**Universitatea Politehnica Timișoara**

**Facultatea de Automatică și Calculatoare**

**Departamentul de Automatică și Informatică Aplicată**

Get Rich, not Richard

* Proiect la Programare Vizuală –

Student:

Daniel HORȚ

**Timișoara**

**Ianuarie 2022**

# Cuprins

[Cuprins 1](#_Toc93096401)

[Introducere 2](#_Toc93096402)

[Studiu bibliografic 3](#_Toc93096403)

[Fundamentare teoretică 4](#_Toc93096404)

[Specificațiile aplicației 5](#_Toc93096405)

[Implementare 6](#_Toc93096406)

[Utilizarea sistemului 7](#_Toc93096407)

[Concluzii 8](#_Toc93096408)

[Bibliografie 9](#_Toc93096409)

# 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 si aventură deoarece sunt în general relaxante si te recompensează pentru că te abați 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 îi 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 in 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 si 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 si 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 si alegeri care vor antrena creativitatea și spiritul de observație. Va exista și un aspect moral, îi putem spune, care va deveni mai aparent în capitolul 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 si 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 sării 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*.

O imagine care conține text, tabelă de marcaj

Descriere generată automat

Fig. 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 si, 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 si exista 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 in *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*.

O imagine care conține text

Descriere generată automat

Fig. “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 oferii acces la locații de mare importanță.

# Fundamentare teoretică

(2—3 pagini)

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 si 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.

O imagine care conține text

Descriere generată automat

Fig. 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.*

Fig.

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. 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. meniu de selectare a unui obiect/caracter

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

O imagine care conține verde, înot

Descriere generată automat

# Specificațiile aplicației

(2—3 pagini)

Aici descrieți designul aplicației voastre. Faceți o schemă bloc a sistemului care este un desen în care, prin dreptunghiuri (sau altfel), indicați elementele componente ale sistemului și, prin săgeți, felul în care acestea sunt conectate. Bineînțeles, și povestiți în cuvinte tot ce conține schema, nu doar o desenați. De asemenea, prezentați pe scurt, ce va face aplicația, Nu intrați în detalii de implementare, pentru că ele vor apărea în capitolul următor.

# Implementare

(5—7 pagini)

Descrieți concret cum ați implementat aplicația voastră, cumva reflectând schema bloc prezentată anterior. Faceți capturi de ecran în care să se vadă implementarea vizuală și descrieți în cuvinte ce se întâmplă.

# Utilizarea sistemului

(1—3 pagini)

Este un manual de utilizare pentru cei care se presupune că vor folosi aplicația. În acest capitol nu au ce să caute detalii de implementare (ele apar în capitolul anterior). Aici prezentați doar modul în care utilizatorul (pe care nu vrea să‑l doară capul despre modul cum ați implementat voi) interacționează cu aplicația. Altfel spus, dacă ar fi să vindeți aplicația, ce ar trebui să știe un utilizator să o poată folosi. Bineînțeles că aici apar și capturi de ecran (câte considerați voi necesare), împreună cu explicațiile aferente. Vă rog să nu puneți doar poze, ci și text. Aceasta este o lucrare științifică, deci nu folosiți expresii de genul „deschide fereastra cutare și apasă pe butonul Save”, ci „se deschide fereastra și se apasă butonul”, adică un mod de adresare impersonal.

# Concluzii

(1—3 pagini)

Aici reluați intențiile prezentate în introducere și spuneți dacă au fost realizate și, dacă e cazul, cum au îmbunătățit viața utilizatorului. De asemenea, puteți menționa problemele de care v-ați lovit și cum le-ați rezolvat. Nu în ultimul rând, cum s-ar putea îmbunătăți aplicația voastră.

# Bibliografie

|  |  |
| --- | --- |
| [1] | „Kodu Game Lab,” [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. |

**Observații**

În plus, vă atrag atenția asupra următoarelor aspecte, mai mult sau mai puțin de tehnoredactare:

* Lucrarea este scrisă în limba română. Adică folosiți diacritice. După mine, e o chestiune de bun simț să scrieți corect (și nu mă refer aici la greșelile de ortografie). În plus, puteți folosi un *speller* de limba română care vă scoate în evidență greșelile
* Fiecare capitol din lucrare începe de la pagină nouă. Trecerea la pagină nouă NU se face apăsând tasta *Enter* la disperare, ci inserând un *Page Break* (sau combinația de taste *Ctrl*+*Enter*). În acest fel, orice modificare în capitolul anterior, nu vă va da peste cap toată paginația din capitolul curent. El va începe întotdeauna de la pagina nouă.
* Titlurile capitolelor și subcapitolelor se marchează ca atare (*Heading 1*, *Heading 2* etc.). Acest lucru vă va fi util pentru generarea automată a cuprinsului. Insist asupra acestui lucru pentru că e cel mai simplu și mai corect mod de creare a unui cuprins. Altfel, o să aveți mai mult de lucru și o să arate urât.
* NICIODATĂ pe ultimul rând al unei pagini nu este un titlu sau primul rând dintr‑un paragraf. Căutați în *Help* cum se rezolvă această problemă, denumită *Widow/Orphan control*.
* Dacă simțiți nevoia să indentați paragrafele (adică să puneți alineat), acest lucru NU se face cu tab la începutul paragrafului ci în mod automat, formatând paragraful. Pentru orice nelămuriri nu ezitați să folosiți *Help*-ul editorului Word.
* În afara situației în care aveți o poză foarte mare (care ocupă toată pagina), NICIODATĂ pe o pagină nu apare doar o poză. Și nici o poză imediat după titlu nu se poate. TOATE imaginile sunt numerotate și explicate (exp. Fig. 1.1 Relaționarea tabelelor). Explicația este plasată sub imagine. Răsfoiți o carte tehnică și vedeți cum e scrisă. Nu folosiți exprimări de genul „figura de mai sus/jos”, ci în Fig. 1.1. În acest mod, puteți pune figura acolo unde are loc și nu lângă paragraful care o menționează.
* După un titlul nu urmează o imagine ci un text (măcar un paragraf) și apoi o imagine. În ultima vreme întâlnesc tot mai des această greșeală.
* Lucrarea are *Header* și *Footer*. Nu ezitați să le folosiți. Pentru a avea *Header* diferit în fiecare capitol, se împarte lucrarea în secțiuni.
* Prima pagină a lucrării de diplomă (cea de tilu) nu are nici *Header*, nici *Footer*. Prima pagină a unui capitol are doar *Footer*, niciodată *Header*.
* Într‑o lucrare științifică, toate cuvintele în altă limbă decât cea de redactare se marcheză cu italic. Toate bucățile de cod se scriu cu font Courier New și, în general, mai mic decât cel al textului normal.
* Pentru ca paragrafele să arate frumos, folosiți împărțirea automată în silabe *(Automatic Hyphenation)*. Pentru asta este nevoie să scrieți în română.