

Course Final Project: Make a Playable Game

Goal

Create a playable game in Unity and deploy your product (for grading).

Team Members

Your team will consist of 4-5 total members. The intent is to assign duties to each team member; however, those duties should not restrict your participation in other aspects of the project. ***Regardless, everyone should be testing.*** For clarity, you should decide on the group member roles: narrative, design, and programming.

Versioning

All members of the group must use a versioning system; you can sign up for a free [Unity Collaborate](#) account. If you are not familiar with this type of tool, it is time to become more comfortable; versioning is an important part of the development landscape in industry and academia.

Due to differing versions of Unity and differences in operating systems, it is strongly advised that all team members coordinate their IDE to minimize problems. To begin, your team should create a simple empty project, upload it to the repository and each team member should clone and execute the project.

Narrative

Your game requires a narrative component. The narrative can be born out in game or as an external document. Regardless, narrative should be reflected in the game and gameplay.

Please also see the section on Final Product Submission.

Game Mechanic

Regardless of the type of game you create, the game should have a core mechanic. You might arrive at this mechanic by considering some potential themes:

gravity, office, dreams, nightmares, vegetation, jokes (but not puns), swarms, fire, toys, ice, etc.

It may be easier to restrict yourselves to a central theme and develop the mechanic with that theme in mind.

Requirement: An Artificial Intelligence Component

By definition, this assignment is open to interpretation; it is truly your chance to be creative. However, your final game must include an implementation of some form of artificial intelligence.

There are many common examples for AI in a game; however, once again, be creative. Here are a few common and not so common examples.

- Define an AI bot to function as an enemy. The bot should evidence intelligence; that is, it should evidence decision making and *not* adhere to a predictable protocol (e.g., changing directions according to a random number generator).
- Define a sidekick AI bot: one that stays with the player and assists (or hinders) their play.
- For a word game, provide a non-random hint; this is more difficult than it seems.
- Some set of reactive behaviors of a bot when a player is within range. The bot then resets when the player moves out of range.
- Non-trivial pathfinding.
- A cost-based analysis where a bot selectively moves around a terrain (changing levels, jumping, takes non-obvious shortest paths, etc.).

Meetings and Milestones

For this project we will adhere to a strict schedule detailed below.

Date	Week	Goal	Description of Milestone(s)
Tuesday, March 16	1	Repository Formation	The programmers should set up a repository, assist the rest of the team in cloning the repository, and executing the shared project locally.
Thursday, March 18	1	Game Idea Formulation	Each group will submit two (2) high-level ideas for video games. These ideas should include a brief description of the overall game concept and possible ideas for the main mechanics. You are encouraged to develop a large set of ideas (as a group) and cull the list to two of your best ideas for submission.
Thursday, March 25	2	Game Idea Chosen	<p>The group will decide on one (1) of the eight (8) ideas formulated by the class. From this idea, you should individually complete the following tasks:</p> <p><i>Narrative:</i></p> <ul style="list-style-type: none"> • An initial idea for game narrative. • Meet with the designer to discuss assets. <p><i>Design:</i></p> <ul style="list-style-type: none"> • Identify assets related to the narrative idea and game idea. • A draft of a 'level' is constructed on paper or started in Unity. • Meet with the programmer to discuss inclusion of AI. <p><i>Programmer:</i></p> <ul style="list-style-type: none"> • Assisting the designer with constructing a level using placeholders (no assets). • Meet with the designer to discuss inclusion of AI.
Thursday, April 1	3	Game Mechanic Defined	<p>The main game mechanic is defined by the group; issues with implementation of the mechanic are strongly considered.</p> <p><i>Narrative:</i></p> <ul style="list-style-type: none"> • Building the game world. • An idea for explaining the main mechanic in the narrative. <p><i>Design:</i></p> <ul style="list-style-type: none"> • Continue level design with the mechanic in mind. • Finalize the inclusion of AI. <p><i>Programmer:</i></p> <ul style="list-style-type: none"> • Implementing the main mechanic. • Setting the groundwork for the AI.
Thursday, April 8	4	Game Building	All parties continue their tasks; progress over the previous week must be clear.
Thursday, April 15	5	Game Building	All parties continue their tasks; progress over the previous week must be clear.
Thursday, April 22	6	Testing	<p>All major source code components should be complete.</p> <p>All group members should be testing and refining the game.</p> <p>All groups members should be reviewing the narrative.</p>
Monday, April 26	7	Final Demo	<ul style="list-style-type: none"> • Final game demonstration to the entire class. • Submission of final documents.

Possible Ideas

- Extend one of the prototypes from the book. This extension must be significant and not just limited to the ideas expressed in the textbook (at the end of each chapter).
- Pinball games can be rich in subtlety and story while also having engaging gameplay. As an example, Google Addams Family or Star Trek: The Next Generation pinball games. A modified pinball game would also require relatively simple use of linear algebra, interpolation, and/or Bezier curves.
- Modify the notion of pinball to create a game where the goal is to place ball(s) into sequential drop points.
- Create a mimic of a game from the 1980s (Astrosmash, Moon Patrol, Pitfall, etc.) and extend the functionality.
- Take a current AAA game and scale it down to a prototype version; game narrative should be within the existing world, but unique in its story.



Prohibited Projects

- Simple extensions to lab prototypes.
- Using an existing game found anywhere online and modifying components unless approval is received from the instructor. Choosing this path will result in heavy scrutiny specifically with narrative.

Assets

- We encourage your group to explore the Unity asset store to make your game pop and have a unique flair. You may purchase assets, but it is **not** recommended.

Final Product Submission

- Please provide the instructor access to your repository. Your repository must contain well-commented and documented source code (see the rubric for more details).
- The executable will be provided to the instructor for review.
- All world-building, background components, and written elements must be submitted in the form of an old school *game insert*: a pamphlet stapled in the middle. This game insert must have cover art for the game; the game insert need not contain any art beyond the cover.
- Please submit a PDF of your game insert on Moodle.

As an optional submission document you may submit `review.pdf`. If you have any concerns about your group members and their level of participation, please feel free to submit a description detailing any issues. Reviews may be submitted up to 24 hours after the deadline of the project. ***If there are issues prior to submission, please make sure to inform your instructor as soon as possible.***