-services/academic-handbook/section-4/4o-grade-marking-briefing-for-students.pdf>

# Assignment Task

The assessment for this unit consists of the software design and implementation of a 3D game using a variety of good software practices and technical components, such as, work-breakdown structures, testing plans, high-performance graphics, game mechanics, optimisation, and collision detections.

You are to work **individually** for this assessment. AE2 consists of the software implementation of a 3D game, using the software design developed in Assignment AE1.

# Feedback

Feedback will be given in the laboratory session so students are able to discuss their work and justify/show any choices, issues or custom features of innovation that are relevant.

# Game Specification

As with AE1, the following requirements must be met:

1. It should be a 3D game, written with Object Oriented C++, using Visual Studio and DirectX.
2. The player should be able to move around an environment.
3. The environment should be quite large, and must have static and moving obstacles (e.g., blocks, statues, and rolling rocks). These must be models (though they can be simple if you like). Some of these objects should be able to be pushed by the player, and others should be collectable.
4. The objects in the environment should have textures and some form of lighting.
5. The player should collide with objects; either stopping for static objects, or pushing them if they are moveable, or pick them up if they are collectable.
6. The environment must contain non-player entities represented by a model, or at least a series of connected 3D shapes, using textures and lighting.
7. The entities should move around the environment in some fashion (e.g., patrol an area, chase or run away from player, and fly).
8. The entities should collide with objects. This should cause the entities to perform an action (e.g., random, predefined direction change, and fly away).
9. There should be some form of interaction between the entities and the player based on collision detection (e.g. player/entity damage/death, change of behaviour, and start conversation).

Additional grades are achieved through enhancements to the basic game requirements (e.g., innovation, sophistication, and robustness).

Examples of such enhancements include:

* extending the game to use advanced features such as physics or AI;
* using advanced DirectX/Windows/Shader techniques;
* using more robust and sophisticated techniques for key game features, such as, managing the scene and collision detection;
* optimising game performance;
* good object-orientated design, with the game and its constituent objects extensible and reusable.

These are just a small set of possibilities, look at what other games do and use your imagination to come up with others. Some of these enhancements will require additional research of 3D, object-oriented and game programming techniques not explicitly covered in the unit.

# What you need to do

You need to **individually** produce a software product and a written report documenting the software development process. The report should describe the technical effects (e.g., evidenced with screenshots), any important optimisations or relevant information. This evidence should include additional design artefacts, performance statistics, code snippets or testing results.

Using the analysis and design from the previous assignment (AE1), develop code and document the development process. The application should be developed using Object Oriented C++ in Visual Studio and DirectX plus any necessary 3rd party libraries, such as, image-loading libraries or model-loading libraries.

For clarity, this is **not a group project**. Each student needs to produce their own implementation and report based on their own original design.

Reasons for your decisions must be included, along with any problems you encountered, changes to the design, and the findings of any additional research you needed to perform and how you applied it.

Run, and document, a series of tests on your working program to show that it works correctly (as shown in the submitted video).

Finally, you should produce a reflective conclusion and discussion on your implementation decisions, including the justification of any key techniques, reasons, problems and solutions and potential improvements.

# What you are required to hand in

Each student should hand in a unique project (i.e., avoid copying or plagiarising work from other students).

**The final submission is a single .zip containing (maximum 250 Meg):**

* **.doc report (less than 10 pages)**
* **source code**
* **short video**
* **working game executable and assets**

Note: All work is expected to use correct spelling, grammar, references with images and equations cited in the text explicitly.

# Assessment Marking Scheme

The marking scheme provides a set of guidelines for the student while allowing a certain amount of flexibility and creativity. The final submission is unique to the individual student and should be customize based on their original design, specification and technical requirements (such as, lighting in different contexts, indoors, outdoors, water, tessellation, reflection, procedural content, …). The submission should be demonstrated to an instructor in the lab-session so the student is able to explain and list essential features. The submission should be done online digitally and will include a single .zip, made up of a formal report (\*.Doc), source code, video, and a working executable.

*0 – no feature*

*1 – attempt (but incorrect)*

*2 – working (basic but needs work)*

*3 – outstanding (excellent example that is faultless)*

 Report [9]

 Formatting

 Sections/Content

 Images/Equations

*Note: The report should be a professional document that explains various concepts/techniques while explaining the mechanics/principles in relation to the design (e.g., design, organisation, optimisations, graphical effects, logic, patterns, and limitations).*

 Source Code [12]

 Project/Solution Organisation

 Comments/Warnings

 Naming/Management

 MagicNums/Consts/Globals/Assert

 Version Control (GitHub 3+ weeks) [3]

 Video [3]

 Game Mechanics [9]

 Matches original specification

 Playable (no crashes/faults, ..)

 Detailed test & analysis

 Non-player entities [9]

 Interact with the player

 Entities have collision detection

 Entities logic (hide/chase)

 Interactive [6]

 Gamepad/keyboard (e.g., pause)

 Timing information

 Collision detection (player)

 Scene [15]

 Transparency

(e.g., textures, mixing, sorting)

 Skybox

(e.g., scene/background image)

 Multiple Cameras

 Explore Environment (e.g.,

mouse/key view movement)

 Procedural (e.g., height map)

 Multiple Transforms [15]

 Rotation

 Translation

 Scaling

 Combining (matrix concatenation)

 Hierarchy

 Texturing [12]

 Basic Texturing (Decals)

 Textures for Other Effects (displacement, water, ..)

 Texturing for Lighting Calculations (e.g., bump mapping)

 Text on screen (e.g., HUD)

 Lighting [12]

 Diffuse/Specular/Ambient

 Gouraud

 Phong

 Multiple Light Types

 Optimisation [6]

View Clipping

Profiling/Partitioning

 Shaders [6]

 Pixel

 Vertex

 Other (Innovative Features) [9]

 ………….

 ………….

 ………….

**TOTAL / 126**

# Assessment Grading/Learning

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CRITERIA** | **A1 – A4** | **B1 – B3** | **C1 – C3** | **D1 – D3** | **F1 – F3** |
| **IMPLEMENTATION:**  **P1,P2**  **Suitability and quality of problem solving techniques, appropriateness of design.**  **(75%)** | **Application fully realizes design.**  **Advanced functionality of significantly increased complexity beyond the standard tasks has been implemented.**  **Excellent documentation of code and problems encountered.**  **Wide ranging discussion of game/OO/3D development techniques and/or design changes.**  **The final product is functional and fulfils all User Stories** | **Application fully realizes design.**  **Additional functionality with extra levels of complexity beyond the basic requirements has been implemented.**    **Good documentation of code and problems encountered.**  **Good discussion of game/OO/3D development techniques and/or design changes.**  **The final product is functional and fulfils all Critical User Stories and most “stretch” User Stories** | **Some basic additional functionality implemented in addition to a robust and fully working basic implementation of the game, but may not fulfil entire design.**  **Some documentation of code and problems encountered.**  **Basic discussion of game/OO/3D development techniques and/or design changes.**  **The final product is functional and meets all of the critical user stories and some of the stretch stories** | **Basic implementation of the allotted tasks.**  **Little documentation of code or problems encountered.**  **Little discussion of game/OO/3D development techniques or design changes.**  **Some features may not be fully operational.**  **Final Game fulfils all Critical User Stories**  **The final product may not be completely functional and the analysis may be partially incomplete and/or inaccurate.** | **Significant deficiencies in understanding of program syntax.**  **No discussion of game/OO/3D development techniques or design changes.**  **Application missing or non-functional, or clearly does not fulfill core requirements.**  **Analysis is missing or simplistic, with extensive inaccuracies and omissions.**  **Critical User Stories not fulfilled** |
|  |  |  |  |  |  |
| **CRITERIA** | **A1 – A4** | **B1 – B3** | **C1 – C3** | **D1 – D3** | **F1 – F3** |
| **INTEGRATION PROCESS**  **AND TESTING:**  **C1**  **Documenting of combining code and testing application**  **(15%)** | **A complete documented attempt has been made to bring together individual members' work to produce a working game.**  **Different approaches are considered, and considerations for future code integration are discussed.**  **Wide range of well thought out and detailed tests used to thoroughly check program, with correct and detailed analysis.** | **A complete documented attempt has been made to bring together individual members' work to produce a working game.**  **Good set of tests, results analyzed to help resolve problems and/or inform validity of code.** | **A complete documented attempt has been made to bring together individual members' work to produce a working game.**  **Modest number of tests, results briefly analysed.** | **A documented attempt has been made to bring together individual members' work to produce a working game.** | **No or inappropriate testing.**  **Inadequate attempt at code integration.** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CRITERIA** | **A1 – A4** | **B1 – B3** | **C1 – C3** | **D1 – D3** | **F1 – F3** |
| **REFLECTIVE CONCLUSION:**  **C2**  **(10%)** | **Wide ranging and detailed discussions on problems encountered and how they were resolved are evidenced.**  **A thorough and detailed reflection on the design and development process with detailed consideration of what would be done differently next time.** | **Discussions on problems encountered and how they were resolved are evidenced.**  **Significant evidence of reflection on the design and development process, including valid strengths and weaknesses.** | **Evidence of reflection on the design and development process, including what has been learnt and what would be done differently next time.** | **Some evidence of reflection on the design and development process.**  **May be brief or unclear in parts.** | **Inaccurate and/or largely incomplete reflection and analysis of process.**  **No discussion of strengths, weaknesses, what has been learnt and what improvements could be made.** |