Project Planning Assignment
(50 pts)
A day: Due Thu, Mar 4th at *end* of class
B day: Due Fri, Mar 5th at *end* of class
For those working in teams, this is a team deliverable. Every member of the team should submit the identical file, and every member must submit a peer review to the Game Project Plan Peer Reviews assignment link in canvas.

Read the instructions carefully and submit this document.

Objectives:

- Organize Teams
- Create Project Design Document
- Create Initial Project Plan

Team Organization - Team size is limited to 3 people. You may form a team with someone in a different section of the class. *You do not have to be on a team*, but you cannot change your mind later; you must stick with your initial decision.

- List team members below:

Mahin Moon and Caleb Pollard

- Come up with a team name, and list it below:

No Error Rewards Dumbness (NERD)

Project Design Document (This is the biggest part of this assignment)

Go to Lab 1 in the Unity tutorial. Scroll down on that page and you will see a design doc you can download. You will complete the project design document discussed in this lab (*except for the milestone section, see below*). Watch videos 2-4 to see several examples of the completion of this document. The project examples he shows are more simple than what your project will involve, but most games can still be described in terms of the components of this document.

You will be graded on how well your design document communicates, so be clear. *Most answers should be more than one-word answers*. The point of the document is to give a summary of your game; if the sections in this design document aren't sufficient, then add additional sections as needed in order to completely describe your game. If it communicates better, you can also create a completely different document, as long as that document addresses at least the same issues as the provided design document. *You will be graded on how well your document communicates, so do what it takes to make it clear.*

In the "Other Features" portion of the document, include links to any game(s) that inspired your idea, or games that have some similar elements to things in your game. Even if the sites are just advertising the game, that is fine.

Some Conditions:

1. Your project should be sufficiently complicated that it uses a wide range of the things we have learned (plus some new things). I may ask you to revise the project if the scope looks too easy. It CANNOT be the same project you are doing for Mr. Wooden's class, and it CANNOT simply be working through a big tutorial.

2. If the scope is too big, we'll talk about it and probably narrow it down as we go. In general, you should aim high, and we'll adjust as needed, rather than aim low. This should be something you'll be proud of showing off to family and friends. If a college or potential employer asks for a portfolio of your work, this project should be in it!

3. This is a project for school (not your personal project). So, you need to use common sense and avoid anything offensive or inappropriate for a school context. No violent shooter games, blood, etc. If you wouldn't be proud to show the game to your grandmother, then don't do it. If you have questions about a particular element of your game, ask me and we'll work it out.

4. Do not use anything that you do not have permission to use (i.e., music, textures, etc. that are copyrighted). Even though this is just a school project, academic fair use of copyrighted material does not extend to using anything you want in school projects.

Copy/Paste the completed Design Document (or a link to a shared doc) below.
https://docs.google.com/document/d/1KyW7f_ay-uC_D_6uJPMxXt9XFzDF2Br0gzKLCLgwoYA/edit?usp=sharing

Project Schedule

You should IGNORE the Project Timeline portion of the Unity document. Instead, you will complete the Project Plan described below.

We are dividing the remainder of the semester into multiple "Sprints". A sprint is an Agile project management term; for our purposes, the important part of a sprint is that it ends in a *demonstration* of the work you have done up to that point.

Complete the table below, adding the required information for each task that you plan to complete by the end of the specified sprint. You'll need to add lots of room to the table; this should be a pretty detailed plan!

As you define the tasks, you should consider the following:

- A sprint should result in something demo-able and testable; there must be tangible work products to show.

- Task dependencies: think about what elements need to be in place before it makes sense to work on each task. Don't schedule a task at a point in the project where the task is untestable due to missing tasks that the task depends on.
- Estimated complexity/time of the task: Each task must have a time estimate (in hours).
    - No task should take more than a couple of hours of work; if you expect it to take more than 2-3 hours, then the task is too big for planning purposes and should be broken down into several smaller tasks.
    - Assume that each sprint will involve at least 8 hours of work per individual. You can obviously work more hours than that, but 8 hours is the minimum expected.
    - If the tasks for one person for the sprint add up to significantly more than 8 hours, then you need to re-think it and re-plan.
    - If the task is something you have no idea how to do (i.e., you'll have to do some research), factor that research time into your estimate.
    - Probably every one of your time estimates will be wrong, but the process of thinking through the level of detail to come up with an estimate is still important. It is OK to adjust a plan when you figure out an estimate is wrong, it is not OK to not have a plan.
- I would expect typically a minimum of 4 tasks per sprint.

Sprint	**Tasks Planned** This column has only the name of the task and the hours estimate for the task. The name is important because this is how we will reference the task in status reports.	**Description of Each Task:** Put the name of the task, plus: ● a brief description of the task ● The person assigned to the task (if on a team)
Sprint 1 (2 weeks + Spring Break) March 8th - March 26th	Asset gathering (1 - 2 hours) Basic code (3 - 4 hours) Basic world building (1 hour) Minor playtest (1 hour)	- Look for assets we can use or make our own. - Setup a game manager and a player controller. - Make the basic layout for the game - Test what we have so far
Sprint 2 (2 weeks) March 29th - April 8th	Enemy Creation (2 hours) Basic attack (2 hours) Enemy follow (2 hours) Enemy damage (2 hours) Playtest (1 hour)	- Create the enemy through assets - Make and attack animation that looks passable. - Make a script to make the enemy follow the player - Make a health system and successfully deal damage - Player is able to defeat enemies and enemies are able to cause Game over
Sprint 3 (2 weeks) April 12th - April 23rd	Create different weapons (4 hours) HUD (2 hours)	- Find assets or use what we found to make weapons - Make the heads up display to show

	Menu (1 hour) Upgrade (3 - 4 hours) Playtest (1 hour)	health and other information - Make a working menu to adjust difficulty - Make a shop in the menu to upgrade various things ingame. - Playtest will include having a working menu + upgrade system
Sprint 4 (2 weeks) April 26 - May 7	Assigning assets (2-3 hours) Making an actual world (3-5 hours) Finalizing enemy design (1-2 hours) (Hopefully) adding physics to projectiles Playtest (1 hour)	- Getting all assets to the place they need to be - Make the world look better by adding textures - Making more enemies and possibly bosses - Add physics to the combat system if time allows. - Assets work with our code and the world looks good
Sprint 5 (2 weeks) May 10 - May 21	Cleaning up the game (4 hours) Bug testing and fixing (4 hours)	- Finalizing everything, debugging, double checking everything and focusing on finishing anything not done.

Extra tasks/features you don't think will fit into the above schedule go here. This will be a nice-to-have list beyond the MVP. Possibly adding boss waves and endless mode