# Rubrics: Examples

A rubric can focus on one specific criteria related to a content description as in this example focusing on students’ knowledge and skills related to webpages and use of CSS.

| Achievement standard  **Explain** simple data compression, and why content data are separated from presentation. | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
|  | Quantity of knowledge | | | Quality of understanding | | |
| Webpage design: Separating style from content | No evidence of understanding | Student is able to describe the purpose of CSS. | Student can apply CSS styles to a given webpage. | Student is able to design and apply CSS code to their own webpages. | Student is able to apply knowledge of CSS to their own original webpages and demonstrate different designs using identical content. |
| Optional Score | 0 | 1 | 2 | 3 | 4 |
| Why is content data are separated from presentation? | | | | | |

Page 1 of 3

| **Achievement standard (elements assessed)** | **Assessed through** | **Excellent** | **Satisfactory** | **Needs improvement** |
| --- | --- | --- | --- | --- |
| **Design** and **implement** modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities.  **Design** and **evaluate** user experiences and algorithms.  **Test** and **predict** results and implement digital solutions. | Algorithm for game  Testing process documentation | The student presents a range of algorithms that show correct branching, iteration and functions, as well as correct elements of structured algorithms (eg correct symbols/keywords). Their algorithms demonstrate the extent of the game and are clearly linked to the game plan.  The student demonstrates testing of algorithms prior to programming. They find solutions and document them in the troubleshooting guide. | The student presents algorithms that show correct branching, iteration and functions, as well as mostly correct elements of structured algorithms (eg correct symbols/keywords).  The student demonstrates testing of algorithms prior to programming. They find solutions and document them in the troubleshooting guide. | The student presents algorithms that have some correct features. |
| **Design** and **evaluate** user experiences and algorithms. | User interface of game  Evaluation | The game is functional, accessible and usable. It has a clear, easy-to-use, well-designed user interface.  Evaluation of the functionality, usability and user interface is clear and reflective. | The game is functional, accessible and usable. It has a clear, easy-to-use user interface.  Evaluation of the functionality, usability and user interface is clear. | The game is functional, accessible and usable. It has a clear, easy-to-use user interface.  Description of the functionality, usability and user interface is clear. |
| **Design** and **implement** modular programs, including an object-oriented program, using algorithms and data structures involving modular functions that reflect the relationships of real-world data and data entities. | Completed game | The student demonstrates a range of control structures, and objects developed in Unity. Complex programming features are used within the game. | The student demonstrates a range of control structures, and objects developed in Unity. | The student attempts to demonstrate a sequence. |

Page 2 of 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Achievement standard criteria*** | ***E*** | ***D*** | ***C*** | ***B*** | ***A*** |
| ***Managing a digital project*** | *Poor management of project* | *Needed reminders to stay on track* | *Used class GANTT timeline to manage project* | *Adhered to class GANTT timeline to manage project* | *Managed project independently in school & home time* |
| ***Defined & decomposed the problem in terms of functional requirements & restraints*** | *Poor understanding of the problem* | *Needed assistance to define & decompose the problem* | *Defined & decomposed problem. Completed sample projects to help understand restraints.* | *Clearly defined & decomposed problem. Applied knowledge of function & restraints.* | *Excellent ability to define & decompose problem in terms of functional requirements & restraints.* |
| ***Designing:***  ***Design user experiences*** | *Unfinished/unclear, unworkable design* | *Design communicates intentions; shows some necessary components; some consideration of end user* | *Design clearly communicates intentions; shows necessary components; demonstrates function. Consideration given to end user* | *Design clearly & accurately communicates intentions; is annotated & detailed; shows necessary components; demonstrates function. Focus on needs of end user.* | *Design is unique & innovative; clearly & accurately communicates intentions; is detailed; shows necessary components; meets needs of variety of end users* |
| ***Assembling hardware*** | *Construction is incomplete and/or inaccurate. No understanding demonstrated.* | *Components are assembled following instructions with teacher assistance.*  *Limited under-standing of how components work in the circuit.* | *Components are assembled safely & correctly, following the diagrams, with assistance as required. Demonstration of basic understanding of circuits.* | *Components are assembled safely & correctly with understanding of polarity and circuits. Some assistance may be given* | *Components are assembled independently, safely & correctly demonstrating understanding of electronics.*  *Hardware allows for automation of heat pump.* |
| ***Programming:***  ***Design algorithms incorporating branching & iterations*** | *Programming is incomplete, little effort* | *Program is copied from sample code with little understanding* | *Program works & allows for data collection & presentation, includes branching & iterations. Basic understanding of code* | *Program is efficient & annotated allows for data collection & presentation, includes branching & iterations. Good understanding of code* | *Programming is elegant & efficient. Excellent understanding. Programming allows for automation of heat pump* |
| ***Modifying:***  ***Test & modify digital solutions*** | *Solution not tested* | *Solution tested, no consideration given to modifications* | *Solution tested & modified where required* | *Solution tested. Various modifications implemented.* | *Solution tested in different conditions. Modifications made to optimise performance & aesthetics.* |

Rubric for student self-assessment and teacher assessment, courtesy of Cindy Thornton (Flinders Island)

Page 3 of 3

Page 3 of 3