

Prototip borbe 2D videoigre u Godotu

Računalna animacija: Fran Jelavić, 0036523987

Sažetak

Prototip borbe implementiran je u Godot-u u sklopu 2D igre, s fokusom na stvaranju funkcionalnog i zabavnog sustava borbe. Istražuju se koncepti zabavnih mehanika borbe u videoigrama te kako te mehanike surađuju ili funkcioniraju jedna s drugom.

Vizualna tematika prototipa nastoji prikazati glavnog lika, inspiriranog Aragornom iz Gospodara prstenova, kako se sukobljava s neprijateljskim Uruk-hai-ima u bliskoj borbi tijekom Bitke kod Helmve klisure.

Abstract

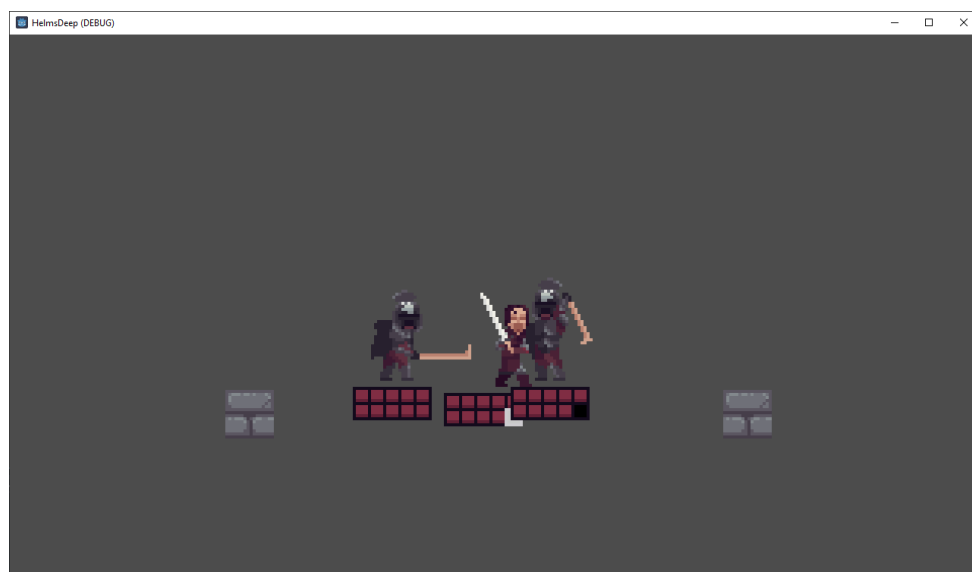
The combat prototype was implemented in Godot as part of a 2D game, with a focus on creating a functional and fun combat system. It explores concepts of fun combat mechanics in video games and how these mechanics cooperate or work with one another.

The visual theme of the prototype aims to show the main character, inspired by Aragorn from The Lord of the Rings, confronting enemy Uruk-hai in close combat during the Battle of Helm's Deep.

Pokretanje Godot Build-a

Ovaj projekt ne uključuje izveznu datoteku, ali moguće je pokrenuti Godot *Build*.

1. Pokrenite Godot Engine verzije 4.2.
2. Učitajte projekt pritiskom na tipku Load na početnom zaslonu te odaberite direktorij gdje ste smjestili preuzeti projekt.
3. U pregledniku projekta kliknite na ikonu *Play* u gornjem desnom vrhu prozora.
4. Igrajte prototip.
5. Zatvorite prozor pritiskom tipke F8 na tipkovnici.



Upravljanje

Kretanje lika

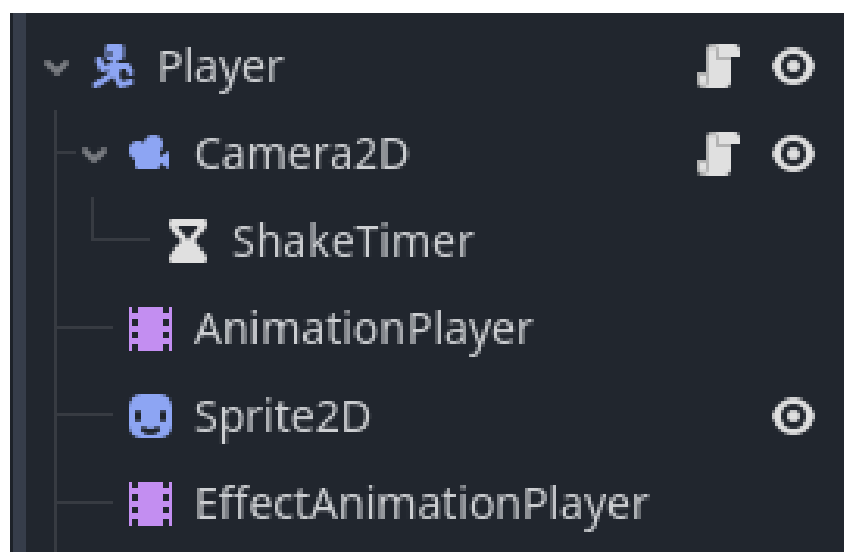
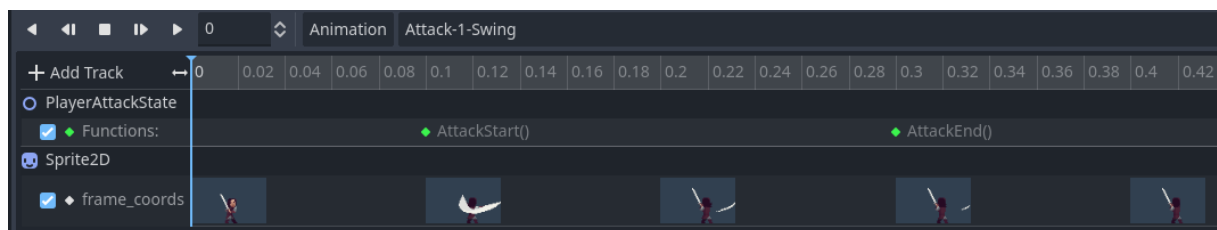
- Hodanje i okret ulijevo ili udesno igrač postiže tipkama A i D.
- Držanje tipke Shift tijekom kretanja lika, lik će krenuti trčati.

Kontrola napada lika

- Lijeva tipka miša pokreće napade. Konkretnije, puštanjem tipke počinje izvršavanje napada.
- Lik napada samo u smjeru kojem je okrenut (lijevo ili desno).
- Kretanje je usporeno tijekom napada te nije moguće promijeniti smjer usred izvršenja napada.
- Tri vrste napada:
 - Brzi napad – instantni pritisak tipke
 - Punjeni napad – držanje tipke, ali ne značajno dugo
 - Teški punjeni napad – držanje tipke značajno dugo
- Šteta i sila odbacivanja variraju od najniže (brzi napad) do najviše (teški punjeni napad) razine.

Animacije i vizualni elementi

Vizualni elementi su vlastiti rad, nacrtani i animirani pomoću programa Aseprite, te zatim složeni u funkcionalnu animaciju u Godotu.



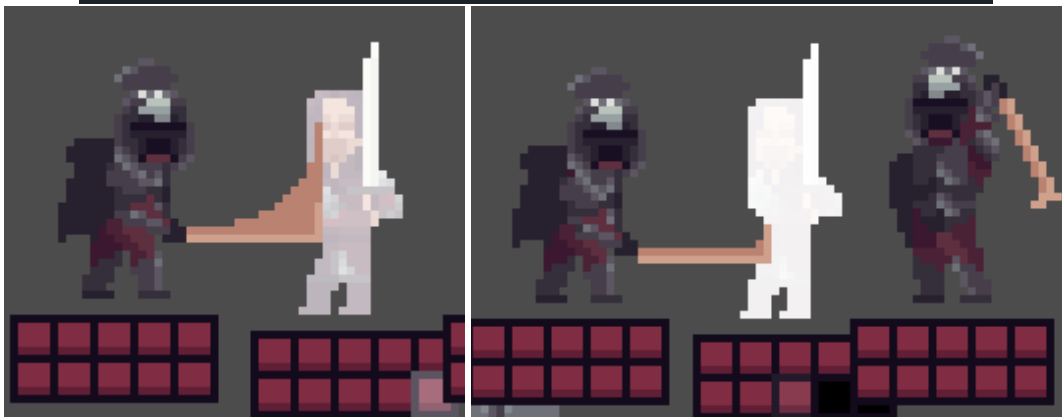
Sprite i Animacija

Igrač i neprijatelj imaju Godot čvor (*node*) *Sprite2D* za slikovnu reprezentaciju koja je animirana čvorom *AnimationPlayer*. *EffectAnimationPlayer* služi isključivo za prikaz bljeska na liku tijekom primanja udarca.

Bljesak pri udarcu

Bljesak primanja udarca postignut je pomoću sjenčara za promjenu boje čvora *Sprite2D*. Kroz animaciju u *EffectAnimationPlayer*, iz okvira u okvir mijenja se vrijednost uniformne varijable iz 0 u 1 i nazad. Vrijednost 1 mijenja neprozirne piksele u bijelo, dok vrijednost 0 zadržava izvornu boju piksela.

```
uniform float flash_intensity : hint_range(0.0, 1.0);  
void fragment() {  
    >| // Called for every pixel the material is visible on.  
    >| vec4 input_color = texture(TEXTURE, UV);  
    >| vec4 new_color = vec4(1.0, 1.0, 1.0, input_color.a);  
    >| COLOR = mix(input_color, new_color, flash_intensity);  
    }  
}
```



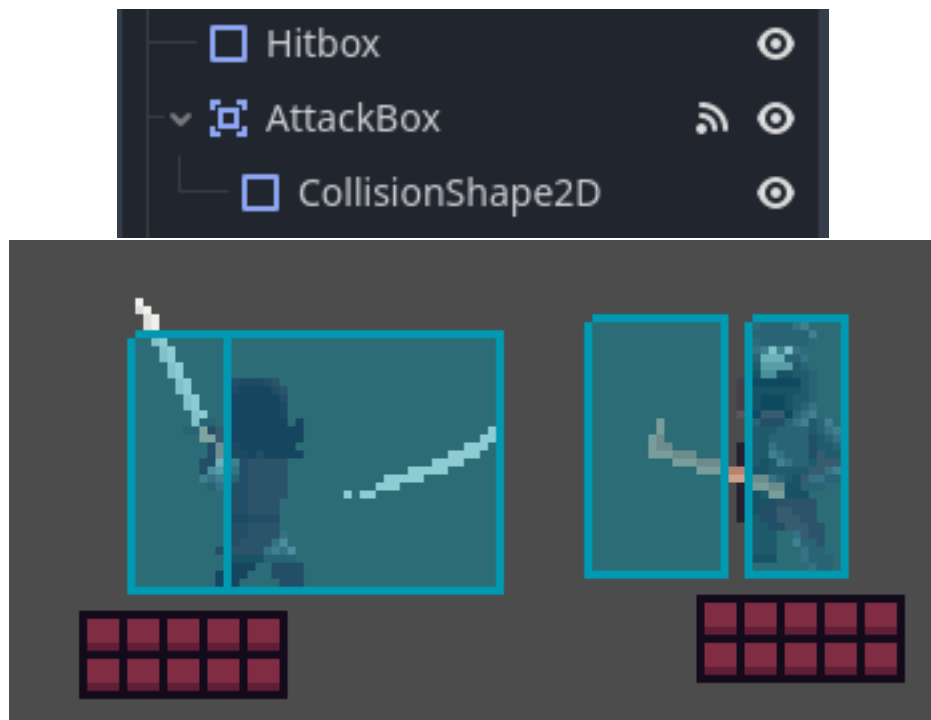
Kamera efekti

Igrač posjeduje čvor *Camera2D* za olakšano upravljanje kamerom iz perspektive igrača. Kamera pri udarcu igrača ili neprijatelja podrhtava kako bi pojačala dojam udarca.

Detekcija kolizija

Hitbox i Attackbox

Igrač i neprijatelj imaju vlastite odvojene *hitboxove* i *attackboxove*: područja na kojima udarci imaju utjecaj, odnosno područja izvršavanja napada. *Attackboxovi* aktiviraju se samo tijekom napada te pri sudaru s *hitboxima*, šalju signale napada vlasnicima pogođenim *hitboxima*. Ovisno o kojem je tipu napada riječ, informacije o tom napadu biti će prenešene udarenom liku te će taj lik pretrpiti štetne posljedice tog napada.



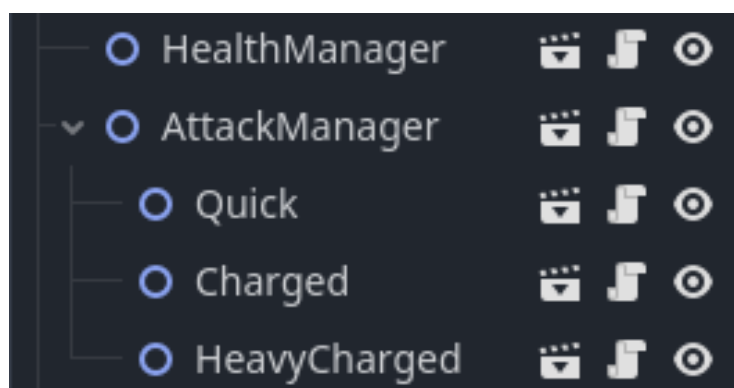
Sustavi komponenata

Upravljanje zdravljem i napadom

Igrač i neprijatelj imaju komponente *HealthManager* i *AttackManager*.

HealthManager olakšava praćenje zdravstvenih podataka poput maksimalnog i trenutnog zdravlja, vremena nepobjedivosti nakon udarca, itd. Uz to omogućuje i sučelje za trenutno zdravlje.

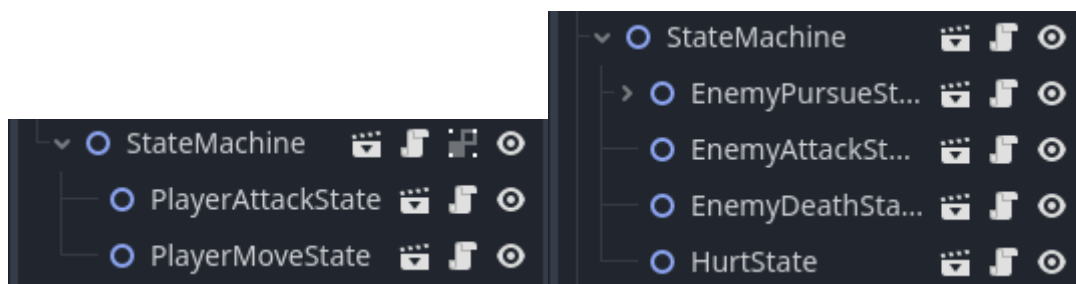
AttackManager pohranjuje različite vrste napada koji lik može izvršiti, svaki s vlastitim vrijednostima za štetu, silu odbacivanja, trajanje ošamućenosti, itd.



Dodatne značajke

Stroj stanja

Instance igrača i neprijatelja koriste se strojem stanja (*state machine*) radi jednostavnije definicije ponašanja, a isto tako jednostavnije ugradnje pravila ponašanja u objekte. Svako ponašanje definirano je u pojedinom stanju, iz koje se prelazi u drugo stanje u trenutku ispunjenja postavljenih uvjeta. Igrač ima stanje kretanja (koje ujedno uključuje i stajanje na mjestu) i stanje napada. Neprijatelj ima četiri sljedeća stanja: naganjanje igrača, napadanje igrača, stanje udarca, stanje smrti.



Animacija smrti

Neprijatelj ima posebnu animaciju smrti s odrubljenom glavom. Ovo je postignuto pokretanjem akcije jedne jedine čestice sa slikom glave neprijatelja. Čestica se pokreće istovremeno kad *AnimationPlayer* izvodi animaciju pada obezglavljenog neprijatelja.



Tvorac neprijatelja

Radi lakšeg testiranja, tvorac (*spawner*) neprijatelja generira neprijatelje jednog po jednog u intervalima nasumične duljine.