UFCF9M-30-2 Game Engine Programming

Beta Feedback

TEAM NAME: Mouldy Ghost

FEEDBACK:

A really solid implementation of a number of low level systems, and majority of the aspects of the pipeline and essential gameplay elements clearly in place. This is a really strong position to iterate on to the final game!

Although a few issues!

A serious lack of commenting throughout!

Some particular things you might want to look at: DashAttack & StandardAttack have a lot of repeated code in their constructors which ought to be moved to the Attack constructor. AudioHandler feels a tad hardcoded to specific sounds and could be more data or even event driven.

MARK: 70%

|  |  |  |  |
| --- | --- | --- | --- |
| Student Name | Student ID | Weight /20 | W. Mark / 15 |
| Thomas Sylvester | 16015433 | 24 | 84% |
| Filip Skacanyi | 16015204 | 18 | 63% |
| Arthur Muddiman | 16028552 | 18 | 63% |
|  |  |  |  |
|  |  |  |  |

**Group mark distribution**

Each group will have a number of points to distribute amongst team members, according to their perceived overall contribution to the project. The overall mark for the project will be scaled according to this distribution of points, to make up each student’s individual mark for the module. The number of points allocated for a group will be 20 \* number of students in the group.

Individual student marks are determined based on the formula:

Ms = Ps / 20 \* Mg

Where Ms is the student’s mark, Ps is the points given to the student by the team, and Mg is the overall mark given to the group.

**For example:**Group A consists of 5 students, who will have 100 points to distribute amongst the team members.

Students 1, 2 and 3 are perceived to have contributed equally to the project, while student 4 has put in much more work, and student 5 much less. The team distribute their marks as follows:

1. 20 points

2. 20 points

3. 20 points

4. 30 points

5. 10 points

When marked, the project receives an overall mark of 65%. This mark is scaled as follows, for each student:

1. 20 / 20 \* 65% = 65%

2. 20 / 20 \* 65% = 65%

3. 20 / 20 \* 65% = 65%

4. 30 / 20 \* 65% = 97%

5. 10 / 20 \* 65% = 32%

**Please note:** Group weightings are intended to allow teams to reflect the reality of their development practice throughout the project. However, the module leader reserves the right to adjust or otherwise moderate the metric and/or weightings submitted in the event of exceptional group circumstances occurring.