## Rubric for CA2: Building a JS Game Engine

### Understanding and Usage of JavaScript Syntax (30%)

Excellent (21-30): Demonstrates a comprehensive understanding and flawless use of JavaScript syntax. The code is logically structured, efficient, readable, and free of syntax errors.

Very Good (17-20): Demonstrates a good understanding of JavaScript syntax. The code is well-structured and mostly error-free. Minor inefficiencies or misunderstandings do not substantially affect functionality.

Good (12-16): Demonstrates a satisfactory understanding of JavaScript syntax. The code is functional but may have occasional syntax errors or misunderstandings.

Inadequate (0-11): Demonstrates a poor understanding of JavaScript syntax. The code contains significant syntax errors and is inefficiently structured.

### Use of Source Control (10%)

Excellent (8-10): Uses source control proficiently. Commits are frequent, appropriately sized, and include clear, concise, and meaningful messages.

Very Good (5-6): Uses source control effectively. Commits are generally well-sized and include meaningful messages.

Good (4-5): Uses source control, but there may be issues with commit frequency, size, or messages. These do not significantly hinder development.

Inadequate (0-3): Has difficulty using source control effectively. Commits may be few, large, or include unclear messages.

### Video Explanation of Code and Understanding of 2D Game Engine Functionality (25%)

Excellent (18-25): Provides a thorough, clear, and articulate video explanation of their code and the 2D game engine functionality. Demonstrates a deep understanding of game development.

Very Good (14-17): Provides a good video explanation of their code and the 2D game engine functionality. Demonstrates a solid understanding of game development.

Good (10-13): Provides a satisfactory video explanation but may have areas that are unclear or incorrect. Understanding of game development may be somewhat limited.

Inadequate (0-9): Struggles to explain their code and the 2D game engine functionality effectively. Demonstrates a lack of understanding of game development.

### Code Structure and Organization (15%)

Excellent (11-15): The code is highly organized and well-structured. Proper segregation of responsibilities and effective documentation through comments.

Very Good (8-10): The code is organized and generally well-structured, with some minor issues. Comments are mostly helpful.

Good (6-7): The code is somewhat organized but may lack a clear structure. Some parts are not adequately commented.

Inadequate (0-5): The code lacks organization and structure. Comments are infrequent, unhelpful, or missing.

### Game Functionality and User Experience (20%)

Excellent (14-20): The game works flawlessly and provides an excellent user experience. All elements function as expected, and the game flow is smooth.

Very Good (11-13): The game works well and provides a good user experience. Most elements function as expected, but there might be minor issues.

Good (8-10): The game works satisfactorily but may have noticeable issues or bugs. The user experience is acceptable but could be improved.

Inadequate (0-7): The game has significant issues or bugs that affect its functionality and user experience. The game flow may be confusing or illogical.