Kingless

Realizat de Eduard-Cristian Podaru

Descriere

Kingless este un joc roguelike 2D cu elemente de RPG care se diferențiază de alte jocuri asemănătoare prin accentul pus pe poveste.

Jocul este despre un rege blestemat care s-a trezit într-o zi într-un cimitir al unui sat din regat, sub formă de skeleton. Neștiind ce s-a intâmplat cu el și regatul lui, și neputând să vorbească, acesta trebuie să se bazeze pe ceea ce îi povestește un om misterios pe care îl întâlnește in sat. De asemenea, regele trebuie să treacă printr-o serie de peșteri și catacombe pentru a ajunge înapoi la castel unde îl așteaptă tronul său.

Gameplay

Acțiunea jocului se petrece ori în sat, ori în dungeon. În sat jucătorul poate interacționa cu 4 personaje cu dialog care progresează în funcție de cât de mult te-ai jucat, de câte ori ai murit, la ce Level a ajuns regele, etc. De la unul dintre aceste personaje poți cumpăra iteme precum o hartă care să îți faciliteze gameplay-ul. Alt personaj este omul misterios care îi cere jucătorului să îi aducă suflete găsite în dungeon. Când acesta este plătit cu suflete, povestea avansează și jucătorul află mai mult despre lumea jocului și eventual află identitatea secretă a omului.

Tot în sat jucătorul poate să își crească nivelul pentru a deveni mai puternic folosind Experiență, pe care o acumuleaza omorând monștri din dungeon.

În dungeon jucătorul trebuie să treaca prin 3 nivele diferite pentru a ajunge la ultimul nivel al jocului unde va avea loc un Boss Battle între rege și cel care l-a blestemat. Aceste nivele sunt generate diferit de fiecare dată, astfel, chiar dacă jucătorul moare, data viitoare când încearcă să ajungă la tron, aventura până acolo va fi aproape total diferită.

Tehnologii și resurse folosite

* Unity 2018.3.7f1
* Codul este scris în întregime în C#, folosind și diverse clase din Unity
* Photoshop pentru editarea materialelor grafice
* Elementele grafice sunt preluate de pe opengameart.com și modificate și/sau adaptate la nevoile aplicației de către mine

Documentație

Din punct de vedere grafic jocul este redat folosind Tilemap-uri în Unity. Am importat tileset-uri de pe opengameart.com pentru a îmi crea Tile Palette-uri cu care am desenat pe straturi, pe un grid, întreg satul, nivelele din dungeon și layout-ul fiecărei camere din dungeon. Această stratificare permite jucătorului să apară în fața sau în spatele anumitor obiecte, și să poată sau nu să treacă prin ele.

Pentru mișcarea jucătorului și a inamicilor este folosit sistemul de fizică din Unity, în concordanță cu Coroutine care permit rularea mai multor secvențe de cod în același timp. Rularea mai multor secvențe de cod simultan permite unui caracter să facă mai multe acțiuni în același timp.

Un Prefab în Unity este o stocare a formei exacte a unui obiect care poate fi instanțiat oricând, având toate caracteristicile obiectulu original.

Generarea nivelelor este realizată de instanțe ale clasei FloorGenerator. Pentru aceasta se folosesc prefab-uri ale unor camere goale(cu aspect diferit pentru fiecare nivel) echidistante. Inițial este doar o cameră în nivel, după care pentru fiecare coridor spre altă cameră se adaugă într-o coadă poziția pe care trebuie sa fie instanțiată o cameră noua și se reține poziția într-o listă de poziții deja ocupate.

Coada este de tipul clasei RoomPosition, și pentru fiecare element din coadă se instanțiază o cameră care se potrivește în nivel(cu coridoarele corespunzătoare spre camere alăturate deja instanțiate), apoi se instanțiază un layout pentru cameră(obstacole precum pietre, gropi) și un set de inamici. Camera potrivită alesă, obstacolele din cameră și grupul de inamici sunt fiecare alese random, ceea face nivelele sa fie diferite de fiecare data când sunt jucate.

Am folosit o clasă numită GameManager în care sunt scrise metode care implementează funcționalități tehnice ale jocului precum schimbarea nivelelor, update-uri la highscore-uri, înregistrarea datelor salvate etc. Metodele statice din această clasă facilitează un cod scris eficient și ușor de înteles de către alt programator.

Dialogul cu caracterele din sat este bazat pe 3 clase: Dialogue, DialogueStarter și DialogueData. Clasa Dialogue reglează UI-ul și afișează replici în funcție de prefab-uri de instanțe a clasei DialogueData care sunt accesate prin intermediul clasei DialogueStarter când jucătorul apasă tasta E lângă un caracter.

Replicile alese de clasa DialogueStarter pentru a fi afișate se aleg în funcție de progresul jucătorului(variabile precum de câte ori a murit, ce nivel are, dacă deține un anumit item, etc.) Fiecare caracter are un script atașat care alege instanța de DialogueData care ar trebui să fie folosită.

Codul care pune in funcțiune boss-ul de la finalul jocului folosește Coroutine și metode precum ShootFireball() pentru a ataca jucatorul cu diverse mișcări specifice care se execută dinamic din poziții diferite din camera în care se petrece lupta.