**Capstone Coding Project Grading Rubric**

**Complexity**

The Capstone Coding project’s complexity should reflect a minimum of **12 hours** of focused development.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **20** | **18** | **16** | **14** | **12** | **10** | **8** | **6** | **4** | **2** | **0** |
| 12 + Hours |  |  |  |  | ~6 Hours |  |  |  |  | Very Little Time |

**Style**

Variable names are meaningful and conventions (capitalization) have been followed. Code is well-organized through effective use of functions. Code is generally quite readable.

|  |  |  |  |
| --- | --- | --- | --- |
| **10** | **8** | **6** | **<= 5** |
| Consistent best-practices are used for variable and function naming. Code is readable and well-organized. | Coding style is generally good in the project, but improvements could be possible in either variable naming, use of functions, or organization of code. | Several issues with code style are evident. Code reuse could be improved by refining current functions or adding additional ones. | Code style needs significant rework to reach a satisfactory level of quality. |

**Inline Comments and Comment Header**

Capstone Coding Project should have a comment header at the top of the program (Name, Date, Project Title, and a short description of what the program is). Throughout the code, inline comments should be added to improve organization and readability of code.

|  |  |  |  |
| --- | --- | --- | --- |
| **10** | **8** | **6** | **<= 5** |
| Comment header and inline comments are thorough, and increase readability of your code. | Some small deficiencies in the comment header or inline comments. Header may be incomplete, or inline comments could be added to. | Several issues with comments. Header may be missing or inline comments may be too sparse. | Inline comments and header need significant improvements. Many elements are missing |

**Reliability**

Does the project work as intended? Are there any bugs or notable issues?

|  |  |  |  |
| --- | --- | --- | --- |
| **10** | **8** | **6** | **<= 5** |
| The program is reliable in operating as intended. There are no evident bugs. | The program generally runs correctly, but one or two bugs can be found while using it. | The program has noticeable bugs that occur during typical use. | The program has several issues that cause it to operate in an unreliable way. |

**Overall Impression and Features**

This project should demonstrate understanding of the skills you’ve developed over the semester, encapsulated in an interesting and polished project. Does the finished product feel like it is done? Objects must be incorporated in some way.

|  |  |  |  |
| --- | --- | --- | --- |
| **20** | **16** | **12** | **<= 10** |
| Great work! This project nicely showcases the skills you’ve developed in each of the course modules. Final product is very polished, and includes a meaningful incorporation of objects. | This project demonstrates competence in the different computing structures covered. Finished product is close, but isn’t as polished as it could have been. Usage of objects might be further expanded. | The Capstone Product has several areas or features that don’t feel fully complete. Additional refinement would strengthen the finished work. Usage of objects might be minimal or missing. | This project is not complete to the expected degree. Further extension could have better demonstrated skills developed over the semester. |