

Get Rich, not Richard

– Proiect la Programare Vizuală –

Student:

Daniel HORT

Timișoara

Ianuarie 2022

Cuprins

Cuprins	1
Introducere	2
Studiu bibliografic	3
Fundamentare teoretică	5
Specificațiile aplicației.....	8
Implementare	10
Utilizarea sistemului.....	16
Concluzii	17
Bibliografie	18

Introducere

Jocurile video există ca o formă de distracție și relaxare pentru unii, poate pentru majoritatea, uneori reprezintă o provocare sau o sursă de frustrare și aici vorbesc din proprie experiență. Pentru acest proiect am ales să realizez un joc scurt de explorare care să pună la dispoziție multe oportunități utilizatorului. Îmi plac jocurile de explorare și aventură deoarece sunt în general relaxante și te recompensează pentru că te abate de la drumul principal. E foarte important să existe mai multe moduri de a ajunge de la A la B, în afară de cel evident. Astfel îți sunt puse la testare abilitățile de observație ale jucătorului. De asemenea orice capăt de drum, orice destinație principală sau secundară trebuie să conțină ceva de valoare, care poate să se afle sub diferite forme în funcție de care e mecanica principală a jocului.

Acest proiect va fi realizat în mediul vizual de programare *Kodu* [1]. Motivul principal pentru care am ales această platformă a fost faptul că are suport genial pentru *gamepad*, atât de bun încât nici nu trebuie să folosești mouse-ul și tastatura, nici măcar pentru a “scrie” cod. Un alt motiv ar fi biblioteca de obiecte, animații și evenimente sau interacțiuni dintre obiecte care există predefinite. În același timp este imposibilă crearea sau adăugarea de noi *asset*-uri, deși mă îndoiesc că asta va reprezenta o problemă. Mai sunt puse la dispoziție o serie de instrumente care te lasă să te dezlănțui creativ în crearea lumii jocului. În cele din urmă am ales *Kodu* [1] pentru că mi s-a părut cel mai distractiv.

Acest mediu de programare este dedicat mai ales copiilor cu vârstele între 8 și 14 ani și este folosit ca instrument educațional pentru a învăța ce este programarea și cum să programezi într-un mod distractiv, perfect pentru copiii mici care vor doar să se joace și au o atenție de scurtă durată. Chiar și-așa, eu consider că oricine se poate bucura de jocul pe care îl voi realiza, oricine cu vârsta de peste 8 ani, care știu să folosească un *gamepad* sau un mouse și o tastatură, pentru că altfel nu se pot juca.

Scopul acestui joc este de a oferi o experiență cât mai plăcută utilizatorilor. De-a lungul jocului vor fi plasate diferite opțiuni și alegeri care vor antrena creativitatea și spiritul de observație. Va exista și un aspect moral, îi putem spune, care va deveni mai aparent în capitoul următor. Planul pe care mi l-am propus este ca toate provocările pe care le voi implementa să aibă mai multe soluții și fiecare alegere să aibă repercusiuni permanente care nu influențează numai rezultatul final dar și lumea jocului. Astfel prin alegerile pe care le face, jucătorul modelează lumea în care se află personajul pe care îl controlează. Pentru harta lumii voi începe cu un *design* aleatoriu și voi construi de acolo.

Mecanica principală a jocului va fi transformarea și progresia liniară, ceea ce înseamnă că de-a lungul drumului vei fi capabil să înveți abilități care îți vor permite accesul în locuri până acum inaccesibile. Vei începe cu un caracter inițial care va putea trage cu rachete pentru a distruge anumite obstacole și vei putea obține un alt caracter cu abilitatea de a sări și un altul cu abilitatea de a înota. Această idee a fost inspirată dintr-un videoclip [2]. Toate jocurile au nevoie de un scop, poate nu toate, dar jocul meu va avea unul. Va trebui să devii bogat, deoarece să arunci o grămadă de bănuți de aur peste tot în lume și să îl pui pe jucător să îi colecteze e o metodă leneșă dar s-a dovedit că funcționează. Nu pot să am doar idei geniale, există un *deadline* până la urma, proiectul trebuie să fie gata anul acesta.

Studiu bibliografic

Lumea spune că sunt un mare *gaymer* [3]. Prin lume mă refer la mine și la vocile din capul meu. Revenind:

Pentru început o să vorbesc despre videoclipul menționat în capitolul anterior [2]. În acel videoclip este prezentat un joc dezvoltat în mediul vizual de programare Kodu [1], jocul este foarte complex și se numește *The Wizard Games*. Principala mecanică a jocului este transformarea, interacționând cu anumite obiecte personajul tău se transformă într-un alt robot. Fiecare robot are propriile abilități care pot fi folosite pentru a trece peste obstacolele puse în calea ta, care sunt diverse și necesită o anumită abilitate. În *demo* este prezentată o situație care are mai multe soluții. Pentru a trece de partea cealaltă a unui râu, ai opțiunea de a te transforma într-un robot mare care poate sparge bolovanii ce blochează podul sau de a te transforma într-un pește care să înoate până pe malul celălalt. De asemenea lupta e un aspect foarte important al jocului, scopul final fiind acela de a învinge toți inamicii.

După ce m-am uitat la *demo*-ul de la *The Wizard Games* am decis ca mecanica principală a jocului meu să fie de asemenea transformarea. Caracterul controlat de utilizator va fi în stare să își schimbe forma cum dorește odată ce învață cum să o facă. Vor fi în total 3 abilități, cea de a trage cu o rachetă, de a sări foarte sus și de a înota .

UNDERTALE [4] e un joc realizat de Toby Fox și publicat pe *Steam* în anul 2015. A prins foarte repede popularitate din cauza modului unic în care poate fi jucat. Povestea începe cu un copil care cade într-o groapă și ajunge într-o altă lume populată de monștrii. Mai departe nu contează atât de mult ce se întâmplă, partea importantă este cum te joci. Din când în când vei întâlnii monștrii pe tot parcursul jocului, fie că apar la întâmplare sau te împiedică să progresezi. Când se întâmplă asta ai 4 opțiuni din care să alegi, după cum se poate vedea în figura Fig. 1.



Fig. 1 exemplu din lupta finală

Poți ataca, iar un atac realizat cu succes va scădea din viața monstrului. Poți folosi un obiect din cele pe care le-ai strâns pe parcursul jocului. Poți realiza o acțiune specifică pentru monstrul cu care te afli momentan în conflict. Ultima opțiune este de a cruța, odată ce un monstru are viața foarte scăzută va

obosi și, în loc să îl omori, îl poți cruța. Dacă execuți corect acțiunile corespunzătoare în ordinea corectă vei putea de asemenea să cruți un monstru. De exemplu există un monstru câine în joc pe care trebuie să îl mângâi până devine foarte fericit apoi obosește și adoarme, după îl poți cruța. Jocul nu te obligă să faci nimic propriu zis, îți spune că nu trebuie să omori niciun monstru, îi poți cruța, dar e alegerea ta ce faci mai departe. Fanii jocului au dat câte un nume pentru cele 2 metode favorite de a juca jocul. Există ruta pacifistă în care cruți pe toată lumea și există ruta genocid în care nu scapă nimeni în viață. Ruta pe care o alegi influențează multe aspecte ale jocului dar mai ales lumea în sine. Mai târziu o să ajungi într-un mic oraș foarte vesel populat de monștrii prietenoși dacă alegi ruta pacifistă, dar dacă alegi cealaltă rută atunci vorba despre un ucigaș în serie va ajunge în oraș înaintea ta și vor pleca cu toții, astfel că orașul va fi pustiu.

Pentru a nu limita creativitatea jucătorului, am decis să nu fac nimic prea evident. Totuși voi face tot posibilul ca să implementez subtil cele 2 rute care există în *UNDERTALE* [4]. Evident ruta genocid va fi cumva alegerea mai ușoară sau mai la îndemână deoarece de la început vei avea abilitatea de a trage cu rachete. Celelalte 2 abilități vor fi necesare pentru ruta pacifistă și vor fi obținute ulterior.

Termenul *metroidvania* se referă la o categorie de jocuri cum ar fi *Hollow Knight* [5] sau *Yoku's Island Express* [6] și multe altele. Nu voi intra în foarte mult detaliu dar există un aspect foarte important, care mă interesează la această categorie. Mă refer la progresie liniară. Nu înseamnă că am un drum drept pe care trebuie să merg, înseamnă că la început am acces doar la o zonă limitată din hartă dar în această zonă pot obține o abilitate sau un obiect care să îmi deschidă accesul la o altă zonă unde există altă abilitate sau obiect care îmi permite accesul altundeva și tot așa. În *Hollow Knight* [5] *Isma's Tear* îți permite să înoți în acid și din senin multe locații sunt acum explorabile Fig. 2.



Fig. 2 "Acid shall be repelled. Swim in acidic waters without coming to any harm"

Pe aceeași idee, diferitele abilități ce vor exista și la mine în joc vor guverna felul în care se realizează progresul și vor oferi acces la locații de mare importanță.

Fundamentare teoretică

Programarea vizuală presupune utilizarea unui mediu de programare „vizual” pentru a scrie programe. E o metodă neconvențională de a „scrie” cod într-un mod non-textual. Din această categorie se disting limbajele de programare orientate pe aspectul educațional, create pentru a-i învăța pe copii ce e programarea și cum să scrie programe și jocuri și limbaje de programare specializate pe domenii tehnice în care o reprezentare vizuală are mai mult sens, cum ar fi *LabVIEW* [7] pentru circuite electronice și altele sau *Simulink* [8] pentru reprezentarea sistemelor matematice complexe. Din cauza naturii lor, aceste limbaje de programare sunt foarte dependente de mediul de programare.

Kodu [1] este un limbaj vizual de programare, deja ați înțeles tema acestei lucrări, iar dacă nu, nu v-o spun. E necesar să fie descărcat de pe site-ul oficial sau instalat de pe *Microsoft Store* pentru a putea fi folosit și este compatibil doar cu sistemul de operare *Windows*, de la *Windows Vista* la *Windows 10*. Din experiența mea recomand și o placă video destul de bună.

În sine, *Kodu* [1] este un „motor” de dezvoltare pentru jocuri 3D care oferă instrumente de modelare de teren, simulare de apă statică (adică e doar vizual, nu și din punct de vedere fizic), o colecție largă de caractere și obiecte statice decorative, multe animații și interacțiuni și setări pentru ați personaliza jocul.

Pe lângă tastatură și mouse, oferă suport total pentru *gamepad*, fiind motivul principal pentru care am ales *Kodu* [1]. În figura Fig. 3 se poate observa cum arată meniul principal de editare a jocului. După cum se poate observa totul e vizual, până și meniul conține doar imagini, dar pentru fiecare opțiune e prezentată și o mică descriere.



Fig. 3 meniul principal de editare

Fiind folosit pentru dezvoltarea jocurilor 3D, consumă multe resurse grafice și RAM, din acest motiv există un indicator, un termometru, ce indică volumul de resurse care sunt folosite. Când termometrul e aproape plin se recomandă să nu se mai adauge niciun element nou în nivel. Eu nu m-am oprit până ce termometrul nu a ajuns ca în Fig. 4.

Codul este alcătuit dintr-o condiție și o acțiune care să se execute când condiția devine adevărată. Atât condiția, cât și acțiunea pot fi omise, dar nu ambele în aceeași linie. Motivul pentru care acest lucru este permis este pentru a putea compune mai multe condiții sau pentru a executa mai multe acțiuni în cadrul aceluiași eveniment fără a rescrie din nou condiția. Un exemplu se regăsește la Fig. 5.

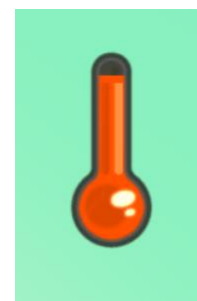


Fig. 4



Fig. 5 exemplu de cod

Pentru a introduce un obiect în joc trebuie selectată o locație și apoi ales obiectul respectiv ce se dorește a fi adăugat, ulterior se poate modifica locația. Pentru situație în care este folosit un *gamepad* meniul care apare este circular și cu sub-meniuri Fig. 6.



Fig. 6 meniu de selectare a unui obiect/caracter

În cele din urmă, ca orice sistem software, are „bug”-uri, Fig. 7.

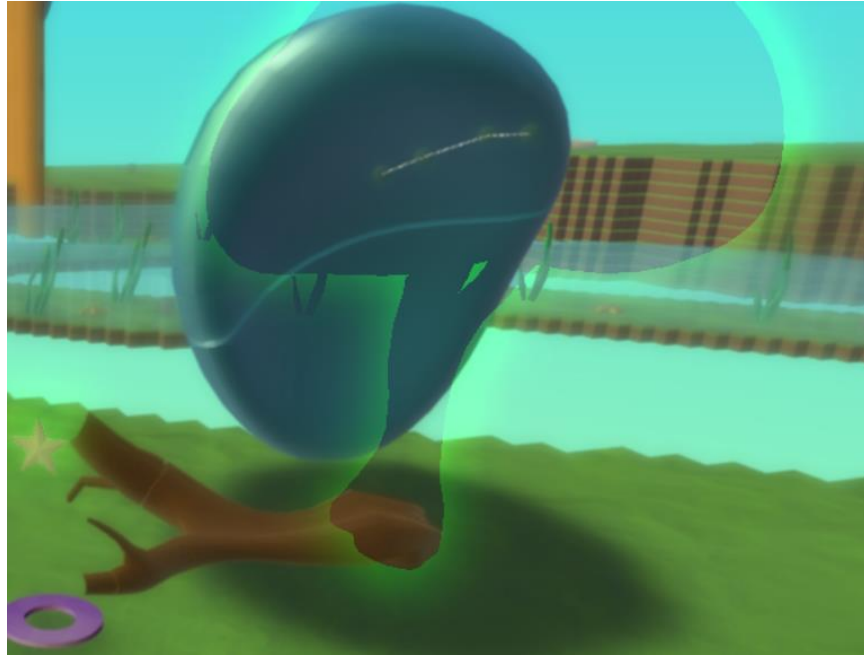


Fig. 7 copacul meu e adolescent și nu mă mai ascultă

Specificațiile aplicației

Procesul creativ a fost unul care s-a pornit mai greu. Fără o idee în cap am trecut la crearea hărții și am făcut-o destul de la întâmplare. Știam doar că va fi o secțiune lungă și îngustă de apă, asemănătoare cu un râu și că trebuie să creez o zonă înaltă accesibilă doar sărind. Când harta a fost gata și roaie de idei mi-au năvălit în minte și am început să populez nivelul cu obiecte și obstacole. Schema din Fig. 8 descrie evenimentele și succesiunea lor.



La începutul jocului deja poți face foarte multe lucruri. Îți poți croi drum distrugând totul în cale, aceasta ar putea fi cea mai evidentă soluție. Dacă înveți repede să sari poți trece ușor peste primul obstacol sau să ceri ajutor de la natură vorbind cu un copac ce se află la mare înălțime. Atacând un copac vei atrage furia naturii asupra ta și nu te va mai ajuta în aventură. Obstacolul al 2-lea poate fi doar distrus, după care înveți să înoți, culegi toți bănuții din apă și câștigi, dacă îi ai și pe ceilalți.

Jocul devine interesant când încerci să te abați de la drumul din fața ta. Înainte să înveți să înoți, plutești deasupra apei și poți să plutești pe toată suprafața apei până la locul unde înveți să înoți. Tot acolo se află niște case care aparțin muncitorilor de la fabrica de muniție și desigur fabrica de muniție. Obstacolul al 2-lea e alcătuit din 3 ziduri care pot fi distruse cu rachete dar distrugând casele cu rachete va rezulta în distrugerea zidurilor. Pe schemă se poate observa relația aceasta exemplificată printr-o linie întreruptă. De asemenea prin distrugerea fabricii, tunurile de la primul obstacol vor rămâne fără muniție și vor fi dezactivate.

Singurul obiectiv al jocului este de a deveni bogat colectând bani de aur care se află în mod convenabil peste tot. Pentru a face acest lucru e obligatoriu să înoți în apă, deoarece există bani acolo și nu poți deveni bogat dacă nu îi colectezi pe toți. Pentru ruta genocid nu este necesar să înveți cum să sari, ambele obstacole pot fi distruse dar pentru ruta pacifistă e necesar să poți sări peste primul obstacol sau să ceri binecuvântarea pădurii ca să blocheze tunurile.

Pentru a fi mai interesant sau mai enervant, pe toată harta se află niște zepeline care aruncă cu bombe din aer. Din cauza asta poți fi lovit oricând pierzând tot progresul. Grozav, nu? Nu. Dar nu se simțea suficient de greu așa că a trebuit să adaug o provocare în plus. Zepelinele nu pot fi distruse și muniția lor sunt stele căzătoare, nu au nevoie de fabrica de muniție ca să le alimenteze așa că nu poți scăpa de ele.

Implementare

A fost odată ca niciodată o lume goală în *Kodu* [1]. Deși nu avea nimic de arătat încă, forțe de neconceput aveau deja planuri mărețe pentru ea. Și așa a început totul, pe un pământ verde, mic și pustiu. Prima dată a fost *kodu* (robotul) și de acolo multe minuni s-au înfăptuit.

Primul lucru pe listă a fost implementarea mecanicii de transformare. Pentru ceva de genul acesta *Kodu* [1] pune la dispoziție ceva numit *Creatables* [9]. Un *creatable* este un obiect care încorporează setări și cod, pe lângă modelul 3D asociat, care nu apare în joc decât dacă e creat de un alt obiect. Prima încercare a fost destul de simplă, *kodu* și *cycle* au fost transformați în *creatables* și pe fiecare a fost scris cod astfel încât la apăsarea butonului Y de pe *gamepad* să îl creeze pe celălalt și apoi să dispară. Din păcate atâta timp cât butonul era apăsat *kodu* și *cycle* se tot creau unul pe altul foarte repede până ieșeau de pe hartă. Prin adăugarea unui *cooldown* problema a fost rezolvată. Un *cooldown* este o perioadă de timp care trebuie așteptată după folosirea unei abilități pentru a putea fi din nou folosită. În mod arbitrar am ales să fie 10 secunde. Acum când butonul este apăsat, scorul alb este setat pe 10 înainte să se întâmple pașii discutați anterior și dacă scorul nu e mai mic sau egal cu 0 atunci nu se va întâmpla nimic Fig. 8.

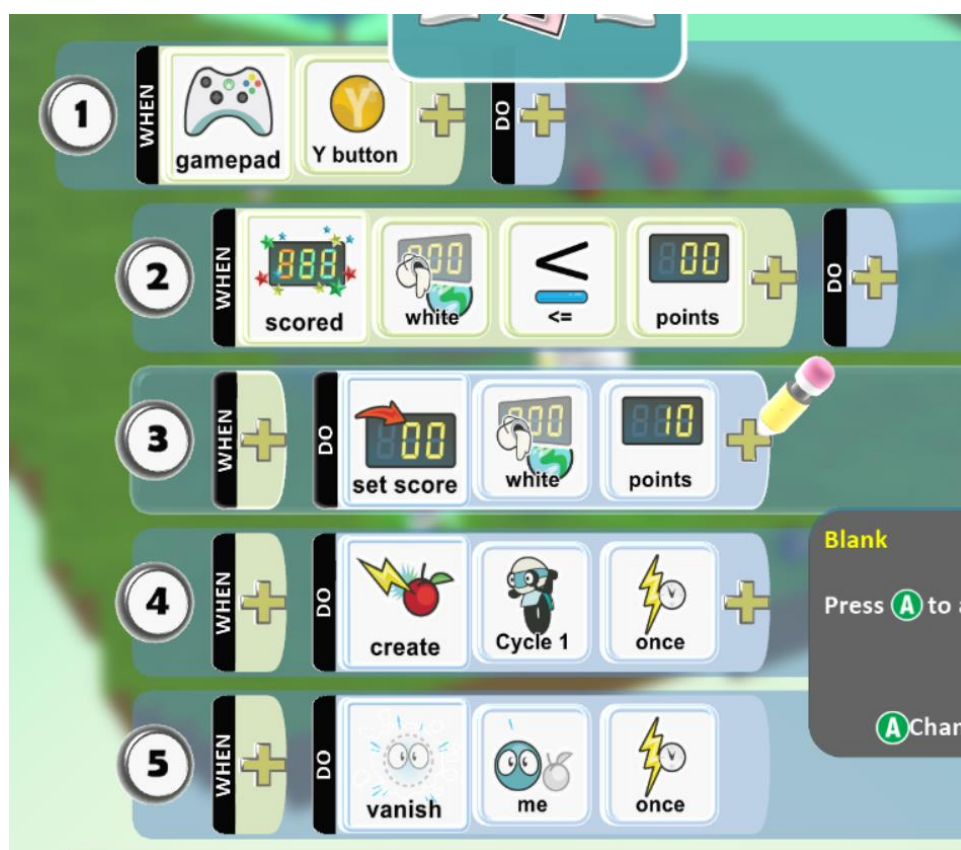


Fig. 8 codul pentru transformarea condiționată lui kodu în cycle

O piatră este plasată la începutul jocului care are rolul de a-l crea pe *kodu* și de a scădea scorul alb, care reprezintă *cooldown*-ul, câte un punct pe secundă Fig. 9.

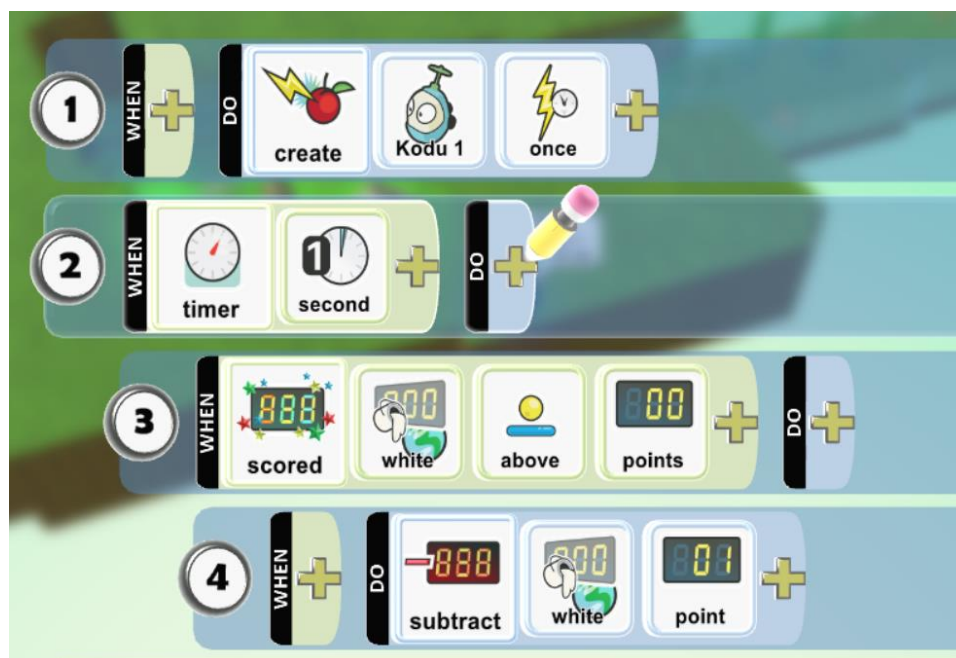


Fig. 9 de aici începe totul

Am plasat în joc mere de diferite culori și *kodu*, mâncându-le, deblochează abilitatea aferentă culorii. Mărul roșu, Fig. 10, îi permite să se transforme în *cycle* care poate sării, mărul albastru, Fig. 11, îi permite să se transforme într-un submarin când se află deasupra apei și înapoi în *kodu* când iese pe uscat. Mărul verde, Fig. 12, se află într-un loc înalt accesibil doar de *cycle* și va crea un șir de copaci care se vor interpune între *kodu* și gloanțele tunurilor Fig. 13.



Fig. 10 mărul roșu ascuns între flori



Fig. 11 mărul albastru între case și fabric, la finalul drumului

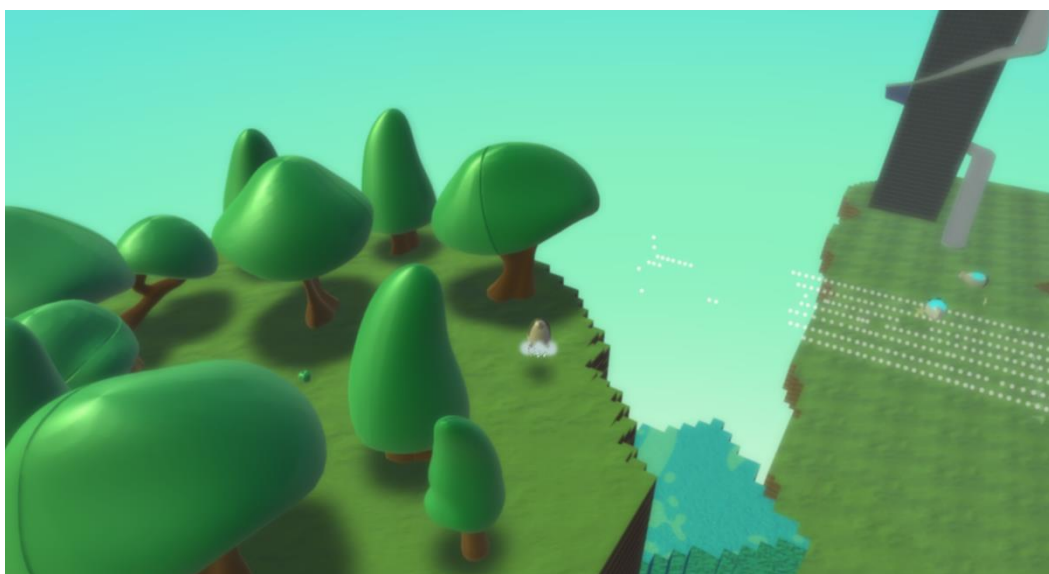


Fig. 12 mărul verde în padure, la mare înălțime



Fig. 13 copacii crescui pentru a-l proteja pe kodu după ce a primit binecuvântarea pădurii

În joc urmează să fie adăugate o serie lungă de evenimente ce pot avea loc, pentru a propaga aceste evenimente peste tot unde e nevoie de ele am venit cu ideea asocierii fiecărui eveniment cu un scor global (nu este afișat pe ecran și este același pentru toate obiectele). Evenimentele au structura formată din cauză (declanșatorul care setează scorul pe 1) și efect (oriunde e nevoie se testează valoarea scorului și dacă e peste 0 atunci se execută niște pași ca urmare a ceva). Un exemplu de eveniment cauză-efect se poate observa în Fig. 14, Fig. 15.

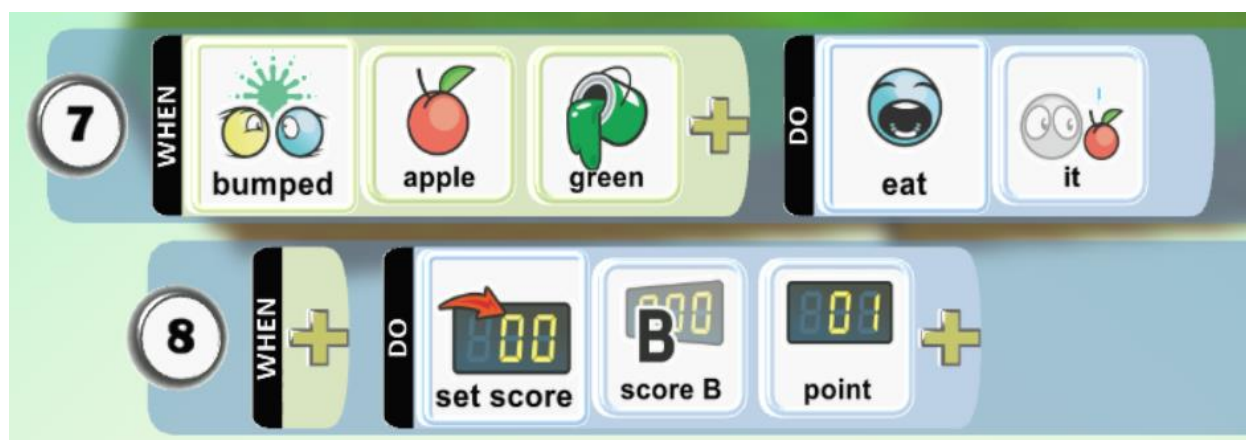


Fig. 14 cauză generate de mâncarea mărului verde, cod de pe kodu

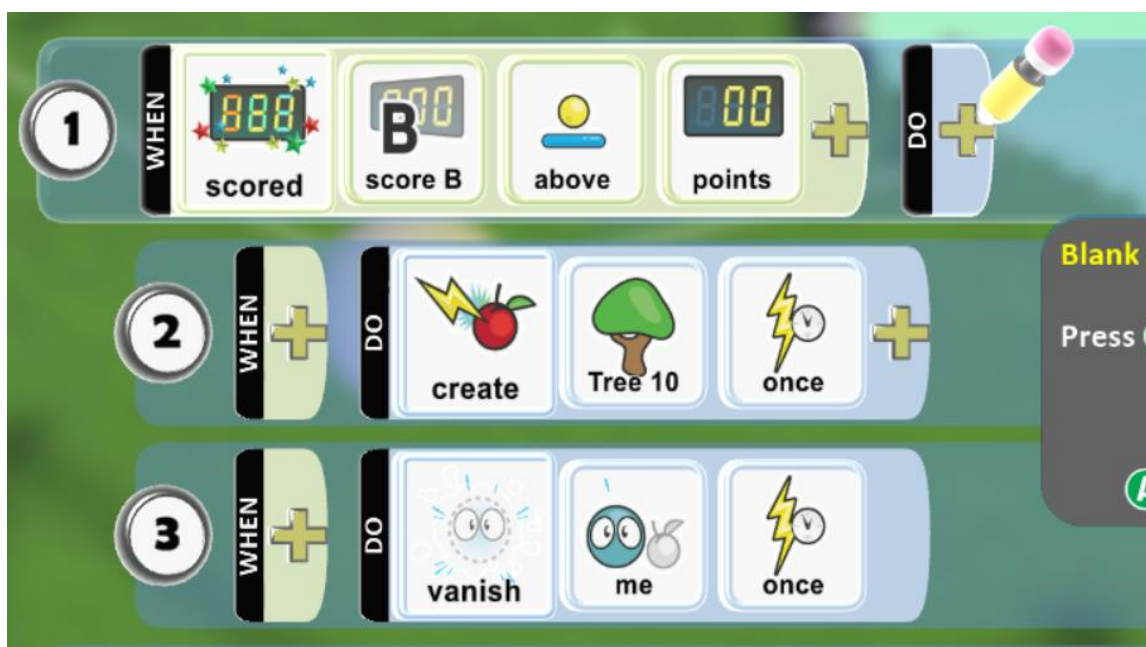


Fig. 15 efect în care se crează un copac din şirul de copaci, cod de pe o piatră invizibilă

Există multe alte exemple de astfel de evenimente. În Fig. 11 se pot observa cele 3 case de culori diferite. Obstacolul al 2-lea este compus din 3 ziduri care au aceleaşi culori ca şi casele Fig. 16, nu e o întâmplare. Distrugând casele zidul ce are aceeaşi culoare va fi distrus de asemenea, *vice versa* nu este adevărat, distrugând zidul casele vor rămâne intacte.

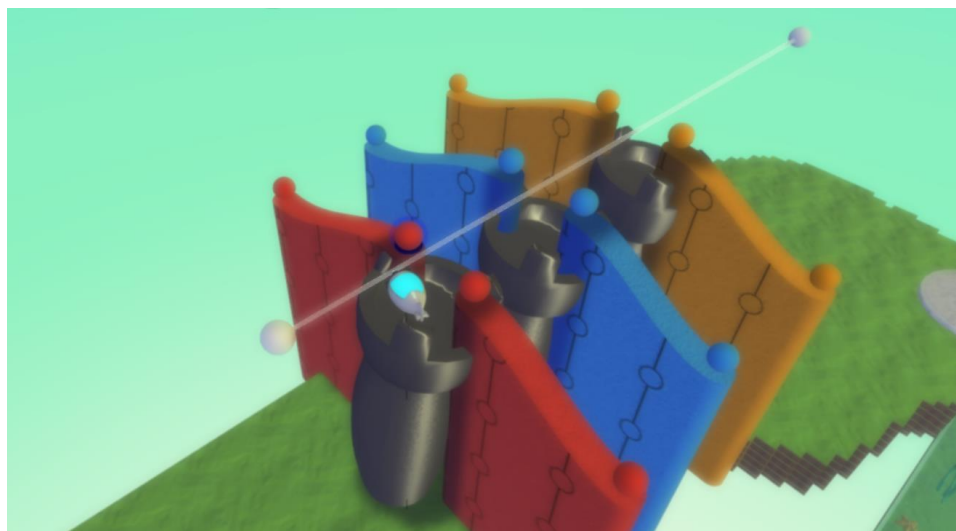


Fig. 16 ziduri colorate, obstacolul al 2-lea

În cele din urmă am decis să adaug mai mult suspans şi aşa au luat naştere zepelinele. Rolul lor este de a arunca cu stele din cer şi dacă una te nimereşte ai pierdut. Fiecare zepelin urmăreşte un anumit drum

creat din puncte Fig. 17. Codul pentru a face zepelinele sa urmărească linia deschisă de puncte este cel puțin trivial Fig. 18.

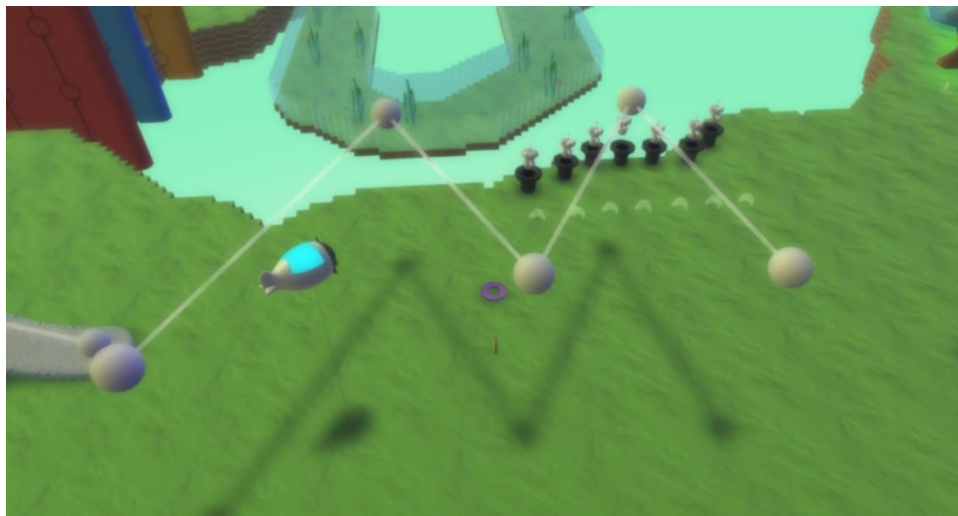


Fig. 17 drumul descris de puncte pe care merge un zepelin



Fig. 18 cod trivial

Deși există mai multe aspecte despre care aş putea vorbi, nu sunt atât de importante şi le voi omite.

Utilizarea sistemului

Înainte de a începe jocul, pe ecran va apărea o descriere, *Fig. 19*, care explică scopul jocului și cum se controlează personajul, ce butoane fac ce. În rest totul rămâne la mâna utilizatorului să descopere pe cont propriu.

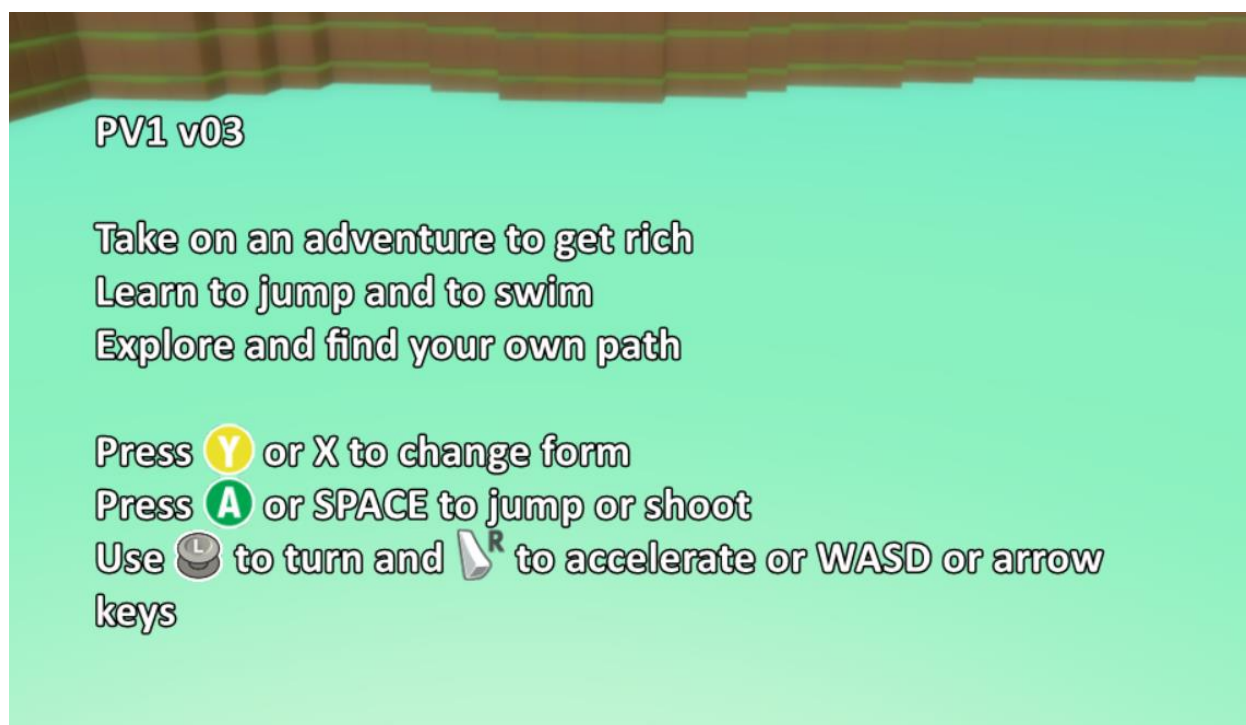


Fig. 19 descrierea jocului

Concluzii

Pentru acest proiect mi-am propus să realizez un joc care, în primul rând, mie să îmi placă și consider că am reușit să îndeplinesc acest aspect.

Între ideile inițiale și rezultatul final au avut loc mici modificări dar în mare parte totul a decurs așa cum am plănuț. Singurul impediment fiind implementarea mecanicii de transformare. Am discutat despre această problemă la capitolul de implementare, unde am oferit si soluția.

Aplicația așa cum este în acest moment, nu e un produs finit sau ideal, harta e destul de goală și destul de mică, însă îmi place obiectivul simplu al jocului și pe acesta nu l-aș schimba.

Și desigur, ce contează cel mai mult este că m-am distrat. Asta contează cel mai mult, nu?

Bibliografie

- [1] Kodu Game Lab, „Kodu,” [Interactiv]. Available: <http://www.kodugamelab.com/>. [Accesat 15 01 2022].
- [2] 1Gamersreview, „YouTube,” 12 12 2011. [Interactiv]. Available: <https://www.youtube.com/watch?v=Hvf7IMWq23c>. [Accesat 15 1 2022].
- [3] *Original: gamer, scris greșit intenționat.*
- [4] T. Fox, „UNDERTALE,” Steam, 15 9 2015. [Interactiv]. Available: <https://store.steampowered.com/app/391540/Undertale/>.
- [5] Team Cherry, „Hollow Knight,” Steam, 14 2 2017. [Interactiv]. Available: https://store.steampowered.com/app/367520/Hollow_Knight/.
- [6] Villa Gorilla, „Yoku's Island Express,” Steam, 29 5 2018. [Interactiv]. Available: https://store.steampowered.com/app/334940/Yokus_Island_Express/.
- [7] National Instruments, „LabVIEW,” [Interactiv]. Available: <https://www.ni.com/ro-ro/shop/labview.html>.
- [8] MathWorks®, „Simulink,” [Interactiv]. Available: <https://www.mathworks.com/products/simulink.html>.
- [9] Kodu Game Lab, „Kodu | Tricks | Creatables,” [Interactiv]. Available: <http://www.kodugamelab.com/tips/creatables>.