

Project Programming

Projects in this category are self-executing programs created using recognizable programming languages including, but not limited to, Java, Javascript, C++, Ruby, Objective-C, etc. All parts of the program must be the author's own design. Programs must be identifiable in one of the following categories:

1. Computer-aided instruction or educational/learning games.
2. Business or commercial applications.
3. Personal applications that, with minor alterations, could be marketed for larger commercial audiences.

Area	None	Partial	Mastery
Documentation - 10% Did student(s) include citations for sources and permissions for non-student produced materials?	None of the required documentation present.	Some or most required permissions present.	ALL required permissions present OR none needed.
Complete and Functional - 15% Did student(s) complete the entire project?	Project is incomplete, unedited, or the photograph is not an original student photograph.		Program functions as designed to serve its specific purpose. No bugs present in the demonstration of the program. Program is able to be compiled as a self-executing file.
Creativity - 20% Did student(s) use a higher level of creativity throughout the design process and oral presentation? (Nervousness should NOT count against the student)	Minimal levels of creativity shown in the project design and oral presentation.	Students display lower levels of creativity in the design process and/or oral presentation. (Nervousness should NOT count against the student)	Student displays a high level of creativity throughout the entire program design process. The oral presentation is unique, well-planned, and creative. (Nervousness should NOT count against the student)
Understanding - 25% Did student(s) demonstrate a solid understanding of the software for project development?	Student displayed little to no understanding of the software used.	Student used software to build the app that did not require an in-depth knowledge of programming skills required to build and implement the mobile app.	Mastery in the choice and use of programming language. Student is able to answer specific questions about their project and the methods used to create the program. Student displays mastery in understanding of the programming language used. Student has analyzed many alternate solutions and has chosen the most efficient. Student has included the reasons for the solution chosen.
Intended Purpose - 30% Did all elements of the project work together to serve the intended purpose?	No elements of the design fit the intended purpose of the project.		Program uses outstanding solution that is easy to understand and maintain. Use of comments throughout the code to aid in troubleshooting. Code is organized and does not repeat throughout the project.