CISC 3667: Game Design and Development

Final Project: Check-In #2

Assigned: Thursday, November 5

Due: Tuesday, November 10, 2026, **before class (11:00 AM)**

Submission: Individual Submission to Brightspace

Purpose

Throughout this semester, we have explored different approaches to digital game development, including JavaScript, Unity, and Godot. Through these lab assignments, you have had the opportunity to experiment with different development environments, implement gameplay mechanics, and explore how different tools support the game development process.

Now it is time to **select the primary development tool** you will use for your final project and establish a development plan.

For this check-in, you will evaluate your options, commit to a development approach, and outline how you intend to build your game. Your plan should identify the features you need to implement, the resources you will need, the order in which you will approach development, and any skills you may need to develop independently.

By the end of this check-in, you should have **a realistic roadmap that you can use to guide your work toward a complete, playable game**.

You are not expected to have completed a prototype yet. However, you should now be ready to begin developing your game independently.

What to Submit

Write a brief response to each section below. You may include sketches, diagrams, screenshots, or other visual references if they help explain your plans.

1. Finalize Your Game Concept

Revisit your first check-in and provide an updated description of your game.

Include:

- The game's genre, theme, or setting.
- The player's main goal.
- The core mechanic and gameplay loop.
- The experience you want players to have.
- Any changes you have made to your original concept and why.

Your concept does not need to be completely finalized, but you should be able to clearly explain what you intend to build.

2. Development Tool Selection

Choose the primary development tool you plan to use for your final project.

Examples include:

- JavaScript (HTML, CSS, and Canvas)
- Unity
- Godot
- Another game engine, framework, or development tool (with instructor approval, if necessary).

Explain your choice by addressing the following:

- Which tool have you selected, and what influenced your decision based on your experiences with the lab assignments?
- Why is this tool appropriate for the type of game you want to create?
- What features or capabilities does the tool provide that will help you develop your game?
- What challenges or limitations might you encounter when using this tool?

- What do you still need to learn or practice before you can comfortably develop your game?

Important: Choose a tool that supports your intended gameplay experience and that you can realistically learn and use within the remaining project timeline. You are not expected to use the most advanced tool; you are expected to choose one that makes sense for your project.

3. Development Requirements and Features

Break your game down into the features and systems you will need to implement.

Identify:

A. Core Features (Required)

List the features that must be implemented for your game to be considered playable.

Examples:

- Main menu or title screen
- Instructions screen explaining the objective and controls
- Player movement and input (if applicable)
- Core gameplay mechanics and interactions
- Collision detection (if applicable)
- Game rules and objectives
- Scoring or progress tracking (if applicable)
- Win/loss conditions or game-ending conditions

B. Additional Features (Optional)

Identify one or two features you would like to implement if time permits.

Examples:

- Multiple levels or difficulty settings
- Sound effects and background music
- Animations or visual effects
- Additional characters, obstacles, or collectibles
- High scores or saved progress

C. Assets and Resources

Identify the assets and resources you anticipate needing.

Examples:

- Player sprites or character models
- Backgrounds and environments
- UI elements, buttons, and icons
- Sound effects and music
- Fonts, textures, or other visual assets

Indicate whether you plan to create these assets yourself, use existing resources, or modify available assets.

4. Development Plan and Workflow

Describe how you intend to approach the development of your game.

Break the process into manageable steps. Consider the order in which you will implement and test the different features.

For example:

1. **Project Setup:** Create the project and organize the necessary files, scenes, or assets.
2. **Core Gameplay:** Implement the main player actions, mechanics, and rules.
3. **Game Systems:** Add scoring, progression, collisions, and win/loss conditions as needed.
4. **UI and Feedback:** Implement menus, instructions, visual feedback, and other interface elements.
5. **Testing and Refinement:** Playtest the game, identify problems, and make improvements.
6. **Finalization:** Complete any remaining features, prepare the final build, and test the submitted version.

You do not need to follow this exact sequence. Your development steps should reflect the needs of your own game. Consider **which features depend on others being completed first** and **prioritize your tasks accordingly**.

5. Challenges, Questions, and Next Steps

Think about what might make developing your game difficult.

Answer the following:

- What part of your game do you anticipate will be the most challenging to implement?
- Is there a particular programming concept, tool feature, or development skill you still need to learn or practice?
- What is one potential problem that could prevent you from completing your game on time? How could you address it?
- What part of your design or development plan are you least certain about? What would you want to test first?

Finally, list **two specific tasks** you plan to complete as your first steps toward developing your game.

Include any questions or areas where you would like instructor feedback.

Submission

Submit your completed Check-In #2 to Brightspace by **11:00 AM on Tuesday, November 10**.

This is an **individual final project**, so your submission should reflect your own game concept, development tool selection, and implementation plan.

Your development plan does not need to account for every detail or predict every challenge. You will likely make adjustments as you build and test your game. The goal is to establish a realistic direction and break your project into manageable steps so that you can begin development with a clear plan.