

# Ročníkový projekt

**Gymnázium, Praha 6, Arabská 14**  
Arabská 14, Praha 6, 160 00



**Předmět:** Programování

**Téma:** Příběhová textová hra

**Autor:** Vítězslav Procházka

**Třída:** 1.E

**Vyučující:** Mgr. Jan Lána

**Třídní učitel:** Mgr. Blanka Hniličková

## Čestné prohlášení:

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne .....

## Poděkování

Rád bych poděkoval panu profesoru Mgr. Janu Lánovi, který byl vedoucí ročníkového projektu. Dále bych rád poděkoval mému otci Mgr. Jiřímu Procházce, se kterým jsem konzultoval několik problémů v kódu. Velké díky také patří mému spolužákovi Jiřímu Petříkovi, který mi pomohl s používáním programu Overleaf (program ve kterém píše tuto dokumentaci).

## **Anotace:**

V tomto ročníkovém projektu, kde projektovým zadáním byla příběhová textová hra, je předkládán program v programovacím jazyce Java, který umožňuje ovlivňování děje pomocí výběru možností a následně jeho vypisování ve formě textu.

## **Annotation (English):**

In this annual project, in which was the assignment a storytelling text game, is presented a program coded in Java. The program allows you to influence the story by selecting options and then listing you the content in form of text.

## **Zadání projektu:**

Vytvořte textovou příběhovou hru.

### **Upřesněné zadání projektu:**

- napište různé varianty příběhu
- program příběh vypisuje ve formě textu
- hráč pomocí voleb 1 ze 3 (nebo méně, záleží na situaci) možností, typu: „Co uděláš?“ , ovlivní další dění ve hře

### **Bonus (nepovinná část):**

- program umožní hráči podívat se na předchozí část textu
- hra disponuje mapou
- hra má grafické uživatelské rozhraní (nevypisuje jenom do příkazového řádku)

### **Platforma:**

- Java
- JavaFx

# Obsah

<b>Úvod</b>	<b>6</b>
<b>1 Vývojové prostředí</b>	<b>7</b>
1.1 Java	7
1.2 JavaFX	7
1.3 IntelliJ IDEA	7
<b>2 Příběh</b>	<b>8</b>
2.1 Představení světa	8
2.2 Zápletka	9
2.3 Hráčův postup příběhem	9
2.4 Předčasné ukončení příběhu	9
<b>3 Architektura programu</b>	<b>10</b>
3.1 Interakce s grafickým rozhraním	10
3.1.1 Menu	10
3.1.2 Vzhled a ovládání hry	11
3.1.3 Vzhled mapy	12
3.2 Funkcionalita programu	12
3.2.1 Skladování textů	12
3.2.2 Výběr odpovědi	12
3.2.3 Určování jaký text má program vypsát	12
3.2.4 Zobrazení předchozího úseku	13
3.2.5 Zobrazení mapy	14
<b>Závěr</b>	<b>15</b>
<b>Seznam obrázků</b>	<b>16</b>
<b>Reference</b>	<b>16</b>

# Úvod

Předmětem této ročníkové práce je textová příběhová hra, která se jmenuje "Plague Lands" a je naprogramovaná v programovacím jazyce Java a příběh je psaný v českém jazyce. Tato hra obsahuje různé varianty příběhu, které se hráči ukazují v okénku jako text. Na hráče tak čeká série na sebe navazujících textů a pod nimi možnosti jak chce postupovat dále. Podle toho, jakou možnost hráč zvolí, se následně objeví nový text a pod ním nové možnosti co udělat. Takto hra pokračuje, až do dosáhnutí jednoho z různých konců.

Toto téma jsem si zvolil, protože jsem na začátku školního roku netušil, jaké budou mé schopnosti, ovládat programovací jazyk Java, v době tvorby projektu.

Vzhledem k tomu, že jsem nikdy neprogramoval (v žádném jazyce), jsem si zvolil něco, u čeho jsem tak nějak tušil, jak bych to asi řešil. Také mi toto téma bylo doporučeno mým otcem Jiřím Procházkou.

Na doporučení od mého otce jsem si zvolil za vývojové prostředí IntelliJ IDEA. A programátorský jazyk Java, protože se ho učíme ve škole na předmětu programování a žádný jiný neumím.

# 1 Vývojové prostředí

## 1.1 Java

Java je objektově orientovaný programovací jazyk, který vyvinula firma Sun Microsystems a představila 23. května 1995. Výhoda programovacího jazyku Java spočívá v tom, že je možné používat Javu na jakémkoliv operačním systému.<sup>[1]</sup> Podle indexu TIOBE byla Java 3. nepoužívanějším jazykem v březnu 2022.<sup>[2]</sup>



Obrázek 1: Logo Java

## 1.2 JavaFX

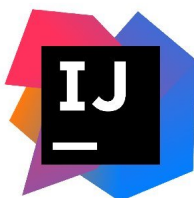
JavaFX je set grafických balíčků, které umožňují tvůrcům programů vytvářet grafické rozhraní pro jejich aplikace.<sup>[3]</sup> Tuto grafickou knihovnu jsem si vybral, protože s ní mám zkušenosti z hodin programování ze školy.



Obrázek 2: Logo JavaFX

## 1.3 IntelliJ IDEA

IntelliJ IDEA je komerční vývojové prostředí pro programování v jazycích Java, Groovy a dalších. Toto vývojové prostředí vytvořila společnost Jet Brains.<sup>[4]</sup> Program si můžete zdarma stáhnout [zde](#).



Obrázek 3: Logo IntelliJ IDEA

## 2 Příběh

### 2.1 Představení světa

Hra je zasazena do světa podobnému našemu středověku, avšak objevují se zde nereálné (fantasy) prvky. Neodehrává se ani na planetě Zemi, ale na planetě jí podobné (podrobněji není popsána). Zde se nachází souostroví různě velikých ostrovů zvané Luční ostrovy. Zde se nachází několik měst, ve kterých se příběh dále odehrává, nejdůležitější z nich je městečko "Hay Lands", protože zde příběh začíná.



Obrázek 4: Mapa Lučních ostrovů



## 2.2 Zápletka

Je to asi půl roku, co zdejší vcelku poklidný život vesničanů a měšťanů narušila záhadná nemoc, která se velmi rychle šíří. Nikdo neví jakým způsobem se šíří nebo léčí, jediné co je všem moc dobře známo, je fakt, že kdo ji chytne tomu není pomoci a do týdne umírá bolestivou smrtí. Pro její typické příznaky ji obyvatelstvo pojmenovalo Hniloba.

## 2.3 Hráčův postup příběhem

Hráč hraje za doktora z Haylands a je právě na něm, aby se vypořádal s Hnilobou. Avšak kam hlavní postavu hráč zavede, je čistě na něm, může se klidně rozhodnout, že žádnou nemoc léčit nebude, jediné omezení (a není malé) leží v mých spisovatelských schopnostech.

Hráč se na začátku hry nachází ve vězení, protože spáchal trestný čin, tento čin byl ale zároveň přínosný pro vědu a mohl sehrát důležitou roli při léčbě hniloby. Je na hráči, aby přesvědčil zdejšího starostu, aby ho pustil a mohl se tak vydat vstříc dobrodružství.

## 2.4 Předčasné ukončení příběhu

Bohužel díky mému špatnému rozvrhnutí času pro práci na ročníkovém projektu a zvolené velké obsáhlosti příběhu, jsem nebyl schopen ani zdaleka příběh dokončit. Tím pádem příběh končí už po části, kdy se má hráč vypravit z cely. Program je tak spíše demoverzí hry než-li plnohodnotnou hrou. Avšak příběh je dostatečně dlouhý, aby demonstroval všechny funkce programu.

## 3 Architektura programu

### 3.1 Interakce s grafickým rozhraním

Grafické rozhraní je vytvořeno pomocí grafické knihovny [JavaFX](#). A většina grafických prvků hry je tvořena v *class* *MainPart*.

Po spuštění programu se otevře čtvercové okno (1000×1000 pixelů), které se nedá zmenšit ani zvětšit (má fixovanou velikost). V tomto okně se odehrává celá hra.

#### 3.1.1 Menu

Jako první se uživateli ukáže startovní menu, které disponuje tématickým obrázkem v pozadí a jedním velkým tlačítkem "start". To po stisknutí spustí hru.

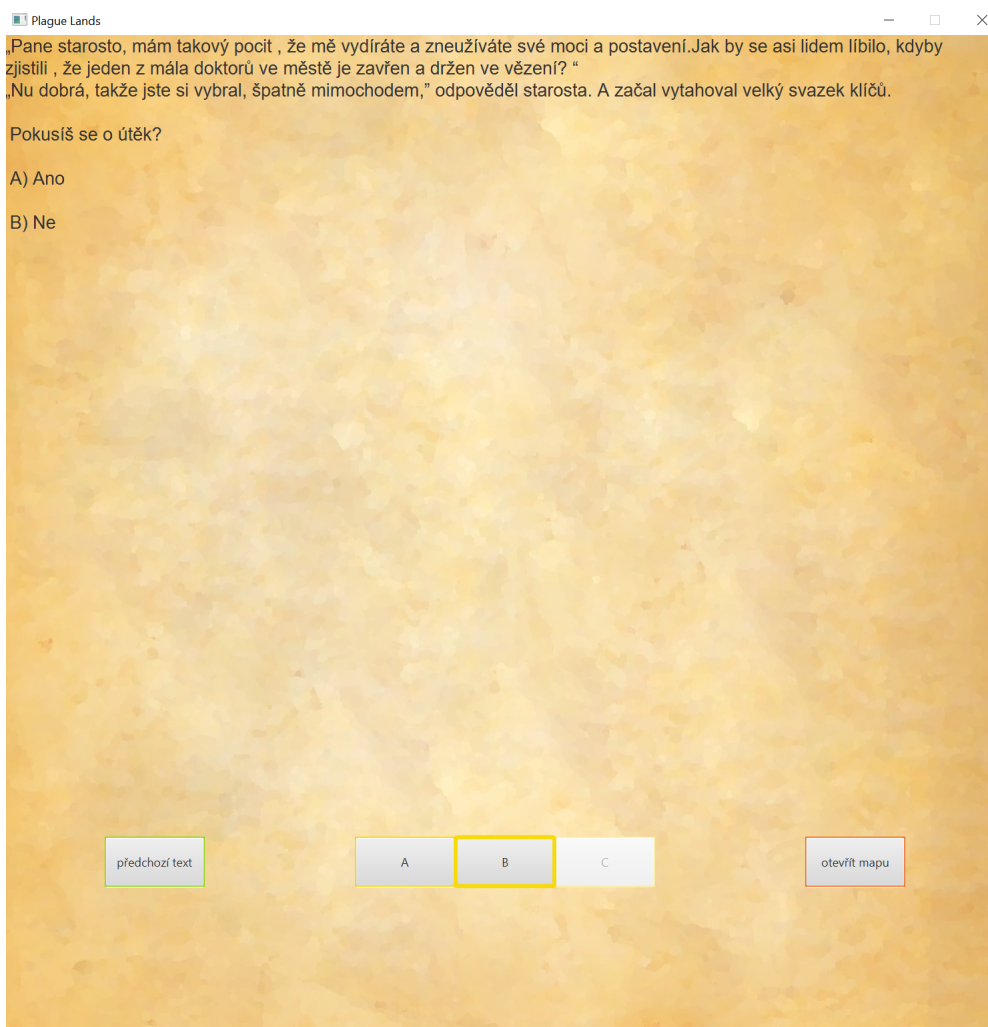


Obrázek 5: Plague Lands menu

### 3.1.2 Vzhled a ovládání hry

Ovládání hry je jednoduché. Ve hře se zobrazí text v horní části okna, vlevo pod ním jsou vypsané dané možnosti co dělat a jsou označeny "A", "B", případně "C". Uprostřed v dolní části okna jsou zobrazeny 3 tlačítka (žlutě ohraničená), každé tlačítko má na sobě napsáno "A" nebo "B" nebo "C". Každé tlačítko reprezentuje právě jednu možnost a po jeho stisknutí se objeví nový text (odpovídající vybrané možnosti) s novými možnostmi postupu. Dále se vpravo na stejné úrovni nachází tlačítko (zeleně ohraničené) s nápisem "předchozí text", které zobrazí text, u kterého se hráč rozhodoval při posledním výběru možnosti a žlutě zvýrazní vybranou možnost, kterou naposledy zvolil, a znemožní kliknutí na tlačítka s výběrem možnosti. Po opětovném stisknutí tlačítka se opět objeví současný text a opět se nám umožní vybrat odpověď. Vpravo od žlutých tlačítek se nachází tlačítko "otevřít mapu" (červeně ohraničené), to otevře obrázek mapy (obrázek 4) v novém okně vpravo na hoře od okna s hrou. Opětovným stisknutím se okno s mapou zavře.

Veškerý vzhled tlačítek je dělán pomocí *CSS Styling*[5].



Obrázek 6: Vzhled hry

### 3.1.3 Vzhled mapy

Mapa byla vytvořena pomocí online softwaru pro tvorbu fantasy map : *Inkarnate*[6]

## 3.2 Funkcionalita programu

### 3.2.1 Skladování textů

Všechny texty jsou skladovány ve formě dlouhých stringů v *hashmapě*[7] *story* v classe *Texts*. Kde je použito jako klíč jednoduché heslo vystihující obsah stringové hodnoty. Spousta hodnot se zdánlivě opakuje, ale vždy je něco pozměněno, aby se text více hodilo do contextu.

### 3.2.2 Výběr odpovědi

Pro výběr odpovědi je vytvořeno pole 3 tlačítek *buttons*. Každé tlačítko dostane funkci, aby zavolalo metodu *processResponse* classy *UserInput* a dalo jí jako parametr pořadí v poli *buttons*.

```
1  for (int i = 0; i < buttons.length; i++) {  
2      int number = i;  
3      buttons[i].setOnAction(event -> userInput.  
        processResponse(number));  
4  }
```

### 3.2.3 Určování jaký text má program vypsát

O vypisování určitého textu se stará metoda *processResponse* classy *UserInput*. V ní se nachází *switch*[8], který určuje pro jaký *gameStatus* a pro jaké číslo odpovědi, má program zavolat metodu *getStory* classy *Texts*, která pod určitým klíčem (který je zadán ve *switchi*) vezme text z *hashmapy* a vypíše ho do hracího okna. Dále se zde také nastaví nový *gameStatus*.

```
5  void processResponse(int a) {  
6  
7      switch (gameStatus) {  
8  
9          case UVOD:  
10             if (a == 0) {  
11                 content.getStory("RozhovorSeStarostou");  
12                 gameStatus = SPAT;  
13             }  
14             if (a == 1) {  
15                 content.getStory("okno");  
16                 gameStatus = OKNO;  
17             }  
18             if (a == 2) {  
19                 content.getStory("dvere");  
20                 gameStatus = DVERE;  
21             }  
22             break;  
23         }  
24     }
```

### 3.2.4 Zobrazení předchozího úseku

Tento problém je řešen na několika místech kódu zároveň. Když hráč zvolí nějakou možnost, program si zapamatuje, jaký klíč použil minule a jaký klíč nyní, to se stane v metodě *getStory*. Při dalším výběru možnosti program ze současného klíče udělá minulý a nastaví nový současný klíč.

```
25 public void getStory(String name) {
26     Mainpart.text.setText(story.get(name));
27     UserInput.previousAnswer = UserInput.currentAnswer;
28     UserInput.currentAnswer = name;
29 }
```

V calsse *Texts* se nachází metoda, která nemění současný ani minulý klíč, jenom vypíše text do hracího okna, jmenuje se *writeStory*.

```
30 public void writeStory(String name){
31     Mainpart.text.setText(story.get(name));
32
33 }
```

Tlačítko pro vypsání minulého textu zavolá metodu *showPreviousAnswer*, která zavolá *writeStory* s minulým klíčem pokud program nyní ukazuje současný text, v tomto případě dále žlutě zvýrazní vybranou odpověď a znemožní kliknutí na tlačítka pro výběr možností. A v opačném případě : kdy program ukazuje text s minulým textem, program zavolá "writeStory" se současným klíčem a opět umožní kliknutí na tlačítka.

```
34 void showPreviousAnswer() {
35
36     if (isShowingPrevAns ) {
37         content.writeStory(currentAnswer);
38         previousButton.setText("minuly text");
39         for (int i = 0; i < buttons.length; i++){
40             buttons[i].setStyle("-fx-border-color: #F7DA09;
41                                 -fx-faint-focus-color: #F7DA09; -fx-focus-
42                                 color: transparent");
43         }
44         for(int x = 0; x < buttons.length - buttonsDisabled;
45             x++ ) {
46             buttons[x].setDisable(false);
47         }
48         isShowingPrevAns = false;
49     }else {
50         content.writeStory(previousAnswer);
51         previousButton.setText("soucasny text");
52         buttons[buttonSelected].setStyle("-fx-background-
53             color: #F7DA09;-fx-border-color: #000000; ");
54         for (int x = 0; x < buttons.length; x++) {
55             buttons[x].setDisable(true);
56         }
57         isShowingPrevAns = true;
58     }
59 }
```

### 3.2.5 Zobrazení mapy

Mapa má pro sebe vytvořené své vlastní okno, které se otevře když, hráč stiskne tlačítko "otevřít mapu". To má přidělenou funkci zavolat metodu *showMap* classy *UserInput* a dostane jako parametr současné souřadnice hlavního okna. To je kvůli tomu, aby se mapa vždy otvírala vpravo nahoře od hlavního okna. [9]

```
56 mapButton.setOnAction(event -> userInput.showMap(stage.getX(),  
    stage.getY()));
```

V metodě *showMap* už je to jednoduché, je zde jednoduchý mechanismus, který okno s mapou zobrazí nebo ho zavře, pokud je nyní otevřené. To je udělané pomocí metody *"isShowing"* [10]

```
57 void showMap(double x, double y) {  
58  
59     newStage.setX(x + 1000);  
60     newStage.setY(y);  
61  
62     if (newStage.isShowing()) {  
63         newStage.close();  
64         mapButton.setText("otevrit mapu");  
65     } else {  
66  
67         newStage.show();  
68         mapButton.setText("zavrit mapu");  
69     }  
70 }
```

## Závěr

Zadání mého ročníkového projektu jsem splnil jen částečně. Hra je sice z hlediska programové části plně hratelná a funkční, ale je zde problém s nedostatkem obsahu (obsahu).

Nestihl jsem svou vizi příběhu ani zdaleka sepsat. Stalo se to kvůli mému špatnému rozvrhnutí času pro vypracování této části ročníkového projektu. Příště si musím vymezit více času pro tvorbu obsahu nebo si musím vymyslet rychleji plynoucí příběh a nerozepisovat ho tolik do detailů.

## Seznam obrázků

1	Logo Java . . . . .	7
2	Logo JavaFX . . . . .	7
3	Logo IntelliJ IDEA . . . . .	7
4	Mapa Lučnických ostrovů . . . . .	8
5	Plague Lands menu . . . . .	10
6	Vzhled hry . . . . .	11

## Reference

- [1] wikipedia.org. Java (programovací jazyk). January 2022. [https://cs.wikipedia.org/wiki/Java\\_\(programovac%C3%AD\\_jazyk\)](https://cs.wikipedia.org/wiki/Java_(programovac%C3%AD_jazyk)).
- [2] © 2022 TIOBE Software BV. index tiobe. January 2022. <https://www.tiobe.com/tiobe-index/>.
- [3] Monica Pawlan. What is javafx? April 2013. <https://docs.oracle.com/javafx/2/overview/jfxpub-overview.htm#A1104465>.
- [4] JetBrains. Why intellij idea. 2022. <https://www.jetbrains.com/idea/>.
- [5] Jakob Jenkov. Javafx css styling. May 2016. <https://jenkov.com/tutorials/javafx/css-styling.html>.
- [6] Inkarnate - online map making tool. <https://inkarnate.com>.
- [7] w3schools. Java hashmap. [https://www.w3schools.com/java/java\\_hashmap.asp](https://www.w3schools.com/java/java_hashmap.asp).
- [8] w3schools. Java switch. [https://www.w3schools.com/java/java\\_switch.asp](https://www.w3schools.com/java/java_switch.asp).
- [9] Slaw. How to get the x and y location of a javafx node for displaying a popupcontrol - answer. 2020. <https://stackoverflow.com/questions/59760124/how-to-get-the-x-and-y-location-of-a-javafx-node-for-displaying-a-p>
- [10] Slaw. Check if a stage is already open before open it again javafx - answer. October 2018. <https://stackoverflow.com/questions/52693199/check-if-a-stage-is-already-open-before-open-it-again-javafx>.