**MINION TO THE RESCUE**

Предмет на оваа семинарска работа претставува платформска игра во духот на анимираниот филм Despicable me. Идејата за истата не претставува копија на определена игра и секоја од класите и целокупниот код е наша сопствена имплементација, базирана и испишана во едноставен C#, без употреба на посебни библиотеки за полесно креирање на игри.

Накратко речено она што е цел на играчот, кој е претставен како Minion е да ги помине сите три стази кои се пат до ослободувањето на една од трите посвоени ќерки на фамозниот Felonius Gru, кои беа киднапирани од страна на негативецот Vector.

Сите користени форми во проектот ги поставивме да бидат дел од еден MDI Container со цел за полесна навигација низ целата апликација. Она со што се среќава првично корисникот **(слика1)** е форма за зачленување (доколку немате веќе свој профил) или логирање во веќе постоечкиот профил, кој го поткрепивме со воведување на SQL база, која го зачувува единствениoт Username на корисникот, како и поените (добиени преку собирање на парички и банани, трансформирани манјони со прости математички пресметки) од веќе поминатите стази.



Слика1

По регистрацијата или логирањето се наоѓате на својот профил **(слика2)** кој како опција нуди да смените своја слика, Ве отпоздравува и Ви нуди опција да ги забележите најголемите скорови кои се изиграни и зачувани во локалната база од различни веќе постоечки играчи **,** да увидите по која веќе изиграна стаза сте најдобри. Исто така може да се запознаете со инструкциите на играта, како и да добиете еден краток опис за истата **(слика3)**.



Слика2



Слика3

Откога ќе истражите што е основата на оваа апликација преку бирање на копчето New Game имате можност да изберете која од трите девојчиња Margo, Edith или Agnes ќе се обидете да ги спасите овој пат. За полесна прегледност од Ваша страна ги ослободуваме сите три нивоа, без притоа да е зависно да се победи првото ниво за да се влезе во второто и слично, иако идеалната замисла беше токму таа. Во секоја од стазите имате поставено одреден број на парички и банани кои треба да се обидете да ги соберете без притоа да го изгубите животот. Иницијално имате 3 живота, но со собирање на 50 парички, играчот добива додатно еден живот. Низ целата игра се обидувате да ги ретрансформирате лошите манјони така што им вбризгувате инекција со тајната напивка која ќе ги врати во нивната почетна состојба. Зависно од нивото на стазата која ја играте манјоните имаат различна брзина, број и доаѓаат од различни насоки. Доколку манјонот ве фати губите еден живот и се враќате онаму до каде сте дошле. Внимавајте лошите манјони доаѓаат многу брзо!

1. Margo **(слика4)**:

Во ова ниво се спасува најстарата сестра, па при тоа манјоните се пуштени само од една страна и тоа токму онаму каде Вие се движите за да го пронајдете девојчето. Нивната брзина и фреквенција на надоаѓање е најниска од сите други стази. Доколку ја изодете целата патека која содржи 150 парички и стигнете до крајот каде ја чуваат заробената сестра добивате скор за определниот левел. **(слика6)** Постои додатно копче кое ве носи повторно на избирачката форма каде избирате кое ниво ќе го играте следно.



Слика4

2. Edith **(слика5):** Многу слично на претходното, се борите со лошите манјони, но овој пат тие се насекаде, со многу поголема брзина, подготвени да Ви ги уништат сите животи.



Слика5

3.Agnes **(слика6):**

Во третото ниво, воедно и најтешкото се соочуваме со ултра најбрзите лоши манјони и посетата на Vector кој нема да се обиде да ви направи ништо но доколку го поразите добивате x 50 за секој негов пораз.

 Слика6

Бидејќи целата логика на играта на еден начин е вметната во еден тајмер TmrMoving и на неговиот тик се случуваат и проверуваат многу состојби, ќе ја објасниме накратко идејата во овој тик настан. Искористен е еден тајмер за целата игра кој е дел од класата MainForm со интервал од 15ms, но преку низа бројачи се реискористува (наместо многу тајмери) и за други состојби на играта.

* MovingTimer() со оваа функција проверуваме дали корисникот кликнал на левата, десната стрелка или се обидел да скокне, (при притискање на копчињата се променуваат определните bool атрибути кои тука ги проверуваме) како би ги апдејтирале координатите на играчот.
* Со counterShooting кој се тригира на секои 2 вакви тика повикуваме функција, која имплементира логика таква што доколку инекцијата е видлива се придвижува зависно од насоката а исто така се апдејтира и правецот на инекцијата во случај во меѓувреме да се свртел играчот. Доколку инекцијата не е видлива на екранот тогаш може повторно да се пука.
* Со counterKilling се проверува дали инекцијата е сеуште видлива на екранот и дали е убиен некој лош манјон. Исто така со countermove и counterAddEnemy се придвижуваат негативците и на секоја скоро една секунда се додава по еден манјон.
* Вметнати се и некои плус проверки за дали карактерот е убиен, дали се собрани 50 парички (плус живот) и доколку изгуби еден живот да се врати повторно во игра, исто така доколку е дојден крај на левелот (изминати сите 150 парички низ екранот) тогаш се поставуваат како visible панелата и лабелата за крај во оваа форма.
* Додатно доколу се работи за последното ниво вметната е слична логика и за непријателот Vector.

Освен ова интересна логика која ја имплементиравме во проектот е и начинот на кој добивме илузија на scrolling background. Најпрво за секој левел во една листа вметнавме по две слики кои низ текот на целата игра ги апдејтираме. Во суштина кога играчот ќе започне да се движи на десно дозволено му е “вистинско“ движење во однос на позициите на формата се до една фиксна позиција каде го фиксираме карактерот и и при секое ново придвижување позадината се движи согласно кон каде се движи нашиот карактер. Како би дознале дали играчот е на таа позиција имплементиравме функција checkIfOnPosition() која се повикува на тајмерот и која ни менува вредност на променливата MoveBackground која подоцна се користи при движење на играчот. (ако е вистинита играчот е фиксиран, во спротивно играчот навистина се движи).

/// <summary>

/// функција која проверува дали играчот при стартување на играта стасува до

/// фиксната позиција каде започнува илузијата.

/// </summary>

public void checkIfOnPosition()

{

if (Character.X == FixedPoint && !ComesBackToStart)

{

MoveBackground = true;

}

}

**ИЗРАБОТИЛЕ: ДАЈАНА ВЕЛИЧКОВСКА И МАРТИНА ДЕНКОВИЌ**