# Solent University

# Coursework Assessment Brief

# Assessment Details

|  |  |
| --- | --- |
| Unit Title: | Advanced Games Programming |
| Unit Code: | CGP600 |
| Unit Leader: | Philip Alassad |
| Level: | 6 |
| Assessment Title: | AE2 – Individual Project |
| Assessment Number: | AE2 |
| Assessment Type: | Individual Project Submission |
| Restrictions on Time/Word Count: | 1000 words (report should be less than 10 pages, should also include source code, video, executable and assets as evidence) |
| Consequence of not meeting time/word count limit: | There is no penalty for submitting below the word/count limit, but students should be aware that there is a risk they may not maximise their potential mark.  It is essential that assignments keep within the time/word count limit stated above. Any work beyond the maximum time/word length permitted will be disregarded and not accounted for in the final grade. |
| Individual/Group: | Individual |
| Assessment Weighting: | 75% |
| Issue Date: | 24th September 2018 |
| Hand In Date: | Wednesday 09th January 2019 4pm |
| Planned Feedback Date: | 4 weeks after submission |
| Mode of Submission: | on-line |
| Number of copies to be submitted: | 1 |
| Anonymous Marking | This assessment: Is exempt from anonymous marking. |

# Assessment 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 criteria

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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.** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Report [9]** | |  |  |  |  |  |  |  |  |
| ***Formatting*** | |  |  |  |  |  |  |  |  |  |
| 1 |  | Inconsistent fonts/headings | | |  |  |  |  |  |  |
| 2 |  | .doc. Formatted styles used. Consistent fonts. Relevant headings used | | | | | |  |  |  |
| 3 |  | Professional formatting with clear headings and contents. Implements some custom styles | | | | | | | |  |
| ***Sections/Content*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Little relevant content. Some explanation of mechanics. Some attempt at testing carried out. | | | | | | | | |
| 2 |  | Content is relevant. Further explanation of mechanics, graphics, design patterns and logic | | | | | | | |  |
| 3 |  | As above, plus links to similar games. Proper use of Harvard referencing. Appendices used. | | | | | | | |  |
| ***Images/Equations*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Fewer than 3 images, with no title or referencing, **OR** fewer than 3 equations | | | | | | |  |  |
|  |  | (Other than basic transforms etc.. Covered in class) | | | | |  |  |  |  |
| 2 |  | More than 3 referenced and titled images, **OR** 3 or more equations | | | | | |  |  |  |
|  |  | (Other than basic transforms etc.. Covered in class) | | | | |  |  |  |  |
| 3 |  | More than 3 referenced and titled images **AND** more than 3 equations | | | | | |  |  |  |
|  |  | (Other than basic transforms etc.. Covered in class), with understanding demonstrated | | | | | | | |  |
|  | **Source Code [12]** | | |  |  |  |  |  |  |  |
| ***Project/Solution Organisation*** | | | |  |  |  |  |  |  |  |
| 1 |  | Little extra OO programming, outside of lab work. Little organisation | | | | | |  |  |  |
|  |  | (indents, whitespace etc..) | | |  |  |  |  |  |  |
| 2 |  | Some attempt made at OO solution. Some extra classes created outside of lab work. | | | | | | | |  |
|  |  | Some attempt at organisation | | |  |  |  |  |  |  |
| 3 |  | Fully OO implementation. Excellent organisation (use of whitespace and indentation etc..) | | | | | | | |  |
| ***Comments/Warnings*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Few to no comments apart from those stated on tutorial sheets. Many warnings | | | | | | |  |  |
| 2 |  | Some extra commenting. Few warnings (10 or less) | | | | |  |  |  |  |
| 3 |  | Fully and clearly commented. No warnings | | | |  |  |  |  |  |
| ***Naming conventions*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Inconsistent naming conventions used | | | |  |  |  |  |  |
| 2 |  | Consistent naming conventions used | | | |  |  |  |  |  |
| ***Magic numbers*** | |  |  |  |  |  |  |  |  |  |
| 1 |  | One or more magic numbers used | | |  |  |  |  |  |  |
| 2 |  | No magic numbers | |  |  |  |  |  |  |  |
| ***Constants*** |  |  |  |  |  |  |  |  |  |  |
| 1 |  | Good use of constant values (**NOT** including CB values) | | | | |  |  |  |  |
| ***Global variables*** | | |  |  |  |  |  |  |  |  |
| 1 |  | No global variables used | |  |  |  |  |  |  |  |
|  | **Backups/Version control [3]** | | | |  |  |  |  |  |  |
| 1 |  | Evidence of some sort of backup system used | | | |  |  |  |  |  |
|  |  | (backup software, file copies and storage, Google docs) | | | | |  |  |  |  |
| 2 |  | As above, plus some attempt at version control | | | |  |  |  |  |  |
| 3 |  | Version control fully used, with correct ignore file implemented | | | | | |  |  |  |
|  |  | and evidence of 3+ weeks use | | |  |  |  |  |  |  |
|  | **Video [2]** | |  |  |  |  |  |  |  |  |
| 1 |  | Video supplied, though doesn't fully demonstrate software | | | | | |  |  |  |
| 2 |  | Video supplied and fully demonstrates software | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | **Game Mechanics [9]** | | |  |  |  |  |  |  |  |
| ***Matches original specification (reportAE1)*** | | | | |  |  |  |  |  |  |
| 1 |  | Some/major changes or completely different with little to no justification | | | | | | |  |  |
| 2 |  | Some changes with some justification/explanation. | | | | |  |  |  |  |
|  |  | May deviate from the original specification | | | |  |  |  |  |  |
| 3 |  | Fully realises original specification, with all changes fully justified | | | | | |  |  |  |
| ***Playable*** |  |  |  |  |  |  |  |  |  |  |
| 1 |  | Game plays, though crashes or has some major faults | | | | |  |  |  |  |
|  |  | (i.e. bad transforms, visual glitches, stuttering etc.…) | | | | |  |  |  |  |
| 2 |  | Game plays, with only minor faults (artefacts, slight stuttering etc..) | | | | | |  |  |  |
| 3 |  | Game fully playable with no faults whatsoever | | | |  |  |  |  |  |
| ***Test and analysis*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Little attempt at testing with no analysis/ solutions presented | | | | | |  |  |  |
| 2 |  | Some white/black box testing. Not all faults fixed, | | | | |  |  |  |  |
|  |  | though some possible solutions to problems presented | | | | |  |  |  |  |
| 3 |  | Game fully tested. Most or all faults fixed. Where fixes not implemented, | | | | | | |  |  |
|  |  | possible solutions presented with justified reasoning as to why this was not implemented | | | | | | | |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Non-player entities [9]** | | |  |  |  |  |  |  |  |
| ***Interact with the player*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Some enemies with no interaction | | |  |  |  |  |  |  |
| 2 |  | Basic interaction (simple chase, shoot in the general direction etc...) | | | | | |  |  |  |
| 3 |  | Entities fully interact with player (chase when player gets near, | | | | | |  |  |  |
|  |  | shoots at the player, runs away from the player etc.…) | | | | |  |  |  |  |
| ***Interact with environment*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Simple movement in a set path with no interaction | | | | |  |  |  |  |
| 2 |  | Simple pathfinding implemented, avoiding walls and obstacles (move back, stop etc.…) | | | | | | | |  |
| 3 |  | Advanced pathfinding implemented - explain technique researched and used | | | | | | |  |  |
| ***Collision detection*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Sphere colliders correctly implemented | | | |  |  |  |  |  |
| 2 |  | Sphere colliders and per-triangle collisions implemented | | | | |  |  |  |  |
| 3 |  | As above, also incorporating scene management (scene nodes/multi-part game objects) | | | | | | | |  |
|  | **Interactive [9]** | | |  |  |  |  |  |  |  |
| ***Gamepad/keyboard*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Windows event system used with keyboard/mouse | | | | |  |  |  |  |
| 2 |  | Direct input used with keyboard and mouse | | | |  |  |  |  |  |
| 3 |  | Separate input class implemented with direct input/Xinput | | | | | |  |  |  |
|  |  | (keyboard and mouse/gamepad/both) | | | |  |  |  |  |  |
| ***Timing information*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Simple game timer implemented | | |  |  |  |  |  |  |
| 2 |  | Delta time used for things like movement | | | |  |  |  |  |  |
| 3 |  | Time class implemented with FPS information correctly shown | | | | | |  |  |  |
|  |  | and delta time implemented for things like movement | | | | |  |  |  |  |
| ***Player collision detection*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Sphere colliders correctly implemented | | | |  |  |  |  |  |
| 2 |  | Sphere colliders and per-triangle collisions implemented | | | | |  |  |  |  |
| 3 |  | As above, also incorporating scene management (scene nodes/multi-part game objects) | | | | | | | |  |
|  | **Scene [15]** | |  |  |  |  |  |  |  |  |
| ***Skybox*** |  |  |  |  |  |  |  |  |  |  |
| 1 |  | Skybox implemented | |  |  |  |  |  |  |  |
| 2 |  | Cubemap implemented for reflections on some objects | | | | |  |  |  |  |
| ***Multiple cameras*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Single game camera works correctly | | |  |  |  |  |  |  |
| 2 |  | 1 or more additional cameras used (Top down view, mini map, Dynamic environment map) | | | | | | | |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Exploration*** | |  |  |  |  |  |  |  |  |  |
| 1 |  | Simple move around environment | | |  |  |  |  |  |  |
| 2 |  | Can fully explore and interact with environment | | | | |  |  |  |  |
| 3 |  | As above, with keys/buttons to look around and at least one extra action (jump, crawl) | | | | | | | |  |
| ***Scene building*** | |  |  |  |  |  |  |  |  |  |
| 1 |  | Simple scene created using only simple geometry (cubes, spheres etc..) | | | | | | |  |  |
| 2 |  | Simple technique used for creating scene (Text file or array etc..). Some object pooling | | | | | | | |  |
|  |  | or instancing attempted. Must use at least one object downloaded from the internet (Not cubes or spheres) | | | | | | | | |
| 3 |  | Advanced technique used for creating scene (explain technique researched and used). Full use of object | | | | | | | | |
|  |  | pooling and instancing. Must use at least one object downloaded from the internet (Not cubes or spheres) | | | | | | | | |
| ***Particles*** |  |  |  |  |  |  |  |  |  |  |
| 1 |  | Simple particle fountains/explosions used (Must be relevant i.e. not just randomly executed) | | | | | | | | |
| 2 |  | Particles change colour/shape according to use | | | |  |  |  |  |  |
| 3 |  | More advanced particles used (alpha/colour blending, textures, animated etc..) | | | | | | |  |  |
| ***Text*** |  |  |  |  |  |  |  |  |  |  |
| 1 |  | Simple text used for displaying timing information, score, health/lives etc.. (must have relevance) | | | | | | | | |
| 2 |  | More advanced text used (custom font, alpha blended etc..) | | | | | |  |  |  |
|  | **Transforms[3]** | | |  |  |  |  |  |  |  |
| ***Scale,Rotate,Translate*** | | |  |  |  |  |  |  |  |  |
| 1 |  | Implemented correctly with matrix concatenation | | | | |  |  |  |  |
| 2 |  | As above, plus Inverse and transpose (and possibly with inverse transpose combination) | | | | | | | |  |
| 3 |  | As above, fully utilising scene management and collision detection/ray casting(scene node) | | | | | | | |  |
|  | **Texturing [3]** | | |  |  |  |  |  |  |  |
| 1 |  | Basic textures used (lab work and single textures per model. | | | | | |  |  |  |
|  |  | More than one or two textures used in scene) | | | |  |  |  |  |  |
| 2 |  | Multiple textures/ textures for reflections or both | | | | |  |  |  |  |
| 4 |  | More advanced textures used (bump maps, normal maps, dynamic environment maps, | | | | | | | |  |
|  |  | alpha/colour blended maps for text/particles etc..) | | | | |  |  |  |  |
|  | **Lighting [6]** | |  |  |  |  |  |  |  |  |
| 1 |  | Basic lighting implemented (Ambient and Diffuse Gouraud lighting) | | | | | |  |  |  |
| 2 |  | Basic lighting, with specular highlight | | | |  |  |  |  |  |
| 3 |  | Phong Ambient, diffuse and specular (**MUST** use all three lighting types) | | | | | | |  |  |
| 4 |  | As above, with point light/spot light | | |  |  |  |  |  |  |
| 5 |  | Spot light/ point light has attenuation | | | |  |  |  |  |  |
| 6 |  | Multiple lights (more than one of each type of Phong Diffuse, point and spot) | | | | | | |  |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Optimisation [4] - Marks for each item** | | | | |  |  |  |  |  |
| 1 |  | Correct view/object clipping | | |  |  |  |  |  |  |
| 1 |  | Object pooling | |  |  |  |  |  |  |  |
| 2 |  | No memory leaks/dangling pointers - show evidence | | | | |  |  |  |  |
| 3 |  | BSP trees |  |  |  |  |  |  |  |  |
|  | **Shaders [3]** | |  |  |  |  |  |  |  |  |
| 1 |  | Vertex shader used for lighting | | |  |  |  |  |  |  |
| 2 |  | Pixel shader used for lighting | | |  |  |  |  |  |  |
| 3 |  | Other shaders used (Geometry, compute etc.…) | | | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | **Other Features - Marks for each item [9]** | | | | |  |  |  |  |  |
|  | ***I will accept things like shadow maps, procedural height maps,*** | | | | | |  |  |  |  |
|  | ***lighting effects (i.e. Bokeh, lens flare, god rays and Bloom.),*** | | | | | |  |  |  |  |
|  | ***Depth of field, Fog/rain shader effects, animations,*** | | | | |  |  |  |  |  |
|  | ***dynamic environment map created for advanced reflections,*** | | | | | |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| **Total :** |  | of | **100** |  |  |  |  |  |  |  |

# 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:

1. 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;
2. 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;
3. 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-principles-and-regulations.pdf?t=1534423842941>

# 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 the Student Hub for advice.

A summary of guidance notes for students is given below:

<http://portal.solent.ac.uk/documents/academic-services/academic-handbook/section-2/2p-extenuating-circumstances.pdf?t=1534423896787>

# 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/information-for-students/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>

**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 the portal and in the Student Handbook.

<http://portal.solent.ac.uk/documents/academic-services/academic-handbook/section-2/2o-annex-2-assessment-regulations-grade-marking-scale.pdf?t=1534424273208>

**Guidance for online submission through Solent Online Learning (SOL)**

<http://learn.solent.ac.uk/onlinesubmission>