# The Scots College Computer Studies Department Assessment Task Notification

|  |  |
| --- | --- |
| Subject | 11 Software Design & Development |
| Task ID | Task 03: Group RPG Project |
| Weighting | 30% |
| Outcomes | 1.1, 1.2, 1.3, 2.2, 4.1, 4.2, 4.3, 5.1, 5.1, 6.2, 6.3 |
| Submission Dates | **Friday 31st August** |
| Submission Method | Via Schoolbox and BitBucket. |

## RPG Project

In pairs, your task is to build an old style tile based Role­Playing Game (RPG).

Your game should:

* read in a file to create the game map. A tool has been provided to enable you to create map files easily.
* store important information such as:
  + characters,
  + levels,
  + items, and,
  + monsters.

Movement by the player should trigger events in the game. It is up to you what events these will be. An example would be a combat screen or item discovered screen, etc.

Note

The aesthetics of the game are not being marked. They need to make the game playable but should not occupy much of your time. If you use images from the Internet you will need to reference them.

# Marking Criteria

|  |  |  |
| --- | --- | --- |
| Task Component | Top Band Descriptor | Marks |
| Defining the Problem | | |
| Problem Definition | Definition of the problem highlights what needs to be achieved to complete the task, the steps that will be undertaken and possible problems. | 5 |
| Planning the Solution | | |
| System Flow Chart | Properly constructed System Flow Chart that illustrates how the combat system works in your RPG | 5 |
| Gantt Chart | A properly constructed Gantt Chart illustrating the tasks required to complete the project, the person who will be completing the task and an estimation of the time the task will take. Individual tasks within the group need to be specified. | 5 |
| Data Dictionary | A thorough Data Dictionary. Data types used are appropriate for the application. | 5 |
| Building the Solutions | | |
| Modular | One process per sub­routine, modules used where appropriate, functions used where appropriate, custom data types used where appropriate. | 5 |
| Coding Conventions | Proper names for variables, sub routines, white space and indentation | 3 |
| Combat Components | Component of the game acts a free form combat system separate from the main game.   * Player Movement * Opponents move on the map (1 are there, 2 move randomly, 3 at player) * Combat System (sophistication and elegance count) * Opponents can die * Player can die * Weapons/Items/Special Abilities * Characters can be created and level up * A variety of monsters and or traps are available to fight | 2  3  5  2  2  5  5  3 |
| Components of the Game | * Reads in your map file (Files are read in error free and displayed properly) * Character Triggers Events (Character can move around the map and trigger  events like loading new maps, starting combats, etc.) * Character Generation (Your program allows characters to be created and saved) | 2  3  5 |
| Bug Free | ­1 Mark for each major bug found (little bugs that can be fixed easily will not count). | 5 |
| Testing the Solution | | |
| Code Check­ins | Code is checked in to Bitbucket that contributes to the overall development of the application in a meaningful way. Edits and changes to theory is also contributed to Bitbucket. | 15 |
| Development Journal | A development journal is kept on the SDD forums (twice weekly posts describing development and issues faced and resolved. Comments on other’s posts required as well – 3 min) | 5 |
| Modifying the Solution | | |
| Personal Reflection | Personal reflection that is an honest appraisal of what you have done well, and where you could improve. | 10 |