| Project Design Document | | *mm/dd/yyyy*  Student Name | | --- | |
| --- | --- | --- |

**VIDEOLLAMADAS:** [**https://meet.google.com/oem-aeij-dsg**](https://meet.google.com/oem-aeij-dsg)

| Project Concept | | | |
| --- | --- | --- | --- |
| 1 **Player Control** |  | You control a   | spaceship | | --- | | in this   | front view | game | | --- | --- | |
|  | where   | *Arrow Keys*  *Space (optional)* | | --- | | makes the player   | *Move (Left, Right, Up, Down)*  *Shoot* | | --- | |

| 2 **Basic**  **Gameplay** |  | During the game,   | *meteorites/planets/spaceship enemies* | appear | | --- | --- | | from   | *towards us (z-coordinate; x, y fixed) or stationary* | | --- | |
| --- | --- | --- | --- | --- | --- | --- |
|  | and the goal of the game is to   | *get from start to finish point, avoiding the enemies (limited number of lives)* | | --- | | |

| 3 **Sound**  **& Effects** |  | There will be sound effects   | *when the spaceship collides with an enemy / when it grabs a power-up* | | --- | | and particle effects   | *small particle explosion when the spaceship collides with an enemy / game over* | | --- | |
| --- | --- | --- | --- | --- | --- |
|  | [*optional*] There will also be   | *background music* | | --- | | |

| 4 **Gameplay**  **Mechanics** |  | As the game progresses,   | *more and more enemies spawn* | | --- | | making it   | *more difficult to reach the goal* | | --- | |
| --- | --- | --- | --- | --- | --- |
|  | [*optional*] There will also be   | *power-ups that increase lives* | | --- | | |

| 5 **User**  **Interface** |  | The   | *lives* | | --- | | will   | *decrease* | | --- | | whenever   | *the spaceship collides with an enemy* | | --- | |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | At the start of the game, the title   | *DODGING IN SPACE* | will appear | | --- | --- | | | and the game will end when   | *the lives reach 0 OR when the spaceship reaches the finish point* | | --- | |

| 6 **Other Features** |  | | * *Score* * *Levels (easy - medium - hard)* | | --- | |
| --- | --- | --- | --- |

# 

# Project Structure

**Assets:**

- Audio

- Materials

- Models (por ahora vacío)

- Prefabs:

- Characters (player y enemies)

- Pickups (por ahora el pickup de la vida)

- Managers (por ahora solo el GameManager)

- Scenes:

*\* Landing/MainMenu*

*\* Game/Level 1*

- Scripts:

- Core

**\*** *GameManager.cs*

- Control

*\* EnemyController.cs*

*\* PlayerController.cs*

*\* PickupController.cs* (por ahora solo controla el pickup de la vida)

- UI

\* *LandingController/MainMenuController.cs*

*\* HUDController.cs*

*\* GameOverController.cs* (controla el UI del canvas cuando pierde)

*\* GameEndController.cs* (controla el UI del canvas cuando gana)

- UI:

- Fonts

- Images

# Project Timeline

| Milestone | Description | Due |
| --- | --- | --- |
| **#1** | | * *Hierarchy and folder structure and repository creation* | | --- | | | *3/2 - 7/2* | | --- | |
| **#2** | | * *Franco: PlayerController, agregar Debug.Log a objetos que colisione, probar cosas con el Test, agregar un limite izquierdo y derecho para que la nave no pueda desplazarse más allá de estos límites.* * *Marina: EnemyController, agregar cosas con el ProBuilder,* | | --- | | | *7/2 - 11/2* | | --- | |
| **#3** | | *- Marina: generar enemigos (spawn) con corrutina. Agregar más cosas a Enemy: Distintos movimientos (ej: persigue al player) y velocidad según dificultad. Varios tipos de enemigos (estáticos/no estáticos).*  *- Franco: Main menu, powerups* | | --- | | | *14/2 - 18/2* | | --- | |
| **#4** | | * *Marina: (AGREGAR UN BRANCH) spawn rate del enemy (cálculo de enemigos), efecto del choque del player con el static (rebote y desaparición): se puede agregar un physics material; → OK* * *Franco: (AGREGAR UN BRANCH) modificar límites del mapa desde el código en vez de usar un “túnel”: rotation queda igual pero la position cambia solo la coordenada en particular; cuando llega al exit, mostrar que ganamos; agregar boton de MENU en el HUD, modificar el tamaño del panel del main menu → OK* | | --- | | | *21/2 - 25/2* | | --- | |
| **#5** | | * ***Marina:*** * *persistencia de datos con json y/o PlayerPrefs dependiendo de la seguridad (nombre del jugador, score, color de la pantalla, etc), agregar inputfield para el nombre del jugador; el highscore y el player dependerán del nivel jugado (en el main menu mostrar un input field para poder ingresar el nombre y en el HUD mostrar nombre y high score del nivel que estamos jugando).* * *Bug: al pasar de un nivel a otro, me muestra el high score anterior (antes de ser pisado)* * *Parte estética: Particle effects, Skybox, Luces, Camera, Sonidos, Background music, assets (importación / probuilder)* * *Agregar botón Quit en el main menu para salir del juego → OK* * *Al hacer click en botón Menú, al volver a main menu mostrar la dificultad seleccionada anteriormente. Ahora se muestra siempre Easy. →OK* * *Cambiar la vista de la cantidad de vidas (ponerle un icono de diamante/corazón X el número de vidas), y colocar +1 arriba del diamante.* * *Sound effects: explosions, pickup, etc.* * *Background music, agregar botón para cancelar el sonido.* * Revisar la posición de spawn enemy. Hacer que se generen enemigos más hacia abajo. Si siempre voy derecho desde abajo, gano el juego sin hacer nada. Ver quizás valores de visibility distance. * Generar enemies (static y non static) con otros colores. * Guardar selección de audio. * ***Franco:*** * *Rotation del player cuando se mueve (con lerp y lookrotation)→ OK* * *implementar el OnCollisionEnter desde la clase base (EnemyController)* * *Game Over y Win: hacer que se detenga todo. (FALTA HACERLO CUANDO GANO: SOLO SE DESHABILITA EL PLAYERCONTROLLER)* * *Disparo de proyectil para el player (eso impactará en el score): tecla del teclado.* * *Agregar boton de PAUSA Y REANUDAR (tecla “P”)* * *Cuando gana, sumar una cantidad X de puntos* * *Poner un spawn point para el proyectil* * *Informar al player con un texto cuando haya superado el high score en el juego* * *Limitar frecuencia de los disparos: por ej q tengas que esperar 1 segundo entre disparo y disparo* * *Agregar botón de PLAY AGAIN cuando ganas o perdes (para jugar al mismo nivel)* * *Corregir rotacion player con LERP para que vuelva a la rotacion default (0,0,0) de forma “suavizada”; intentar con* ***FromToRotation*** * *Ver por qué no rota con velocidad distinta de 2 (verlo a lo último)*   ***FALTA:***   * Revisar la cantidad de vidas que aparecen durante la jugada o inicialmente. Para el nivel easy puede que sean demasiadas. * Cuando va horizontalmente, luego verticalmente y luego horizontalmente, sigue rotado hacia el sentido contrario. No pasa cuando va verticalmente, luego horizontalmente y luego verticalmente. * Escribir readme.txt * Limpiar código * Hay enemies que se generan fuera de los límites del mapa * Borrar highscores y users al hacer build del juego | | --- | | |  | | --- | |
| **Backlog** | | * *Feature on backlog - not a part of the minimum viable product* * *Feature on backlog - not a part of the minimum viable product* * *Feature on backlog - not a part of the minimum viable product* | | --- | | | *mm/dd* | | --- | |

# 

# Bugs

# Project Sketch

