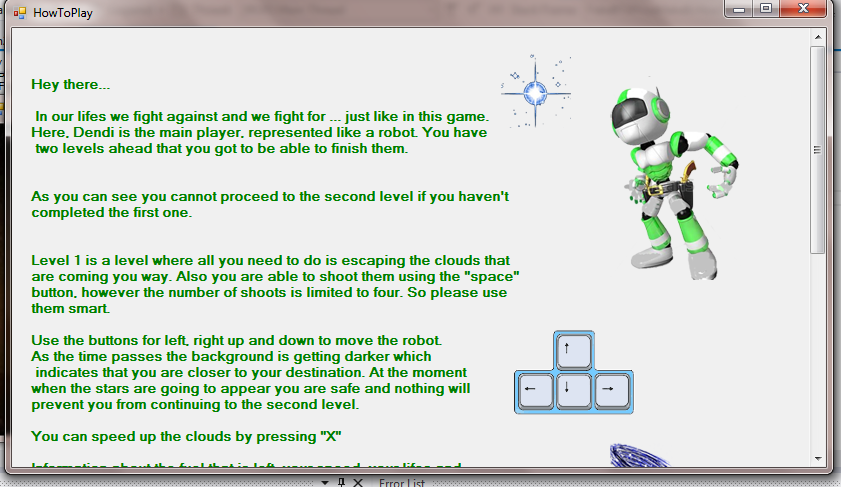
Проектната задача е игра која има две различни нивоа во кои се имплементирани различни функционалности. Идејата за играта, во која има едно роботче кое полетува од Земјата и стигнува во космосот, потекнува од фактот дека сите тргаме од негде, од некоја основа и пробуваме да стигнеме до нашата цел без разлика на препреките. Така и тука.

На почетокот на играчот му се јавува првичниот прозор во кој е овозможено само да го избере првото ниво. Второто ниво ќе може да го игра само ако успее да го помине првото ниво, без да ги загуби сите животи на роботчето. Воедно во почетниот прозор играчот добива и упатство како се игра играта.

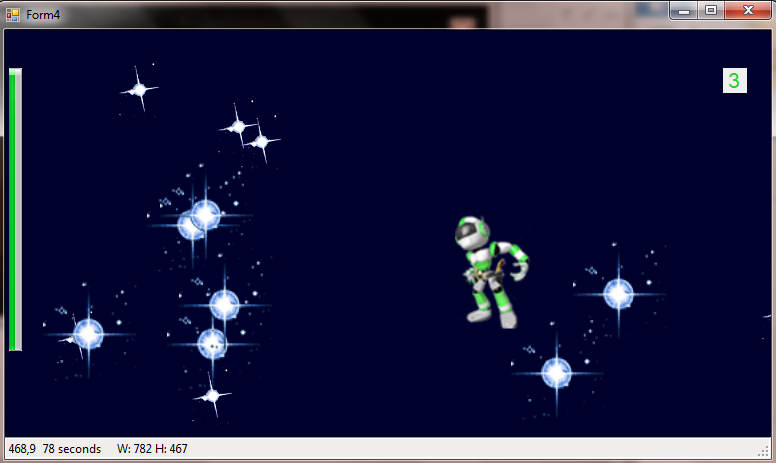


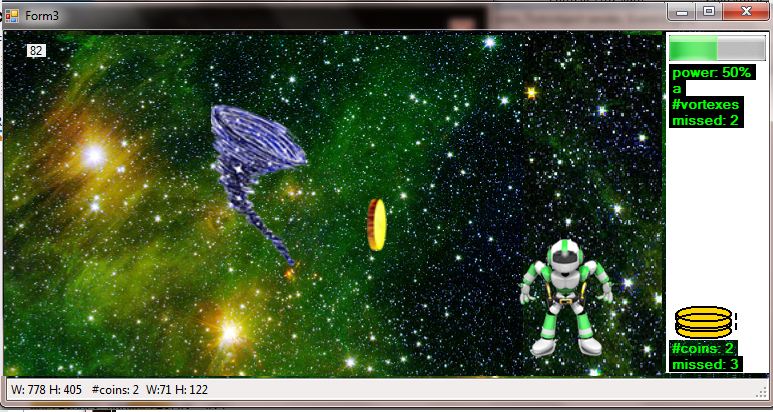
Подетни упатства за како се игра играта играчот може да добие со кликања на Instructions.

Откако ќе биде избрано првото ниво се отвора нова форма во која се прикажува играта каде на роботчето му се одбројува времето до кога тој полетува кон космосот. Како што се качува нагоре, небото темнее од светло сина боја па се до црна боја. Тоа е имплементирано со помош на бројачи кои ја намалуваат својата вредност на секој тик на тајмерот. Преку groundColor = new SolidBrush(Color.FromArgb(brojac1, brojac2, brojac3));

 brojac1 и brojac2 се намалуваат за 4 се додека не добијат вредност 0. За тоа време роботчето додека лета нагоре се судрува со облаци кои се појавуваат рандом. Облаците претставуваат фотографии кои се исцртуваат и појавуваат во определено време.

Исто така на почетокот во прво ниво се појавува авионче кое прелетува хоризонтално.. Роботчето треба да внимава да не се судри со него.

Кога brojac1 и brojac2 добиваат вредност 0, ќе се активира brojac3, кој се намалува до 0 и позадината од темна сина темнее до црна боја (FromArgb(0,0,0)=црна). Облаците кои се појавуваа претходно и со кои ако се судреше роботчето губеше живот, веќе не се генерираат, а наместо тоа почнуваат да се појавуваат ѕвездите што означува дека роботчето стигнало во космосот и првото ниво го завршило успешно.

Со завршување на првото ниво, изборот да го игра второто ниво е сега овозможено. Ако се продолжи на второто ниво, роботчето лета во космосот. Се уште ја менува својата позиција ако се сврти на лево, десно или право, меѓутоа сега неговите моќи се зголемени. Роботчето освен што лета, може и да пука. Неговата задача е да собира паричка притоа не судрувајќи се со препреките претставени во вид на вртлоци по патот. Тука е обрнато внимание паричките и вртлоците кои се појавуваат рандом, да не се генерираат надвор од границите на формата, да не се преклопуваат едни со други и ако рандом позицијата се совпадне со моменталната позиција на роботчето, тие не треба да се генерираат. Пример како е изведено тоа за паричката:

За да не се исцртува паричката надвор од границита на формата, освен можноста да се стави Rectangle во чии граници ќе се исцртува паричката(тоа е применето за некои од другите класи), тука во самата класа,во DrawImage, кога се исцртува фотографијата се ставени границите во кои само ако координатите се наоѓаат е дозволено таа да се исцрта.

public void DrawImage(Graphics g, int width, int height)

{

if (X <= width - coinImg.Width-122 && Y <= height - coinImg.Height-10)

{

coinImg = Resources.coin;

g.DrawImage(coinImg, X, Y);

}

}

Ако позицијата на роботчето не се совпаѓа со паричката, тогаш може да се додаде паричката во листата на парички. Наједноставен начин да се стори тоа, е околу двете фотографии да се создаде Rectangle со висина и ширина на фотографиите.

Rectangle coinRec;

if (countCoins % 30 == 0)

{

cX = random.Next(Width - 100);

cY = random.Next(Height - 70);

coinRec = new Rectangle(cX, cY, coin.coinImg.Width, coin.coinImg.Height);

if (!coinRec.IntersectsWith(robotRec))

coins.AddCoin(cX, cY)

}

Дополнително се искористени гласовите од ликовите на играта Дота, кои се појавуваат со притискање на одредени копчиња во играта, а роботчето се вика Денди.

Денди е креиран со помош на класата која се вика Robot, чии аргументи се:

public enum DIRECTION

{

RIGHT,

LEFT,

DOWN,

UP

}

public DIRECTION Direction { get; set; }

public int X { get; set; }

public int Y { get; set; }

public Image robotImg { get; set; }

public int poeni { get; set; }

public Plane plane { get; set; }

За да го креираме роботчето ни треба конструкторот. Позицијата каде ќе се појави роботчето во секоја од формите е поставено во формите соодветно.

public Robot(int x, int y)

{

X = x;

Y = y;

Direction = DIRECTION.UP;

robotImg = Resources.PSgreenrobotFinal;

}

Со помош на DIRECTION и стрелките на тастатурата ја определуваме позицијата на страната на роботчето, додека со координатите X, Y моменталната негова позиција во формата. Бидејќи во второ ниво Денди собира поени, имаме аргумент за тоа. Роботчето е фотографија која беше обработено во Photoshop и која се исцртува на формата преку

public void DrawRobot(Graphics g)

{

if (Direction == DIRECTION.LEFT)

{

robotImg = Resources.GR;

}

else if (Direction == DIRECTION.RIGHT)

{

robotImg = Resources.GL;

}

else

robotImg = Resources.PSgreenrobotFinal;

g.DrawImage(robotImg, X, Y);

}

За да може роботчето да се сврти на лево или десно кога ќе биде кликнато соодветно копче на тастатурата служат if условите. Главната функција преку која се изведува движењето на роботчето и во која се ставени границите до кои Денди има право да се движи е MoveRobot:

public void ChangeDirection(DIRECTION direction)

{

Direction = direction;

}

public void MoveRobot(int width, int height)

{

if (Direction == DIRECTION.RIGHT)

{

X += 10;

if (X >= width-90)

{

X = width - 90;

}

}

if (Direction == DIRECTION.LEFT)

{

X -= 8;

if (X < 0)

{

X = 0;

}

}

if (Direction == DIRECTION.UP)

{

Y -= 8;

if (Y < 0)

{

Y = 0;

}

}

if (Direction == DIRECTION.DOWN)

{

Y += 10;

if (Y >= height - height / 3 - 30)

{

Y = height - height / 3 - 30; ;

}

}

}

Во двете нивоа Денди може да пука. Бројот на пукања е ограничен на четири, меѓутоа во првото ниво пукањето се прави со помош на „space“ копчето, додека во првото ниво тоа се прави со помош на клик на глувчето. Формата на оружјето е различно. За оружјето во првото ниво потребно е да се чуваат координатите на моменталната позиција на Денди, за да може во хоризонтална положба нагоре, со притискање на space да се пука.

wX = robot.X - 50;

wY = robot.Y - 100;

weaponForClouds = new WeaponForClouds(wX, wY);

А во второто ниво се чуваат само координатите на местото каде е кликнато со глувчето, и ако на тоа место по ширината и должината на орудието се совпадне со ширината и должината на vortex-от, препреката т.е. vortex-от се уништува.

if (vortexRec.IntersectsWith(weaponRec))

{

vortexDocs.vortexes.Remove(vortexDocs.vortexes[i]);

vortexesDestroyed++;

tsslWeapon.Text = String.Format("#weapons left: {0} vortexes destroyed: {1}", weaponCount, vortexesDestroyed);

. . .

Исто така во првото ниво се овозможува забрзано движење на облаците со стискање на копчето X.

Фотографиите кои се тука употребени се претходно дообработени и исечени во Photoshop. Паричките, роботчето, оружјата, облаците се фотографии, и за секој од нив има посебни класи во кои се имплементирани соодветно конструкторите, функциите за движење, исцртување и функциите за дополнителните функционалнисти. Исто така со помош на SoundPlayer се чуваат звуците кои се појавуваат од време на време во играта. Гласовите се превземени од определени ликови во играта Dota од линкот <http://dota2.gamepedia.com/Heroes>.

И фотографиите и звуците се чуваат во Resources.