



forest escape



Forest Escape - Joc în Unity

Descriere joc

Jocul implementat de către noi în **Unity** este un joc de tip runner în care trebuie să ajutați o vulpe să adune hrană din pădure și să se întoarcă acasă la puii ei. Vulpea va alerga la o viteză aproximativ constantă de-a lungul jocului pe un traseu bine definit, încercând să se ferească de diverse obstacole și să deblocheze niveluri, totul în timp ce încearcă să obțină un scor cât mai mare și să adune hrană. Am optat pentru versiunea de **level-based runner** și nu endless runner, construind astfel 3 niveluri distincte.

Features

Jocul realizat de noi dispune de următoarele caracteristici:

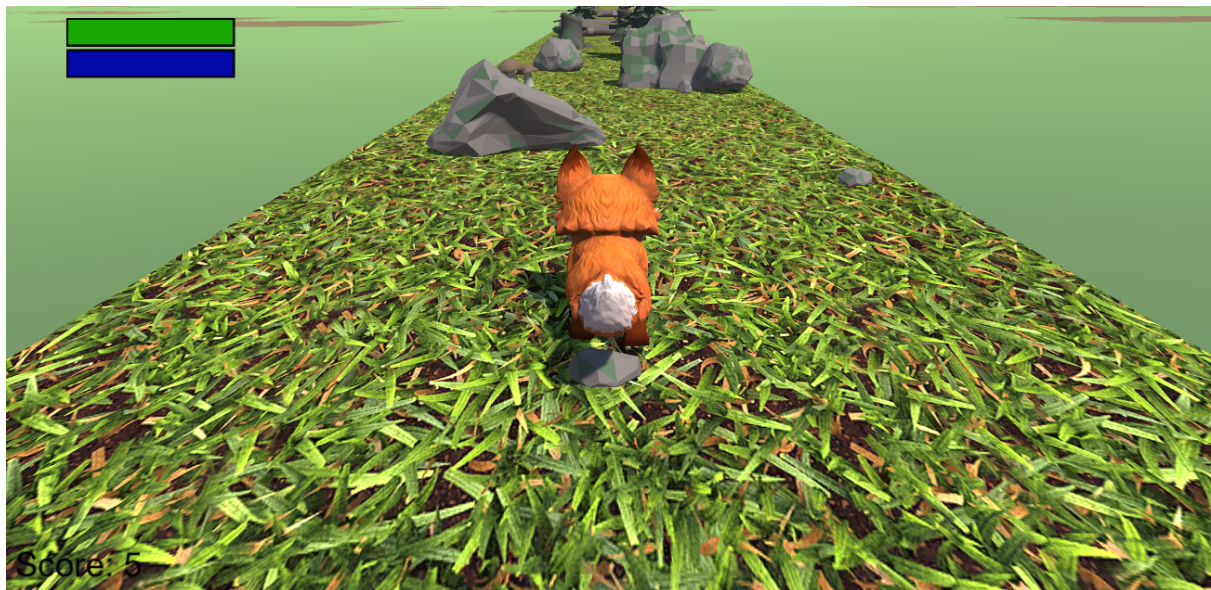
1. Crearea player-ului și a pistei de alergat cu cele 3 benzi: stânga, mijloc și dreapta.
2. Posibilitatea de a muta vulpea de pe o banda pe alta prin click sau cu ajutorul săgeților sau prin apăsarea tastelor **A** și **D**.
3. Realizarea **obstacolelor** de tipul: friendly, default (lovire). Printre cele default de care trebuie să se ferească se numără: obstacole statice care rămân fixe, spinner-ere, obstacole care cad din cer, obstacole care se mișcă spre vulpe și obstacole peste care este nevoie să sară.
4. Realizarea **finish line**-ului.
5. Interacțiunea dintre jucător și obstacole.
6. Existența unui **health bar** care să contorizeze loviturile pe care le primește vulpea: în momentul în care se ciocnește de un obstacol, health bar-ul va scade.
7. Existența unui **mana bar** care dă posibilitatea folosirii imunității.
8. Existența unor obiecte (**items**) ce pot fi colectate pentru a crește mana (ciuperci).
9. Jucătorul poate opri jocul prin apăsarea tastei **P** (pause) și apoi se poate întoarce la joc.
10. Jucătorul poate salva jocul prin apăsarea tastei **R** (save) și poate încărca jocul prin apăsarea tastei **Y** (load).
11. **Persistența datelor**: jucătorul poate da save la joc cu ajutorul tastelor, salvarea obiectelor și a poziției player-ului este făcută în JSON.

12. **Meniul principal** unde jucătorul poate da pauză la joc, poate începe un joc nou și poate vedea un leaderboard.
13. Existența unui **countdown** când revine jucătorul în joc.
14. **Generarea nivelelor** pornind de la primul (cu ajutorul Scene Template).
15. **Îngreunarea nivelelor**: mai multe obstacole, viteza cu care vulpea aleargă este crescută.
16. **Încărcarea nivelelor**: odată ce vulpea ajunge la linia de finish a unui nivel, se încarcă următorul nivel.
17. **Dezallocarea nivelului** terminat.
18. Lane-ul pe care aleargă vulpea se vede continuu în față.
19. Jucătorul poate să își introducă numele la începutul jocului într-un **textbox** și acesta este salvat împreună cu **scorul** maxim obținut.
20. Vulpea poate sări peste obstacole prin apăsarea tastei **Space**.
21. Existența unui **scut** (invincibilitate) care poate proteja vulpea de impactul cu anumite obiecte pentru câteva secunde prin apăsarea tastei **F**.
22. **Încărcarea asset-urilor** pentru fiecare nivel, stilizarea acestora, plasarea acestora în scenă și transferarea funcționalităților de la obiectele inițiale (toate obiectele).
23. **Animatie obiecte pickables**.
24. **Încărcarea asset-ului player-ului** (vulpea) și funcționalitățile acestuia.
25. **Dezactivarea obstacolelor** care se află la un range mare față de player, atât în față, cât și în spate.
26. Dropper-ul își **setează distanța** de la care cade în funcție de viteza player-ului.
27. **UI Game Over**.
28. **Audio** pe parcursul jocului.
29. Playerul poate să își salveze progresul atunci când apasă **Quit** și îl poate relua apăsând **Resume**.
30. Calcularea **scorului** (păstrarea valorii maxime atunci când playerul este împins în spate de un obstacol).
31. Crearea unui capsule collider pentru player și **combinarea animațiilor** acestuia cu mișcările de bază.
32. Documentație.
33. Îmbinarea taskurilor (**version control**).
34. Crearea unui **leaderboard** cu date stocate în json.
35. Player-ul poate da **boost** prin apăsarea tastei Shift cu cooldown de 5 sec.

36. Pentru scenele de dinaintea adăugării asset-urilor din unity (GeneratedScene) s-a folosit **object pooling** și generarea de obstacole random, persistența datelor (player și obstacole generate) când se dă load. Obiectele inactive sunt reciclate, dacă facem request de object și nu avem ce recicla, se alocă unul nou. Pentru scenele finale (TotalScene, TotalScene1, TotalScene2) nu s-a reușit integrarea.

Shortcuts:

R - save
Y - load
Space - jump
P - pause
A - left
D - right
Shift - boost
F - immunity



Teamwork

Această echipă este formată din următorii studenți:

- Popa Iulia Andreea, 341
- Dorneanu Alina-Mihaela, 341
- Bădescu George Gabriel, 341
- Bara Tudor, 341

- Sandu Răzvan, 341

		Andreea	Alina	Gabriel	Tudor	Răzvan
Funcționalitate	1	x		x		
	2	x		x		
	3			x		
	4		x			
	5	x		x		
	6	x				
	7				x	x
	8					x
	9	x				
	10				x	
	11				x	
	12	x			x	
	13	x				
	14		x			
	15		x		x	
	16		x			
	17		x			
	18		x			
	19	x				
	20			x		
	21				x	x
	22		x			
	23				x	
	24			x		

	25				x	
	26				x	
	27	x				
	28		x			
	29	x				
	30	x				
	31			x		
	32		x			
	33				x	
	34	x			x	
	35					x
	36				x	

Inspirație

Pentru realizarea jocului ne-am inspirat din următoarele jocuri: Subway Surfers, Temple Run.

Resurse

Repository-ul de **Github** asociat acestui proiect:

<https://github.com/andreea-popa22/Unity-Project>

Bibliografie

1. <https://www.udemy.com/course/unitycourse2/>
2. <https://forum.unity.com/>
3. <https://docs.unity3d.com/Manual/index.html>
4. https://www.youtube.com/watch?v=dA4oOm3wClc&ab_channel=N3KEN
5. https://www.youtube.com/watch?v=ulxXGht5D2U&ab_channel=TurboMakesGames

