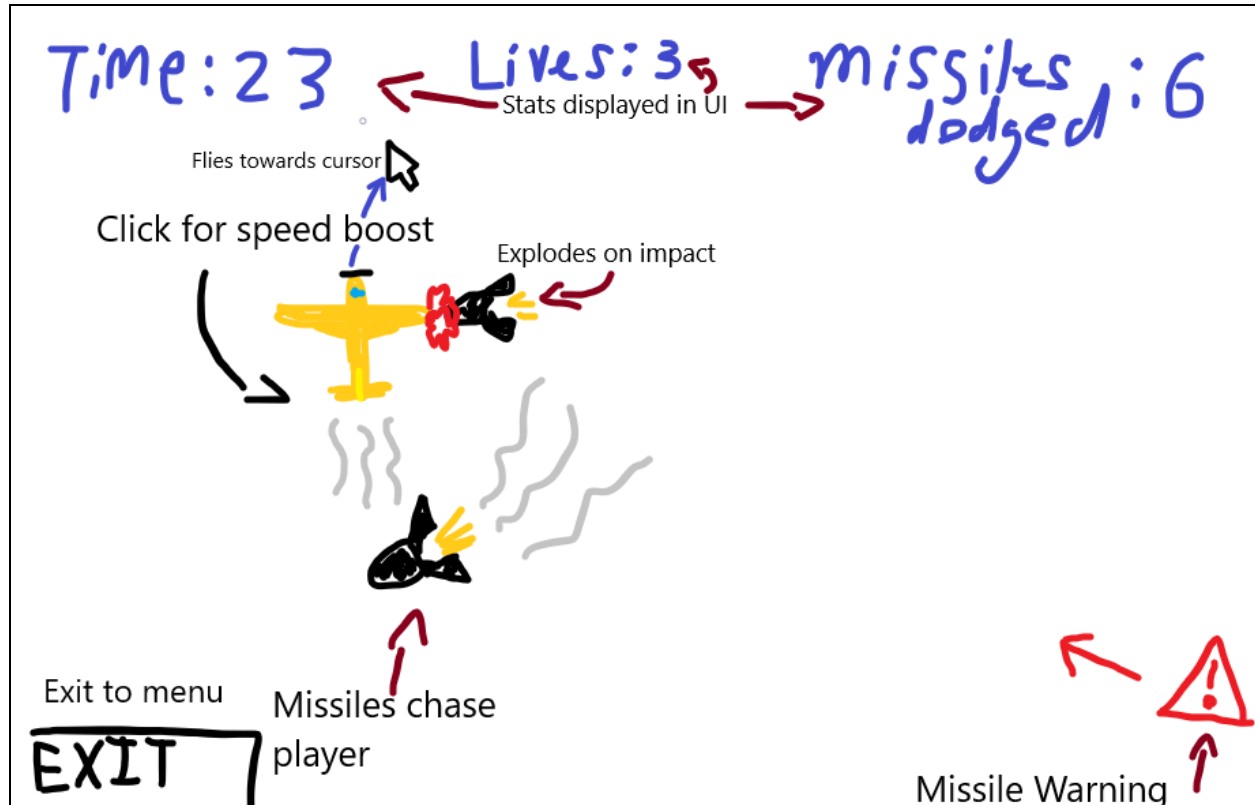


Missile Dodge

Evan Daveikis 991 721 245



Plane moves towards cursor

Speed boost when clicking the mouse

Missiles give a warning and then chase player, exploding on impact or after a few seconds

Score is displayed

The player dies after a few lives are lost

See Assignment/Planning/Planning.txt for more details

Jet.cs (player)

- Goes towards cursor location
- Has a temporary speed boost on click
- Turns to face the cursor and flies forward
- Has limited number of lives
- By nature of following mouse will never fly too far out of bounds
- Rigidbody2D (kinematic) for movement and collisions
- Some kind of 2D collider (maybe capsule)
- Animator for plane animations (maybe wiggles wings when turning)
- Pseudocode:
 - Variables for speed, angularSpeed, boostTime, boostCurve, boostCooldown, boostTimer,
 - (cont'd) rigidbody, targetRotation, animator
 - Gather references to components in Start
 - Update target rotation and animator in Update
 - (cont'd) Check for boost in Update and update HUD
 - Rotate and move in FixedUpdate
 - Let Manager know when this is destroyed

Missile.cs

- Spawned outside of screen
- Tracks towards where plane will be (predicts position)
- Turns slowly (can be out-maneuvered)
- Gains speed over time before exploding after some time
- Spawns explosion effect when blowing up (due to plane or time fuse)
- Destroys plane (and self) on collision
- Rigidbody2D for movement
- Animator for smoke and boost animations
- Collider used to detect crash
- Pseudocode:

- Variables for speed, angularSpeed, velocityCurve, maxLifetime, lifetime,
- (cont'd) rigidbody, targetRotation, plane, animator, explosionPrefab (overall very similar to plane)
- ^^^ Maybe missile and plane will inherit from common base class (or composition smth)
- Gather references to components and plane in Start
- (cont'd) Update 'missiles spawned' count of

HUD/Manager

- Predict plane position and update target rotation in Update
- (con't) Increment lifetime and update speeds accordingly (Destroy if max lifetime reached)
- Rotate and move in FixedUpdate
- Destroy self and plane in OnTriggerEnter2D with plane
- fn Explode() -> spawns explosion particles and destroys self, updates missiles dodged score

MissileSpawner.cs

- Activates a missile warning and spawns a missile every few seconds (randomized)
- Stores a list of spawn areas (boxes) and spawns the missile in a random area of a random box
- No extra components (empty gameobject)
- Pseudocode:
 - Variables for missileSpawnTimeRange, spawnTimer, spawnAreas, missileWarningPrefab
 - Update timer and spawn missile warning if necessary in Update (see below)
 - fn SpawnMissileWarning() ->
 - Pick random spawn area
 - Pick random position from selected area (will be inside screen)
 - Push position backwards to find missile spawn position
 - Spawn missile warning at position (in screen), passing it the missile position
 - Missile warning handles spawn of missile itself

MissileSpawnArea.cs

- Stores where missile warnings and missiles can be spawned
- Just a vector2 for the area and a float for how far back missiles should be spawned
- Pseudocode
 - Variables for areaSize, missilePushback
 - fn GetRandomPosition() -> returns random point in area (will be onscreen)
 - fn GetMissilePosition(spawnPos) -> pushes spawnPos back by missilePushback (will be offscreen)
 - Draw helpful visuals in OnDrawGizmos

MissileWarning.cs

- Spawned by MissileSpawner
- Blinks for a few seconds to warn player of incoming missile
- Spawns missile after said few seconds
- Animator for blinking animation
- Pseudocode:
 - Variables for animator, missilePrefab, warningTime, timer, missileSpawnPos
 - Gather references to animator and set timer in Start (may not even be needed really)
 - Decrement timer in update and spawn missile/destroy self if necessary
 - fn Spawn() -> does the above (spawns missile and destroys self)
 - fn Init(spawnPos) -> sets the missile spawn position

SceneButton.cs

- Just contains functions used by buttons to load scenes
- Used in main menu and game
- Pseudocode:
 - Variable for scene name
 - fn Load() -> loads the desired scene

HUD.cs (heads-up display, score and such)

- Manages the UI like score, lives, etc
- No components in and of itself
- Pseudocode:
 - Variables for livesText, scoreText, highScoreText, missilesDodgedText etc, playerRespawnText, singleton
 - Stores singleton in Awake
 - static fns for SetScore, SetMissileDodges, SetLives, etc which update UI elements

Score.cs

- Saves and loads high scores upon start and end of scene
- Functions for updating scores which updates the relevant UI
- Pseudocode:
 - Variables for various scores (see above), singleton
 - Stores singleton in Awake
 - Loads any saved values in Start and updates HUD
 - Saves values in OnDestroy
 - static fns for IncreaseScore, OnLifeLost, OnMissileDodged, etc

MissileDodgeManager.cs (various housekeeping)

- Respawns player, forces game end, manages game flow
- Pseudocode:
 - Variables for playerPrefab, lives, respawnTime, respawnTimer, isPlaneAlive
 - fn OnPlaneDestroyed() -> sets timer and isPlaneAlive bool, decreases lives
 - Update respawn timer and corresponding text in Update
 - (cont'd) respawn plane near middle of world when timer has elapsed and lives > 0
 - Show game over text and force exit to main menu if lives <= 0