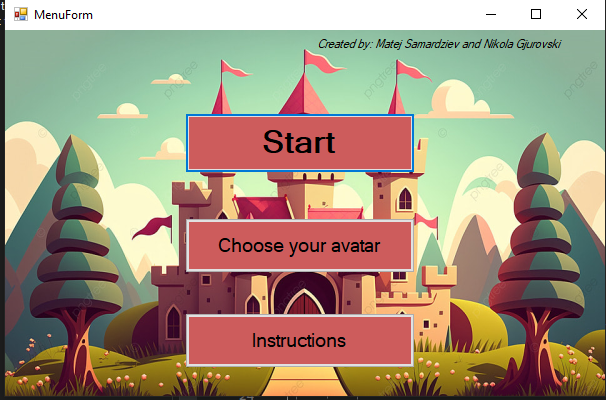
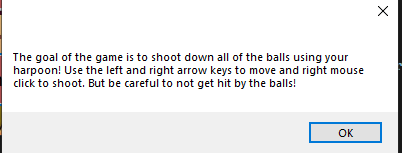
Опис на апликацијата

Апликацијата што ја развивавме е класичната игра Bubble Trouble која може да ја пристапите на следниов линк <https://poki.com/en/g/bubble-trouble>.

Вака изгледа почетниот прозорец на нашата игра

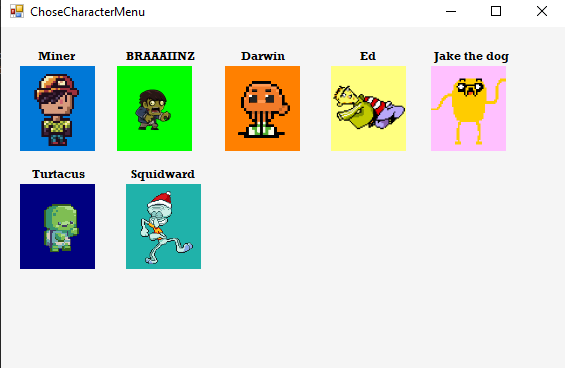


Од овдека корисникот може да ја започне играта со притискање на копчето старт или да одбере со кој аватар сака да игра како и да ги види контролите на играта со кликање на копчето за инструкција, чив прозорец изгледа вака



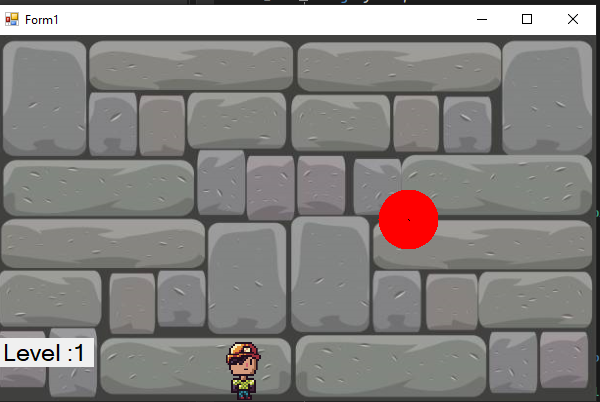
Овдека се објаснува правилата дека херојот не смее да го допре топка , а негова цел е да ги уништи сите топки кои му се појавуваат, притоа да внимава дека со секое пукање кон топче тоа се дели на две помали топчиња. Карактерот се движи со левата и десната стрелка, а ги пука топчињата со десниот клик на маусот.

Со цел да се врати носталгија кај играчот и да се направи поинтересна играта додадовме можност корисникот да избере свој херој со кој би сакал да ја започне играта. Во прилог е како изгледа тој прозорец со неговите карактери



На овој прозорец се сите карактери со кои може да се игра играта. Со десен клик на карактерот играта започнува со селектираниот карактер.

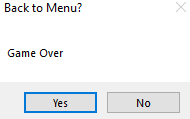
Вака изгледа почетокот на играта избрана со првиот карактер со име Miner



Како што може да се види на прозорецот лево од него се пишува бројот на левелите. Во нашата игра има бесконечно левели, играта нема крај секој левел станува потежок со покомплексни топчиња , нивни локации и движења. Најголема е виолетовата топка, па сината, па црвената, па жолтата. По победувањето на левелот започнува броење на тајмер од три секунди пред да започне следниот левел.



Доколку топче го удри херојот тогаш се појавува прозорец кој прашува дали сака корисникот да се врати во менито или да ја изгаси играта



Организација на апликацијата

Од самата организација на апликацијата се состои од три класи, Circle во оваа класа се содржат методите за движење на топчињата и нивното цртање. Hero , оваа класа е посветена за самиот херој која се состои од метод за цртање на карактерот, неговите движења лево и десно, како и проверка дали има негов удар со некоја топка, додадена е и метода да се внимава карактерот да остане во рамките на дадениот прозорец. Line е класата за линијата која ја испушта херојот, овдека има методи за цртање на линијата, да се пресмета големината на линијата и да се провери дали линијата удрила топка. Scene класата се состои од објект од класата Hero, како и четири листи од класата Circles, именувани по сите бои виолетова, сина, црвена и жолта. Методите овдека се за цртање, да се нацртаат сите топчиња, методи за додавање на соодветно топче по соодветната боја, метод да се додаде линија и да се избрише линија. Важен метод е тој во кој се проверува во одреден циклус дали херојот се удрил со некој од топчињата, потоа проверка за херојот да биде во рамките на мапата, како и метода за дали одредена линија погодила некоја од топчињата, се проверува во соодветен циклус за секоја од топчињата.

Collision Detection with the hero

Најпрвин иако херојот претставува одреден gif, неговиот hitbox е невидлив круг околу него со негова големина. Проверка за дали има судар со одреден друг круг го правиме со Евклидовата формула односно со следната функција во класата Hero:

public bool checkCollison(Circle circle)

{

int distance = (int)Math.Sqrt(

Math.Pow(Center.X - circle.Center.X, 2) +

Math.Pow(Center.Y - circle.Center.Y, 2)

);

return distance <= Radius + circle.Radius;

}

Потоа во класата Scene за секој вид на круг кој што се наоѓа во своја соодветна листа, се повикува функцијата и се проверува дали тој соодветен круг се судрил со херојот.

public bool isHit()

{

foreach(Circle c in BlueCircles)

{

if (hero.checkCollison(c))

{

return true;

}

}

foreach (Circle c in PurpleCircles)

{

if (hero.checkCollison(c))

{

return true;

}

}

foreach (Circle c in RedCircles)

{

if (hero.checkCollison(c))

{

return true;

}

}

foreach(Circle c in YellowCircles)

{

if (hero.checkCollison(c))

{

return true;

}

}

return false;

}

Потоа во класата Form1, во функцијата timer\_tick, се повикува isHit методата на Scene и доколку врати true, играта завршува и се појавува

DialogResult dg = MessageBox.Show("Game Over", "Back to Menu?", MessageBoxButtons.YesNo);

if (dg == DialogResult.Yes)

{

MenuForm menu = new MenuForm();

menu.Show();

this.Hide();

}

else

{

Application.Exit();

}