**Retro Game Jam GDD Template**

**Game Title:**

*Enter your game’s name here.*

**Team Name:**

*Choose a team name that represents your group.*

**Team Members:**

*List all members and their student IDs:*

* Name (Student ID)
* Name (Student ID)
* Name (Student ID)
* Name (Student ID)

**Core Concept**

**Briefly describe the game.**

* What classic retro game are you taking inspiration from? (**You can not do** [**Breakout**](https://www.youtube.com/watch?v=AMUv8KvVt08) **since it will be a topic for a future studio assignment!)**
* What unique twist are you adding to make it different? (e.g., new mechanics, modernized visuals, additional features)

Example: *Our game is inspired by Breakout, but instead of a paddle at the bottom, the player controls a magnet that can attract or repel the ball, adding a physics-based challenge.*

**Core Gameplay**

**Game Loop:**

***Describe the basic actions the player takes and how they interact with the game.***

* What is the player doing repeatedly? (e.g., jumping, shooting, dodging, collecting items)
* How does the game provide feedback? (e.g., visual effects, sound effects, UI updates)

Example:  
*The player moves a spaceship left and right to avoid asteroids and shoot enemies. Destroying enemies grants points and occasional power-ups, making the ship stronger.*

**Player Controls**

***List the controls clearly and simply.***

Example:

* Keyboard (or Controller) Inputs:
  + Move: Arrow Keys / A & D
  + Jump: Spacebar
  + Attack/Shoot: Left Mouse Button / Ctrl
  + Special Ability: Shift / Right Mouse Button

**Level & Progression**

Game Structure:

* How many levels are in the game? (e.g., 5 levels, 10 waves, endless mode)
* Are levels handcrafted or procedurally generated?

Progression System:

* How does difficulty increase? (e.g., faster enemies, more obstacles, limited time)
* Are there power-ups or upgrades? (e.g., speed boost, double jump, new weapons)

**Example:  
*Each level introduces new enemy types and increases their speed. The player can collect shield power-ups for temporary invincibility. There will be 3 levels in total and the player has to win in all 3 to finish the game***

**Scoring & Win/Loss Conditions**

Winning:

* How does a player complete the game? (e.g., reach the final level, defeat a boss, survive a time limit)

Losing:

* What causes a game over? (e.g., losing all lives, running out of time, missing too many targets)

Score System:

* How is progress tracked? (e.g., points, time, collected items)
* Are there multipliers, streaks, or bonuses?

Example:  
*The player earns points for defeating enemies. A combo multiplier increases score for consecutive hits. The game ends when the player loses all three lives.*

**Timeline & Milestones**

**Week 1: Core Mechanics & Gameplay Elements**

***Example:***

* Set up a basic Unity project <Name (of student assigned to the task)>
* Implement player input and core mechanics (moving, jumping, shooting, etc.) <Name>
* Implement scoring system <Name>
* Create a rough UI (score display, health, etc.) <Name>

====

**Week 2: Polish & Finalization**

***Example:***

* Design and implement levels/waves <Name>
* Add particle effects, sound effects, and other polish <Name>
* Improve UI and menus <Name>
* Package the final build <Name>

**Assets**

**Models & Art:**

***Provide links or sources for sprites, models, and animations.***

* Custom-made or downloaded (e.g., OpenGameArt, Kenney Assets)

**Sound & Music:**

***Provide links or sources for music and sound effects.***

* Royalty-free or original compositions (e.g., Freesound, BFXR, Chiptone)

**UI:**

*Provide links or details on icons, fonts etc.*

**Notes: (Remove from your submission!)**

* Remember, this is (most likely) your attempt at making a full game! Keep the scope limited and manageable! A small, completed game is infinitely better than a large unfinished game (not to mention it’s a major chunk of your final grade). A good polished complete submission should be the goal.
* What you list can be open to change and modification down the line. But the earlier you get thinking about the higher-level concepts you would like to design for, the longer time you will have to prototype, test and iterate on it.
* You can pick a different game and a different twist mechanic from the ones listed in the assignment, but remember, you (as a team) are responsible for what you sign up for.
* Finally, your game can be fully 2D, but we will majorly be covering concepts only for 3D development in this course. Most concepts translate very easily from 3D to 2D and vice versa, but there might be some critical differences that you would have to look up.