|  |  |
| --- | --- |
| **new logo (Small).JPG** | Information Processing TechnologySt Mary’s Catholic College |
|  | **BMAP12 – Major Project** |
|  | **Year 11, Semester1, 2015** |
|  |  |
| **Student:** |  |  |
| **Teacher:** | Mr Hoad |  |
| **Review Due:** | **06/3/2015** | **Date Due: 30/4/2015** |
| **Unit:** | **Algorithm Design / Software Programming 1** |
| **Conditions:** | formative - unsupervised – individual6 Week to complete |
| **Submission Requirements:** | You are to submit the following:* A draft by the specified draft date
* Provide a print out of your final submission with the correct task sheet attached to the front with a completed student ownership statement.
* Place a copy of your work in the appropriate folder on

S:\studentuploads\ICT\IPT\ |
|  |  |
| **Student ownership statement:**  | I, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ declare that: 􀀔 I completed this assessment without any direct assistance from any other source, other than those listed in my bibliography or that accepted by the teacher (which includes group work) 􀀔 I completed this assessment with the following assistance: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Student signature) |
| **Standards****Awarded:** | **Knowledge and Application** |  |  |
| **Analysis & Synthesis** |  |  |
| **Evaluation & Communication** |  |  |
|  |  |  |

|  |  |
| --- | --- |
| **Task** |  |
| RequirementsProject Documentation | Create a game using one of the following three software applications:1. Blender
2. Flash
3. Visual Basic
4. Unity

Your game is NOT to involve shooting anything such as a first person shooter genre involving blood and guts. For example is not to be a ‘Left 4 Dead’ type game. Try creating a strategy or puzzle style if stuck for ideas (think strategy or maths if you are creating a game).The topic of your game or application must be approved by your teacher before you begin the design stage.The project will be submitted both in electronic and hardcopy format and must contain:1. the project documentation as a hardcopy document (*see below*)
2. a compiled copy of your game /app, ready for distribution submitted into ‘Student Uploads’ (see below)
3. all the source code of your project **which must be fully commented**

You must submit a hard copy (printed) document containing:1. project overview
2. client context summary (products / services, customers, current method of business, etc)
3. identification of problem within current client context -
4. (thus) your justification of proposed solution!
5. deliverables / objectives of proposed solution for the client (something you can measure...)
6. identification of inputs / processes / outputs
7. object / data dictionary (including variable / control name, data type [if applicable], description)
8. **user interface design** and **algorithm design**, which you must:
* appropriately name controls on user interface (eg. txtUserName, etc)
* design an efficient, modular algorithm in *pseudocode* that will power the user interface
* show which algorithm modules are invoked by which form controls (be sure to name the event eg. “cmdSave click”, etc)
* clearly differentiate between global and local variables (add a separate module to your design if necessary)
* use functions and sub-procedures to modularise your code!
1. an annotated journal of screen dumps during the build stage of your project (as evidence of your own work)
2. a one-page example of a contents page (or a readme file) for a *user instruction guide* to accompany your software
3. a discussion of problems you encountered when building, how you solved them, and which parts of your original algorithm design you had to change (and why!)
4. an evaluation of your work, which must contain:
* at least 3 completed user surveys / questionnaires
* a discussion of the results
* an overall evaluation of the project based on:
	+ efficiency of the final program code
	+ usability / user-friendliness of the finished product
	+ client objectives / deliverables – did the game / app meet client expectations?
1. future directions
 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E |
| The student: |
| K& A | **Core algorithm design** and implementation of standard learned code (ie applying the stuff given in tutorials!) |
| * **effectively and consistently selects and applies knowledge of programming structures to create a range of features including iterations, selections, arrays and record structures**
 | * selects and applies knowledge of programming structures to create most features including iterations, selections, arrays and record structures
 | * **selects and applies basic knowledge** of programming structures to create some features including iterations, selections, arrays and record structures
 | * **selects basic** programming structures to create some features
 | **recalls some**  programming structures |
| Applies **knowledge of programming language syntax** to develop software from an algorithm |
| A&S | translates algorithm design to **accurate and efficient code** with **clear internal documentation** that is well presented.  | **translates algorithm design** to mostly **accurate and efficient code** with **internal documentation** that is reasonably well presented. | creates mostly effective code with some **internal documentation**. | writes some correct **code** with basic **internal documentation**. | writes **code** |
| **Construction and presentation of evaluation** |
| use correct functional grammar and spelling and a wide range of appropriate vocabulary to accurately present ALL the required documentation in the most appropriate format | use functional grammar and spelling and a range of appropriate vocabulary to accurately present the required documentation in an appropriate format | use functional grammar, spelling and appropriate vocabulary to present the required documentation in an appropriate format | uses functional grammar and spelling to present the required documentation in a format with some consistency | uses functional grammar and spelling to produce a document  |
| E&C | **Algorithm and interface design** |
| develops an effective and efficient set of solutions to the problems presented incorporating suitably and consistently named controls with efficient, modular algorithms | develops a mostly effective and efficient set of solutions to the problems presented incorporating suitably and consistently named controls with efficient, modular algorithms | develops a sound set of solutions to the problems presented incorporating consistent naming of controls and some modularity | develops a set of solutions to the problems incorporating an algorithm and presented with some naming of controls  |  |
| **Identification of IPOs and client needs** |
| analyses the problem from multiple perspectives and determines suitable deliverables leading to the identification of clear and logical inputs, processes and outputs | analyses the problem from client perspective and determines deliverables leading to the identification of logical inputs, processes and outputs | analyses the problem from a perspective and determines basic deliverables leading to the identification of inputs, processes and outputs | analyses the problem and some deliverables leading to the identification of inputs, processes and outputs |  |
| **Evaluation**  |  |
| evaluates with detailed justification, using both prescribed and self-determined criteria and standards, and makes informed judgements about the suitability of the final solution in reference to the initial problem | evaluates with justification, using both prescribed and self-determined criteria and standards, and makes mostly informed judgements about the suitability of the final solution in reference to the initial problem | evaluates with some justification, using criteria and standards to make judgements about the suitability of the final solution | evaluates using criteria and standards to make simple judgements about the final solution |  |