

**Gymnázium a Střední průmyslová škola elektrotechniky a
informatiky, Frenštát pod Radhoštěm**

Obor č. 18-20-M: Informační technologie

**Projekt do předmětu PRG
Rizzler**

**František Špačinský
T3A**

2022/2023

ANOTACE

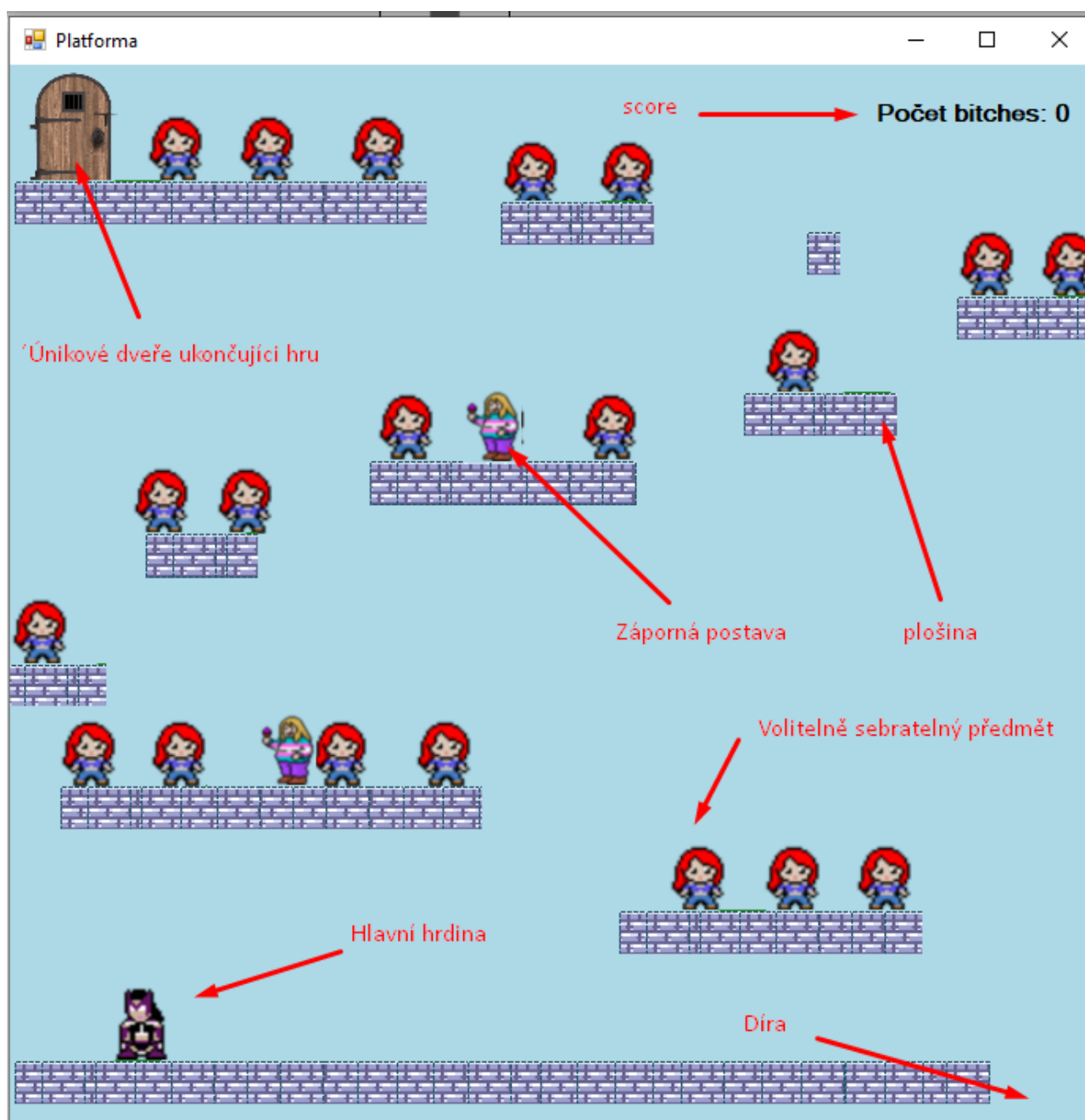
Tato práce se zabývá vytvoření, popisu a realizaci akční 2D plošinové hry s názvem Rizzler. V této práci lze nalézt vysvětlení k jednotlivým částem kódu k lepšímu pochopení daným funkcím ve hře. Cílem této práce je předvést možnosti tvůrčí představivosti za pomoci programovacího jazyku C# a win forms.

1. Popis hry a její funkce

Táto hra se zakládá na tom, že hlavní hrdina (Rizzler) má obvyčejné schopnosti jako jest například chůze, nebo skok. Však čím hlavní hrdina vyniká je jeho vysoký rizz, tudíž jeho šance na to, že rizzne bitch (pěknou dívku) se pohybuje někde okolo 106–112%.

Na jeho cestě ke kouzelným dveřím však se musí dostat přes různé překážky, které dělají tuto hru obtížnější a tudíž více zábavnou a déle hratelnou.

2. Grafický rozbor prvního levelu



3. Přehled funkčnosti jednotlivých prvků

3.1 Hlavní hrdina (hráč)

Hlavního hrdinu jsem vytvořil za pomoci entity „PictureBox“, které jsem poté mohl přidávat další vlastnosti, kterou tuto entity rozvíjely. Jako první věc jsem musel vyřešit způsob jakým bych touto entitou mohl pohybovat a to jsem docílil tím, že jsem využil akci na klávesnici pomocí metod „keyDown“ a „keyUp“.

```
private void keyDown(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Left)
    {
        goleft = true;
    }
    if (e.KeyCode == Keys.Right)
    {
        goright = true;
    }
    if (e.KeyCode == Keys.Space && jump == false)
    {
        jump = true;
    }
}

1 reference
private void keyUp(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.Left)
    {
        goleft = false;
    }
    if (e.KeyCode == Keys.Right)
    {
        goright = false;
    }
    if (jump == true)
    {
        jump = false;
    }
    if (e.KeyCode == Keys.Enter && gameover == true)
    {
        RestartGame();
    }
}
```

Metoda „keyDown“ je vyvolaná při stisknutí klávesy a „keyUp“ opět když klávesu pustíme. Například při stisknutí pravého tlačítka se nám nastaví hodnota goright na true a když pustíme tak poté zpátky na false.

```

if (goleft == true)
{
    MoveCharacter(-playerspeed);
}
if (goright == true)
{
    MoveCharacter(playerspeed);
}
if (jump == true && force < 0)
{
    jump = false;
}

if (jump == true)
{
    JumpSpeed = -8;
    force -= 1;
}
else
{
    JumpSpeed = 10;
}

```

Zde je ukázka hodnot, jak reagují a fungují při když se nastaví do pozice true.

Takže ve výsledky když zmáčknu klávesu zapne se mi metoda keyDown , zhodnotí se mi, které tlačítko jsem zmáčknul a poté nastaví daná funkce, která hýbe s hlavním hrdinou.

Dále jsem řešil případ obrázku hlavního hrdiny, aby se natáčel tím směrem, kudy se hlavní hrdina vydává.

```

2 references
private void MoveCharacter(int distance)
{
    character.Left += distance;

    if ((distance < 0 && isFacingRight) || (distance > 0 && !isFacingRight))
    {
        character.Image.RotateFlip(RotateFlipType.RotateNoneFlipX);
        isFacingRight = !isFacingRight;
    }
}

```

Jednoduše řečeno tato podmínka kontroluje směr pohybu a podle toho otáč

3.2 Plošina

Plošinu jsem musel upravit tak, aby se dalo po ní chodit nebo na ní vyskočit. Daný problém jsem řešil takto

```
if (x is PictureBox)
{
    if ((string)x.Tag == "platform")
    {
        if (character.Bounds.Intersects(x.Bounds))
        {
            force = 8;
            character.Top = x.Top - character.Height;
        }
        x.BringToFront();
    }
}
```

Všechny pictureboxy které jsem si označil jako „platform“ jsou nastavené tak, že pokud se postava hlavního hrdiny překrývá s platformou tak pozice hrdiny je upravena tak, aby byla nad plošinou.

3.3 Sběratelný předmět

Dále jsem do hry přidal nějaké sběratelné předměty, aby hráč mohl získat i nějaké to scóre.

```
if ((string)x.Tag == "bitch")
{
    if (character.Bounds.Intersects(x.Bounds) && x.Visible == true)
    {
        x.Visible = false;
        score++;
        Globals.Score = score;
    }
}
```

Pokud hlavní hrdina dotkne a tím pádem rizzne ženu (v tomto případě označované jako „bitch“) tak daná entita zmizí a stane se neviditelnou. Do toho se ještě přidá hodnota +1 do scóre a aktualizuje se proměnná Globals.score o které si povíme později.

3.4 Scóre

Nastavil jsem si entitu label a pojmenoval ji txtScore. Na tomto nebylo nic těžkého co nastavit jen snad hodnotu počátečního scóre a později co má daná entita ukazovat za text při různých eventech.

```
int score = 0;
↑ reference
private void MainGameTimeEvent(object sender, EventArgs e)
{
    txtScore.Text = "Počet bitches: " + score;
```

3.5 Záporák

Dále aby byla hra o něco obtížnější jsem do hry přidal i nějaké záporáky. Přesněji řečeno dva. Tento problém na vyřešení byl vskutku o něco složitější než předešlé problémy. Nejdříve jsem musel vyřešit to, aby mi záporáci neopouštěli daný úsek plošiny.

```
enemyone.Left += enemyonespeed;
if (enemyone.Left + enemyone.Width >= platform2.Left + platform2.Width)
{
    enemyonespeed = -Math.Abs(enemyonespeed);
}
else if (enemyone.Left <= platform2.Left)
{
    enemyonespeed = Math.Abs(enemyonespeed);
}

enemytwo.Left += enemytwospeed;
if (enemytwo.Left + enemytwo.Width >= platform5.Left + platform5.Width)
{
    enemytwospeed = -Math.Abs(enemytwospeed);
}
else if (enemytwo.Left <= platform5.Left)
{
    enemytwospeed = Math.Abs(enemytwospeed);
}
```

Zde jsem upravil entitu záporáka tak, že jsem nastavil podmínky, které kontrolují, jestli záporák dosáhl hranice své přidělené plošiny. Pokud ano tak hodnota „enemyspeed“ se zneguje a pokračuje na druhou stranu, než se dostane na horizont levé strany plošiny, a tak dokolečka.

Dále jsem přidal to, aby byli záporáci vidět a nepřekrývali se s jinými prvky.

```
enemyone.BringToFront();
enemytwo.BringToFront();
```


Dále jsem do hry přidal to, aby nám záporák dával nějaké poškození. V tomto případě jsem to vyřešil přímým ukončením hry.

```
if ((string)x.Tag == "enemyone" || (string)x.Tag == "enemytwo")
{
    if (character.Bounds.Intersects(x.Bounds))
    {
        timer.Stop();
        gameover = true;
        txtScore.Text = "Počet bitches: " + score + "\n" + "To nebyla žena :(";
        restartButton.Enabled = true;
        restartButton.Visible = true;
    }
}
```

Pokud se hlavní hrdina dotkne nepřítele. Tak se zastaví pohyb ve hře, oznamovací bar vpravo nahoře nám zahlásí hlášku naší smrti a zobrazí hodnotu našeho dosavadního scóre. Jak si můžete povšimnout k tomuto ukončení hry jsem přidal tlačítko „restartButton“ který se nám zobrazí a po jeho stisknutí nám umožní použít metodu restart game.

```
1 reference
private void restartButton_Click(object sender, EventArgs e)
{
    RestartGame();
}
```

Při spuštění metody se stane následující:

```
2 references
private void RestartGame()
{
    jump = false;
    goleft = false;
    goright = false;
    gameover = false;
    score = 0;
    Globals.Score = 0;

    txtScore.Text = "Počet bitches: " + score;

    foreach (Control x in this.Controls)
    {
        if (x is PictureBox && x.Visible == false)
        {
            x.Visible = true;
        }
    }

    character.Left = 66;
    character.Top = 584;

    timer.Start();
    restartButton.Enabled = false;
    restartButton.Visible = false;
    this.Focus();
}
```

V první části jsem musel nastavit ovládací prvky do „false“, aby se hlavní hrdina nezačal automaticky nějak pohybovat při restartu hry. Dále jsem vyresetoval hodnotu score zpátky na 0 a použil metodu foreach která projde každý ovládací prvek a pokud je neviditelný, znovu ho zjeví. Dále to posune hlavního hrdinu zpět na původní hodnoty zapne timer a opět restart tlačítko schová.

3.6 Únikové dveře

Dále jsem nastavil nějaký konec/cíl pro tuto hru a to jsem zvolil dveře, které jsem upravil takto

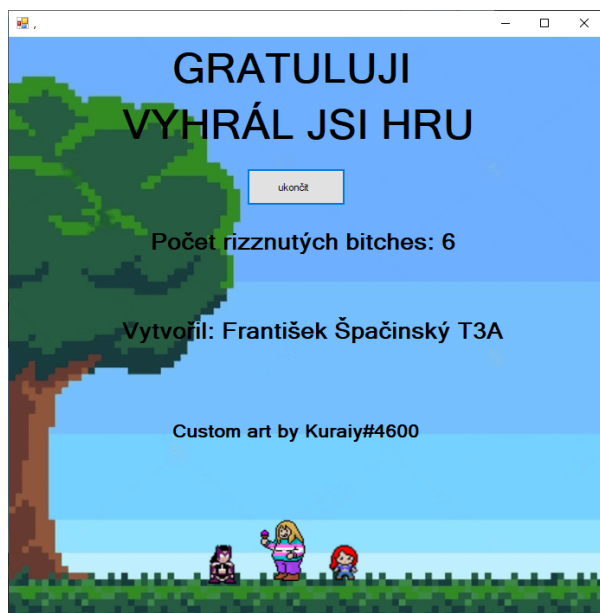
```
if (character.Bounds.IntersectsWith(door.Bounds) && !levelCompleted)
{
    levelCompleted = true;
    konec newLevel = new konec();
    newLevel.FormClosed += (s, args) =>
    {
        this.Show();
        timer.Start();
        levelCompleted = false;
    };

    timer.Stop();
    gameover = true;
    newLevel.Show();
    this.Hide();
}
```

Pokud se hlavní hrdina dotkne dveří tak nám to otevře další form (nový level).

Dále se starý form zastaví a zavře.

3.7 Nový level



Nový level už jsem vytvořil jako zakončující část hry, avšak i na tomto snímku, který vypadá jednoduše jsem musel vyřešit problém s hodnotou scóre z minulého formu. Tento problém jsem vyřešil takto

```
Globals.Score = 0;
```

```
score++;  
Globals.Score = score;
```

```
1  using System;  
2  using System.Collections.Generic;  
3  using System.Linq;  
4  using System.Text;  
5  using System.Threading.Tasks;  
6  
7  namespace konecdubnaseblizi  
8  {  
9  
10     3 references  
11     public class Globals  
12     {  
13         3 references  
14         public static int Score { get; set; }  
15     }  
16 }
```

Tímto jsem nastavil v prvních form, aby se scóre rovnalo globálnímu scóre a při restartu se znovu globální hodnota scóre vynulovala. Poté jsem nastavil veřejnou třídu kde si proměnou scóre pomocí přístupových metod get a set.

```
1 reference
private void konec_Load(object sender, EventArgs e)
{
    rizz.Text = "Počet rizznutých bitches: " + Globals.Score.ToString();
}
```

Poté už jen při načítání nového textu vyžádáme hodnotu z globální proměnné, která nám odpovídá hodnotě ze score z předešlého form a máme hodnotu našeho score v dalším levelu.

A na závěr už jen poslední funkční tlačítko, které nám ukončí hru.

```
1 reference
private void button1_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

Doufám, že každý snad pochopí tuto funkčnost tlačítka už sám :).

4 Závěr

Netuším, co bych dále k závěru dodal. Snad jen to, že u daného projektu jsem se naučil vyhledávat informace, návody a řešení pro mé aktuální problémy. Musel jsem se doučit, jak některé věci fungují a celý projekt na mě působil jako lego, kdy jsem si musel hledat / vytvářet různé bloky kódu ze kterých jsem následně po částech skládal tento projekt. Některé i ve výsledku jednoduché problémy jsem byl schopen na hledání řešení strávit i přes 30 minut svého času, ale euforický pocit při jeho vyřešení na mě působil jako droga a posouval dál. Projekt jsem si ve výsledku i užil a na rozdíl od sepisování lyrických básníků si myslím, že i svým způsobem rozvinulo mé vědomosti a zkušenosti v reálném životě.