

**Univerzita Hradec Králové**

**Přírodovědecká fakulta**

**Katedra kybernetiky**

Interaktivní herní prostředí založené na dynamickém  
webu jako prostředek gamifikace výuky

Bakalářská práce

Autor: Jan Mottl

Studijní program: B1801 / Informatika

Studijní obor: Informatika se zaměřením na vzdělávání –  
Historie se zaměřením na vzdělávání

Hradec Králové

duben 2019

Prohlášení:

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a že jsem v seznamu použité literatury uvedl všechny prameny, z kterých jsem vycházel.

V Hradci Králové dne 26. dubna 2019

Jan Mottl

Chtěl bych poděkovat v první řadě svému vedoucímu práce doktoru Michalu Musílkovi, který vedl mou práci svědomitě a vzorně, abych dosáhl jejího dokončení. Tímto mu děkuji za veškeré konzultace, na kterých byl vždy otevřený k řešení všech problémů, které se během vývoje práce naskytly.

Dále bych chtěl poděkovat svému konzultantovi Mgr. Ondřeji Krahulcovi, že jsem mohl přispět svým dílem do jeho projektu Agama, a snad mnou vytvořené aplikace budou hojně využívány k vzdělávání.

Nakonec bych chtěl poděkovat Janu Kudláčkovi, který mi byl rádcem při programování a díky němuž jsem pronikl hlouběji do této tematiky.

## **Anotace**

MOTTTL, J. *Interaktivní herní prostředí založené na dynamickém webu jako prostředek gamifikace výuky*. Hradec Králové 2019. Bakalářská práce na Přírodovědecké fakultě Univerzity Hradec Králové. Vedoucí bakalářské práce Michal Musílek. 58 s.

Práce se věnuje tématice využití didaktických her, zejména simulačních a diskuzních v edukačním prostředí a s tím spojenou gamifikací výuky; dále formátem těchto her, zpracovaných do podoby webových aplikací a nástrojů nutných k jejich vytváření. Praktická část práce je zaměřená na vytvoření dvou webových aplikací k didaktickým hrám a jednoho webového rozcestníku. Obojí spadá pod projekt Agama, jehož autorem je konzultant práce, Mgr. Ondřej Krahulec z Masarykovy univerzity v Brně. Popsán je vývoj a funkce webového rozcestníku, a hlavně obou aplikací, které byly následně testovány na půdě Masarykovy univerzity.

### **Klíčová slova:**

Didaktické hry, Webové aplikace, Gamifikace

## **Annotation**

MOTTTL, J. *An interactive gaming environment based on the dynamic web as an instrument of gamification in education*. Hradec Králové 2019. Bachelor Thesis at Faculty of Science University of Hradec Králové. Thesis Supervisor Michal Musílek. 58 p.

Thesis focuses on the use of didactic games, especially simulation and discussion, in the education and the gamification of teaching. Furthermore, the format of these games, processed into web applications and the tools needed to create them. The practical part is focused on creating two web applications for didactic games and one web guidepost. Both of them belong to the project Agama, whose author is the thesis consultant, Mgr. Ondřej Krahulec from Masaryk University in Brno. It describes the development and function of the web guidepost, and especially both applications, which were subsequently tested at Masaryk University.

### **Keywords:**

Education game, Web apps, Gamification

# Obsah

Úvod .....	8
2 Základní vhlad do pojmu „Didaktická hra“ .....	9
2.1 Simulační hry .....	9
2.2 Diskuze jako nástroj her (Diskuzní hry) .....	9
3 Gamifikace .....	11
4 Webové aplikace jako edukační prostředek .....	12
5 Představení her a rozcestníku .....	13
5.1 Rozcestníkový web .....	13
5.2 People v. Johnson .....	13
5.3 Grimfield 1665 .....	13
6 Jazyky .....	15
6.1 HTML .....	15
6.2 CSS .....	16
6.3 JS .....	17
6.4 PHP .....	17
6.5 SQL .....	17
7 Sady nástrojů (Frameworks) .....	18
7.1 Bootstrap .....	18
7.2 jQuery .....	18
7.3 Nette Framework .....	18
8 Programy použité pro tvorbu .....	19
8.1 PhpStorm .....	19
8.2 Git a Bitbucket .....	19
8.3 PhpMyAdmin .....	19
8.4 WampServer .....	19
9 Objektově orientované programování .....	20
10 Mechaniky her .....	22
10.1 People v. Johnson .....	22
10.2 Grimfield 1665 .....	22
11 Zpracování rozcestníkového webu .....	25
11.1 Responzivita .....	25

11.2	Dlaždice s hrami .....	25
12	Souborová struktura webových aplikací .....	26
13	Zpracování webové aplikace pro hru People v. Johnson .....	28
13.1	Přihlašovací systém .....	28
13.2	Generátor klíčů.....	28
13.3	Časovač .....	29
13.4	Nový důkaz v průběhu hry .....	30
13.5	Rozsudek.....	31
13.6	Kontrola postupu a přesměrování.....	32
13.7	Statistika případů.....	32
14	Zpracování webové aplikace pro hru Grimfield 1665.....	33
14.1	Systém kol .....	33
14.2	Časovač .....	33
14.3	Stav města.....	34
14.4	Průběh kola.....	35
14.5	Možnost přeskočení kol .....	36
14.6	Výsledek hry .....	36
14.7	Kontrola postupu a přesměrování.....	37
14.8	Statistika obyvatel .....	37
15	Ukázky kódu .....	38
15.1	Přesměrování u hry Grimfield 1665 .....	38
15.2	Odeslání formuláře (ukončení kola) ve hře Grimfield .....	38
15.3	Statistika případu u hry People v. Johnson.....	39
16	Testování obou aplikací .....	40
16.1	Zpětná vazba na aplikaci pro hru People v. Johnson .....	40
16.2	Zpětná vazba na aplikaci pro hru Grimfield 1665.....	41
17	Instrukce k elektronické příloze práce .....	42
	Závěr .....	43
	Seznam použité literatury a zdrojů .....	44
	Seznam příloh .....	46
	Příloha č. 1 – Diagram mechaniky Pvj .....	47
	Příloha č. 2 – Diagram mechaniky Grimfield.....	48
	Příloha č. 3 – Diagram mechaniky Grimfield (2) .....	49

Příloha č. 4 – Souborový strom PvJ .....	50
Příloha č. 5 – Relační model databáze pro PvJ.....	51
Příloha č. 6 – Relační model databáze pro Grimfield .....	52
Příloha č. 7 – Ukázka kódu (Přesměrování) .....	53
Příloha č. 8 – Ukázka kódu (Konec kola).....	54
Příloha č. 9 – Ukázka kódu (Statistika případů) .....	55
Příloha č. 10 – Vzor formuláře zpětné vazby .....	56
Příloha č. 11 – Grafy výsledků zpětné vazby PvJ .....	57
Příloha č. 12 – Grafy výsledků zpětné vazby Grimfield.....	58

## Úvod

Byl to Jan Ámos Komenský, který propagoval dnes již známé heslo „škola hrou“, a i když nás od jeho doby dělí několik staletí, je toto heslo stále platné a v poněkud pozměněném chápání snadno aplikovatelné na současnost.

Jak se dají didaktické hry využít jako edukační nástroj? Jaké klady a zápory můžeme najít na simulačních a argumentačních hrách? A jaké mají využití na poli vzdělávání při současném využití informačních technologií? To jsou otázky, které jsem si pokládal a na které se pokouším ve své práci odpovědět.

Téma, které jsem zvolil, se orientuje na možnost využití simulačních a argumentačních her v softwarovém prostředí v online platformě webu. Díky tomu se edukační materiál dostává všem osobám, které jsou vybaveny potřebnou technologií a možnostmi připojení k internetu. Hry, které zpracovávám a vytvářím k nim dynamické webové prostředí, vytvořil Mgr. Ondřej Krahulec v rámci své diplomové práce na Masarykově univerzitě v Brně. (Krahulec, 2016) Tyto hry byly vytvořeny v rámci jeho edukačního projektu s názvem „Agama“.

Teoretická část si klade za cíl osvětlit základní využití didaktických her ve výuce se zaměřením na hry simulační a argumentační; jako příklady jsou použity převážně hry vytvořené Mgr. Krahulcem. (Krahulec, 2016) K vypracování byla použita rešeršní kompilační metoda, která prozkoumává různé zdroje a informace z nich porovnává mezi sebou.

Dále je věnován prostor samotnému vývoji jednotlivých her jako webových aplikací. Jde o přehled a základní informace o technologiích, ve kterých byly aplikace vytvářeny, a použitých programovacích jazycích. Popsán je i samotný proces vývoje s řešením jednotlivých problémů, se kterými jsem se musel potýkat.

Praktická část zahrnuje vytvořené webové aplikace a webového rozcestníku pro dvě hry. V této části jsou také rozebrány některé části kódu, aby na nich byla ukázána samotná funkce jednotlivých aplikací, konkrétní funkcionality jednotlivých aplikací a jejich vnitřní procesy.



## **2 Základní vhled do pojmu „Didaktická hra“**

Didaktické hry jsou společenské, nebo pohybové hry, které si kladou za cíl působit edukačně, či procvičovat dovednosti studenta během průběhu hry. Za didaktické hry lze považovat nejen různé karetní a deskové hry, ale i videohry, ovšem pouze takové, které jsou zaměřené zejména na logiku, představivost, nebo mají ráz seznamovací, kdy jsou studentovi přiblíženy určité pojmy a jevy. Takové hry mají předem daná pravidla, na která ve školním prostředí dohlíží nejčastěji učitel. (Skořepová, 2011)

Typů her je velké množství, stejně jako různých přístupů k jejich typologii. Přikláním se k typologii podle Eduarda Bakaláře: hry se dělí na seznamovací a rozehrívací, tvořivé, přiřazovačky a kvízy a diskuzní hry. Zde se zaměřuji primárně na hry diskuzní (argumentační) a na hry simulační. (Skořepová, 2011)

### **2.1 Simulační hry**

Simulační hry mají za cíl seznámit hráče s určitou problematikou, nebo vyzkoušet jeho schopnosti pomocí simulace nějaké situace, nějakého prostředí či konfliktu. Klíčové pro simulační hry, také známé jako RPG (z angl. Role-playing Games, rolové hry). Je rozdíl mezi hráčem, rolí a postavou. „Role“ je uměle vykonstruovaný charakter, do kterého se během hry hráč vcítí, a bude v ní průběh hry prožívat, chovat se tak, jako by danou rolí doopravdy byl. Například: student Jan bude během hry o Francouzské revoluci představovat „rolí“ republikánského vojáka Pierra. „Postava“ je termín, kterým se označuje už hráč v dané roli. Pokud tedy řekneme, že „postava“ Pierra se během hry nějak zachovala, mluvíme o tom, jak se student Jan zachoval v „rolí“ Pierra. (Šafarčíková, 2011)

Hry přináší hráčům rozvoj komunikačních schopností a schopnosti hledat řešení, rozvíjí představivost, mezilidské vztahy a empatii hráče, dále hráči procvičují vyjednávání a uplatňují tvůrčí myšlení. (Šafarčíková, 2011)

Příkladem hry s přínosem pro hráče je i hra „People vs. Johnson“ od Mgr. Krahulce (Krahulec, 2016). Ve hře se hráč dostane do role porotce, který jako jeden z několikačlenné poroty rozhoduje o vině či nevině obžalovaného. Hráč během této hry rozvíjí při debatě se zbytkem poroty schopnosti argumentační a diskuzní. Zároveň je zde rozvíjeno tvůrčí myšlení a schopnost řešení problémů, které se uplatní při prozkoumávání důkazů, které k případu porota má. (Šafarčíková, 2011)

### **2.2 Diskuze jako nástroj her (Diskuzní hry)**

Diskuzi je možné začlenit jako aktivizační metodu ve výuce. Diskuzní metoda navazuje na metodu rozhovoru, ten je ovšem veden vždy mezi dvěma subjekty. Diskuzní metody naproti tomu nutí k spoluúčasti celý kolektiv. Mezi diskuzní metody se řadí nejen samotná diskuze, ale i brainstorming a další. Zde se zaměřuji na diskuzi jako klíčový prvek her, které lze pojmenovat jako diskuzní či argumentační. (Nováková, 2014)

Diskuze samotná nutí studenty formulovat svůj názor na dané téma, které diskuze řeší. Je proto důležité volit téma, ke kterému mohou studenti zaujmout jasný postoj. Diskuze musí být oproti dialogu více vedená ze strany učitele, nebo celkového nastavení rámce diskuze. (Nováková, 2014)

Ve hře „Grimfield 1665“ se studenti dostávají do rolí městské rady. Jejich hru tvoří učinění několika závažných rozhodnutí o osudu města Grimfield, ve kterém vypukl mor. Každé rozhodnutí musí rada projednat, přičemž je využita diskuze jako prostředek pro postup ve hře. K další části postoupí hráči jenom komparací jednotlivých názorů a následným finálním rozhodnutím. (Krahulec, 2019a)

### 3 Gamifikace

Ve spojení s didaktickými hrami se často objevuje pojem „gamifikace“, což lze volně přeložit jako „vkládání herních prvků a mechanik do neherního prostředí“. Tyto mechaniky lze rozčlenit do osmi aspektů, které by měla obsahovat každá hra. Tím základním a již zmíněným jsou pravidla. Dalším aspektem je cíl: každá hra měla by mít své zakončení, které přináší vyhodnocení. Zároveň díky cíli lze monitorovat postup hráče, což je pro hru a jejího hráče důležité. Po ukončení hry by měl přijít další aspekt, a to zpětná vazba. Tu by měla hra podat hráči ve vazbě na dosažené cíle. Hráč by měl dále mít možnost projít vlastní chyby, aby měl motivaci se ve hře zdokonalovat. Následuje funkce systému odměn, nejčastěji užívaná je forma odznáček, bodů, či tzv. „levelů“ (úrovní), které hráče odměňují za postup. Hra sama by nemohla existovat bez aspektu řešení problémů, které by mělo být výzvou pro samotného hráče. K řešení problémů se hráč dostává prostřednictvím příběhu, který by měl hráče lákat k dalšímu postupu a motivovat jej k dosažení cíle či vyřešení dílčího problému. Hra by měla na rozdíl od reálného světa poskytovat bezpečné prostředí, ve kterém lze neuspět. Často právě prohra nutí hráče k lepším výsledkům při opakovaném hraní. Hráč by měl cítit smysl pro mistrovství a snahu se zdokonalovat ve svém výkonu. Posledním aspektem je sám hráč, bez kterého by pochopitelně hra nemohla fungovat. (Manas, 2015)

Základní aspekty gamifikace jmenuji výše, hlubší pohled do tohoto pojmu následuje dále. Je otázkou, proč se vůbec snažit uplatnit herní mechaniky ve vzdělávacím procesu. Odpověď je jednoduchá, ze hry si často žáci odnesou více informací než při použití jiných didaktických prostředků, jako je například výklad. Herní mechaniky více motivují žáka rozvíjet své schopnosti, aby dosáhl výsledku ve hře. Zároveň při častém setkání žáků s herními mechanikami skrz různé stolní či počítačové hry je pro žáky snazší si tyto mechaniky osvojit a pracovat s nimi. (Kiryakova, 2013)

Pro správné využití gamifikačních metod je nejdříve třeba determinovat cíl procesu, tedy, co chceme, aby si žák osvojil nebo jakou dovednost rozvinul. Zároveň je nutné myslet na to, že je nezbytné brát v potaz již nabyté schopnosti žáka a nevytvářet něco, k čemu by neměl předpoklady. Dál je třeba zamyslet se nad zapojením jednotlivých aspektů gamifikace (viz výše). V rámci tohoto je nutná inkluze samostatné práce žáků, dále práce s učitelem či vhodným studijním materiálem; důležitá je i práce ve skupině s ostatními žáky. (Kiryakova, 2013)

## 4 Webové aplikace jako edukační prostředek

Pojem, jenž zaštiťuje používání online prostředí pro výuku, je e-learning, který lze definovat tak, že všechen vyučovací text je dostupný na internetu a dává žákovi automatickou zpětnou vazbu. (Devedzic, 2006)

Pro správné využití edukace pomocí webu je důležité dodržet několik základních pravidel. Je důležité mít jasně daný cíl, kterého chceme dosáhnout, neboli informace či schopnosti, které si žák má během procesu osvojit. Je nutné zohlednit hardwarové možnosti žáka, aby web zvládl otevřít doma i ve škole a používat všechny jeho části.

Dále je nutné jednotlivé materiály rozdělit do takové formy, kdy se k nim žák dostává postupně a není přehlcen informacemi. Velmi dobrý způsob je nabídnout žákovi možnost nahlédnout do materiálů a nechat jej určitou dobu studovat, a pak přistoupit k dalšímu kroku. Tím může být například webová aplikace, která dané téma procvičí. Aby byla aplikace a web celkově efektivní, musí být dobře vizuálně navržený. To znamená, že musí navigace po jeho částech být snadná a zároveň je třeba, aby web dobře vypadal na různých obrazovkách (nejlépe, aby byl kompletně responzivní). (Devedzic, 2006)

Důležité funkce edukace skrz webové prostředí jsou, v první řadě, aby mohl být subjekt žáka a učitele oddělen na jakoukoliv fyzickou vzdálenost, ale edukační proces mohl probíhat: žák se vzdělává a nemusí být ve škole, stačí aby byl připojen k internetu. Dále je důležitá možnost monitorování instrukcí a informací, ke kterým se žák dostane. Poslední důležitou funkcí je využití informačních technologií v edukačním procesu a možnost okamžité zpětné vazby pomocí algoritmů určených k vyhodnocení aktivit. (Devedzic, 2006)

Mezi výhody e-learningu se řadí možnost provádět edukační proces kdekoli a kdykoli a možnost usnadnit zapojení žáků, kteří se bojí se veřejně projevit. Online prostředí navíc často zamezí dominantním studentům převzít veškerou iniciativu v procesu skupinové výuky. (Devedzic, 2006)

Naopak mezi zápory lze řadit skutečnost, že informace poskytnuté na webu, nikdy nebudou pro žáky detailnější a přístupnější v porovnání s informacemi předanými prostřednictvím dialogu s učitelem. Žák nemusí vždy porozumět všem instrukcím a jen kvůli tomu selhat při práci s aplikací, i když dané znalosti, či schopnosti měl. (Devedzic, 2006)

## 5 Představení her a rozcestníku

V této části představuji jednotlivé části své práce, webový rozcestník a dvou her jako webových aplikací. Jde jen o krátké představení těchto konceptů, ne o jejich zpracování, kterému se budu věnovat později.

Celý projekt nesoucí jméno „Agama“ slouží k učení anglického jazyka pomocí simulačních a diskuzních her. Jeho autorem je výše zmíněný Mgr. Ondřej Krahulec a vznikl pod záštitou Masarykovy univerzity v Brně, primárně jako diplomová práce. (Krahulec, 2016) Proto jsou obě webové aplikace v angličtině. Rozcestník je v obou jazycích (anglickém a českém), mezi kterými lze přepínat.

### 5.1 Rozcestníkový web

Rozcestníkový web by měl sloužit k celému projektu didaktických her, které vytvořil Mgr. Ondřej Krahulec. Rozcestníkový je nazýván proto, že kromě informací o celém projektu a možnosti stáhnout pedagogické materiály si tento web klade za cíl ukázat nabídku her a jejich anotace tak, aby se uživatel mohl rovnou přesunout na stránku hry.

Kromě již zmíněného se na webu objeví informace o autorovi tohoto projektu, Mgr. Krahulcovi.

### 5.2 People v. Johnson

Jedná se o hru, kdy se hráči vžijí do kůže poroty v roce 1908, která má rozhodnout o rozsudku nad Mr. Johnsonem. Ten se měl, dle tvrzení obžaloby, dopustit vraždy Petera J. Wilsona. (Krahulec, 2019b)

Na webu si hráč může přečíst základní instrukce k hraní, pak přejít ke zkoumání jednotlivých důkazních materiálů. Nacházejí se zde zprávy z novin, fotky z místa činu, poznámky vyšetřujícího detektiva a zpráva od koronera z pitvy Mr. Wilsona. Následně si hráč může pustit výpovědi dvou svědků a samotného obžalovaného Mr. Johnsona. Na stránkách je umístěna také část s vhodnou slovní zásobou pro hraní hry. (Krahulec, 2019b)

Hra samotná musí trvat alespoň 45 minut, během kterých porota diskutuje o vině či nevině obžalovaného. Po 30 minutách je porotě doručen další důkaz, kterým je zpráva od jednoho mladého detektiva. Tento důkaz je klíčový pro celý případ a až ve chvíli, kdy se nad ním porota poradí, může vynést verdikt. Tento důkaz v herním prostředí existuje ve dvou verzích a jen jedna je dána porotě. Každá verze důkazu mění výsledek hry, který porota shlédne formou videa. (Krahulec, 2019b)

### 5.3 Grimfield 1665

Hra Grimfield přenesení hráče do roku 1665, kdy jako členové městské rady města Grimfield v Anglii čelí velké hrozbě. Hned v úvodních instrukcích a po shlédnutí

úvodního videa je hráči představen hlavní problém, a to, že se v městě začal šířit mor. (Krahulec, 2019a)

Na stránkách si hráč nejdříve musí přečíst informace o době samotné a o městě Grimfield. Informace o moru se dozví skrz video. Poslední, co je nutné před samotným hraním, je načerpat vhodnou slovní zásobu, která se taktéž nachází na webu. (Krahulec, 2019a)

Do hry je možné přidat další faktor, kdy se mezi hráče rozdělí jednotlivé role členů městské rady. Některá rozhodnutí jsou potom vázaná na konkrétní postavu nebo jsou různě důležitá pro různé role. Když se tento rolový prvek zapojí, může být hra pestřejší a některá rozhodnutí pro dané členy rady o to těžší. (Krahulec, 2019a)

Hra samotná je rozdělena do několika kol (jedenácti, z čehož lze dvě přeskočit), ve kterých je vždy hráčům představen problém a oni mají jen omezenou možnost, jak jej vyřešit. (Většinou jde o dvě řešení v některých případech je možností více). Na jednotlivá kola mají hráči doporučeno sedm minut a na celou hru okolo hodiny. Konec hry ovlivní jen hráčská rozhodnutí, výsledek se hráči dozvědí formou videa. Konec jsou pouze dva, buď se mor rozšíří do okolí, nebo ne. Variabilní je počet přeživších, na začátku hry je v městě 1 000 obyvatel a různá rozhodnutí v různých kolech navyšují počty obětí. (Krahulec, 2019a)

## 6 Jazyky

V této části popisují jazyky, které jsem použil při vývoji jednotlivých částí projektu. Ukázky jazyků pochází ze stránky W3schools (W3schools, 2019).

### 6.1 HTML

Hypertext Markup Language, zkráceně HTML, je značkovací jazyk pro tvorbu webových stránek a tvoří základní kámen vývoje webové stránky. V rámci vývoje byla použita verze HTML 5, což je již pátá verze tohoto jazyka (vydaná v roce 2014). Velmi jednoduše popsáno se jedná o seznam značek, které si následně webový prohlížeč přeloží a vykreslí podle nich stránku. Lze k němu připojit design skrz CSS (Cascading Style Sheets, kaskádové styly), stejně jako dynamické prvky webu se píše v JavaScriptu (dále JS). JS se také propojuje s HTML. (W3C: HTML and CSS, 2019)

Jak je patrné v ukázce, HTML je velice jednoduchý jazyk, ve kterém se tvoří základní kostra stránky. V základu je rozdělen na „head“ a „body“, kdy v „head“ jsou důležité informace o stránce, jako její titulní název, který můžeme vidět v záložce prohlížeče. Samotný text stránky se píše do „body“, kde můžeme vidět použití dvou základních značek. V „h1“ je napsaná nadpis. Naproti tomu v „p“ je napsaný text odstavce. Všechny párové značky v HTML musí mít svou počáteční a konečnou značku, aby fungovaly správně.

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

*Ilustrace 1: Ukázka HTML kódu*

## 6.2 CSS

Cascading Style Sheets, zkráceně CSS je jazyk pro určení stylu, psaný, podobně jako HTML, ke kterému se CSS připojuje, pomocí značek. Nejnovější verzí je CSS3. CSS popisuje design jednotlivých prvků uvedených v HTML. Lze přímo určovat styl jednotlivým prvkům obecně popsáných pomocí HTML, například pro odstavce (značka „p“). Dalším způsobem je vytvoření buď tzv. „class“, kterou můžeme připojit k více prvkům, nebo „id“, jež by měl mít každý prvek unikátní. Chceme-li změnit pouze jeden odstavec na celé stránce, připišeme mu v HTML id, na které se pak v CSS odkážeme. Pokud naopak chceme jeden styl uplatnit pro více prvků, tak použijeme class. (W3C: HTML and CSS, 2019)

```
body {
  background-color: lightblue;
}

h1 {
  color: white;
  text-align: center;
}

#para1 {
  text-align: center;
  color: red;
}

.center {
  text-align: center;
  color: red;
}
```

*Ilustrace 2: Ukázka CSS*

Funkci CSS si jasněji ukážeme na ukázce kódu. Vidíme syntax CSS a jakým stylem přidává designové prvky k HTML. To, co je napsáno v „body“, je styl, který platí pro všechny prvky, které jsou v HTML napsané mezi značkami „body“. Obdobně platí, že styl napsaný v „h1“ je uplatňuje pro všechny nadpisy stupně jedna, které jsou v HTML napsány.

To jsou obecné selektory, existují také selektory konkrétní; „#para1“ vkládá svůj styl pouze jednomu prvku v HTML, který má u sebe uvedené „id=„para1““, neboli jedná se určování stylu pomocí id. Druhá možnost je určení stylu pomocí třídy, což je v CSS označováno pomocí tečky. Styl napsaný v „.center“ budou mít všechny prvky v HTML, které mají uvedeno „class=„center““.



### **6.3 JS**

JavaScript, zkráceně JS, je programovací jazyk, který je objektově orientovaný. JavaScript se stejně jako CSS připojuje k HTML, které pak volá jeho skripty. Díky JS lze naprogramovat dynamické prostředí na webu. Zjednodušeně je možné říci, že skript v JS může být navázaný například na tlačítko; ve chvíli, kdy jej uživatel zmáčkne, se spustí skript – může například změnit barvu textu, či odhalit skrytý obrázek. (W3C: JavaScript Web APIs, 2019)

### **6.4 PHP**

Hypertext Preprocessor, zkráceně PHP, je programovací jazyk pracující ve formě skriptů. Využívá se při vytváření dynamického webu. Zjednodušeně řečeno jde o psaní skriptů, které se generují do HTML. Všechny skripty se provádí na straně serveru a uživatel vidí až jejich výsledek, což zaručuje například to, že uživatel nemůže nahlížet do samotného kódu, což u stránek psaných pouze pomocí HTML lze. Rozdílné je, že HTML stránky jsou pouze statické, zatímco PHP generuje dynamicky obsah do HTML. (PHP, 2019)

### **6.5 SQL**

Structured Query Language zkráceně SQL, je dotazovací jazyk, který slouží pro komunikaci s relační databází. K takové databázi se lze připojit pomocí PHP a slouží k trvalému uchování informací na serveru, na kterém databáze běží. (MySQL, 2019)

Databáze se skládá z tabulek, SQL slouží k práci s daty v těchto tabulkách. Hry vytvořené v této práci využívají databázi a komunikaci s ní nejen pro ukládání výsledků her, ale i pro jejich samotnou funkčnost. V databázích je také veden přihlašovací systém uživatelů ke hrám. (MySQL, 2019)

## 7 Sady nástrojů (Frameworks)

V této části se věnuji softwarovým strukturám, které obsahují podpůrné programy a knihovny, a urychlují práci.

### 7.1 Bootstrap

Bootstrap je framework, který se používá jako sada nástrojů pro tvorbu webu. Obsahuje rozsáhlou knihovnu, která urychluje práci s HTML, CSS i JS. Je zaměřený na vytváření responzivního webu. V praxi například při použití v CSS má Bootstrap připravené svou knihovnu plnou class, kterou lze přímo aplikovat na HTML. Není třeba definovat v CSS margin (okraje), stačí pouze do class dopsat `.mr-3`, čímž vznikne „margin-right: 1rem“. Díky této sadě je možné ušetřit značné množství času. Ve svém projektu jsem ji použil hlavně při tvorbě rozcestníkového webu. (Bootstrap, 2019)

### 7.2 jQuery

jQuery je podobně jako Bootstrap sada nástrojů. Jedná se o javascriptovou knihovnu, která je zaměřená na dobré propojení JS a HTML. Ulehčuje psaní skriptů v JavaScriptu a práci s ním. (jQuery, 2019)

### 7.3 Nette Framework

Nette Framework slouží k ulehčení tvorby PHP. Latte je potom šablonovací systém součástí Nette. Právě Latte bylo využito při tvorbě obou her v rámci tohoto projektu. (Nette: latte, 2019)

## 8 Programy použité pro tvorbu

V této části budou popsány programy, které byly použity pro vytvoření webových stránek a aplikací.

### 8.1 PhpStorm

PhpStorm je program určený pro psaní webových stránek v různých jazycích. Program pomáhá uživateli formou napovídání či hledání chyb a varování. V tomto programu byly napsány všechny webové stránky a aplikace v této práci. Program není open source, ale má zdarma licence pro studenty. (Jetbrains: PhpStorm, 2019)

### 8.2 Git a Bitbucket

Git je systém spravování verzí a týmovou spoluprací, který lze doporučit užívat nejen při vytváření webových aplikací, ale během jakéhokoliv projektu. Jedná se o možnost ukládat verze projektu po částech, a tak se může vždy vývojář vrátit ke starší verzi svého projektu. (Git, 2019)

Bitbucket je internetové úložiště pro Git. Jednotlivé verze se tak dají spravovat online s více uživateli. Jak pro rozcestníkový web, tak pro obě aplikace jsem vytvořil online repozitář právě na Bitbucketu. (Bitbucket, 2019)

### 8.3 PhpMyAdmin

Jedná se o free software na spravování relačních databází. Nabízí jednoduché, ale velmi užitečné prostředí, ve kterém lze prohlížet databáze. Zároveň se dají takto databáze snadno vykopírovat do SQL kódu. Databáze připojené k oběma aplikacím byly spravovány a následně exportovány právě tímto programem. (PhpMyAdmin, 2019)

### 8.4 WampServer

WampServer je program umožňující mít na zařízení vlastní Apache server, který je potřeba ke spuštění PHP skriptů. Navíc obsahuje relační databáze s uživatelským rozhraním PhpMyAdmin. Právě na WampServeru jsem spouštěl Php skripty aplikací při jejich vývoji. (WampServer, 2019)

## 9 Objektově orientované programování

Aby byla pochopena funkce samotných aplikací je potřeba nejdříve přiblížit samotný termín „objektově orientované programování“ (dále OOP), neboť samotné aplikace napsané převážně v PHP se často pravidly pro OOP řídí.

Tento způsob programování byl vyvinut, aby programátorovi, co nejvíce ušetřil práci. Základním předpokladem pro úspěch v používání OOP je využívání jedné metody, třídy atd. pro více účelů, což se dá shrnout „proč vytvářet něco nového, když můžeme použít to, co již v programu máme, ale jen lehce obměněné“. (Php, 2019)

Základním pojmem OOP je „objekt“. Abych usnadnil pochopení, používám věci z reálného světa, kdy takový objekt, může být třeba „auto“. Každý objekt vždy má nějaké vlastnosti („atributy“) v případě auta, to může být třeba jeho „značka“. Jednotlivé objekty kromě atributů mohou mít i „metody“. Když se vrátím k našemu příkladu, auto může mít metodu „jed'(cíl)“ nebo metodu „zaparkuj()“. Program může volat jednotlivé objekty a například u metody „jed'()“, může kromě zavolání přidat i hodnotu, kterou metoda očekává; v tomto případě hodnotu „cíl“. Tím pádem je možné danou metodu zavolat vícekrát a vždy jí zadat jiný „cíl“. (Php, 2019)

V odstavci výše popisují, jak fungují objekty a jejich vlastnosti (atributy) a metody. Dále představuji pojem „třída“. Třída je něco jako vzor, podle kterého se jednotlivé objekty vytváří. Mohu mít například třídu „auta“, která vytváří jednotlivé instance „auta“, nechává jim přitom stejné metody (zaparkuj(), jed'(cíl)). Jednotlivé instance (tak se nazývají objekty vytvořené třídou) se liší svými vlastnostmi (atributy). Třída „auta“, může vytvořit několik objektů (aut), která budou mít jiné hodnoty vlastnosti „značka“; například jedna instance je „ford“, jiná zase „mercedes“, atd. (Php, 2019)

Důležité pro OOP jsou tři vlastnosti: „zapouzdření“, „dědičnost“ a „polymorfismus“.

Zapouzdření je ochrana určitých částí objektu (metod a atributů), tak aby při práci s objektem nebyla poškozena jeho vnitřní struktura. U našeho objektu „auto“ to může být například vlastnost „typ motoru“. Na této vlastnosti zároveň závisí více proměnných jako „maximální rychlost“ a „spotřeba“. Pro větší bezpečnost práci s tímto atributem, je „typ motoru“ zapouzdřen a vytváříme speciální metodu pro jeho ovládání zvenku nazvanou například „vymeňMotor“. (Php, 2019)

Dědičnost využijeme, pokud chceme vytvářet novou třídu, ale chceme jí přiřadit vlastnosti či metody třídy jiné. Například v našem případě s auty můžeme mít dvě třídy, jednou je třída „řidič“ a druhou je třída „automechanik“. Třídě „řidič“ dáme metody jako „nastartujAuto“ a „zkontrolujPalivo“, což je vše, co chceme, aby třída „řidič“ uměla. Třída „automechanik“ má ale ovládat i metody jako „opravMotor“, „vyměňKapaliny“ atp., které nemá třída „řidič“, zároveň však by měla třída „automechanik“ umět všechny metody, jež umí třída „řidič“. Právě zde se využívá dědičnost: třídu „automechanik“ získáme rozšířením třídy „řidič“ a ona podědí její metody. To znamená, že nemusíme psát dvě stejné metody jedné třídě. (Php, 2019)

Polymorfismus je jen trochu jiné použití dědičnosti. V praxi to znamená, že pokud dědíme metodu, která vykonává nějaký proces, tak pokud se daný proces v určitých parametrech liší, metoda se má pouze přizpůsobit (a nemá se tvořit nová metoda). Pro názornost se vraťme k příkladu s auty: třída „automechanik“ má metodu „změřVelikostSvětla“, která vyvolá jednoduchý proces změření určitého prostoru. Některá auta mají ale světla kruhová a jiná například čtvercová; zde se právě uplatní polymorfismus. Mohli bychom třídě dát dvě metody „změřVelikostSvětlaKruh“ a „změřVelikostSvětlaČtverec“, ale to je zbytečné. Stačí nechat původní metodu a jenom jí zadat parametry, podle potřeby, ať buď měří kruh, nebo čtverec. (Php, 2019)

## 10 Mechaniky her

V této části se více zaměřuji na mechaniky samotných her, abychom bylo snazší porozumět jejich fungování v rámci čistého kódu daných aplikací. K oběma hrám jsem proto vytvořil jednoduché grafy, které lze najít v příloze této práce (Příloha č. 1–3). Grafy jsem vytvořil pomocí free softwarové aplikace Draw.io Diagrams, kterou lze připojit ke Google Drive a pomocí které velmi snadno lze podobné grafy zpracovat. (Draw.io, 2019)

### 10.1 People v. Johnson

Tato hra je o něco jednodušší než hra Grimfield, její mechanismus je velmi dobře znázorněn na diagramu (Příloha č. 1). V této hře je důležitým faktorem čas, není zde pouze od toho, aby hráče „postrkoval“ k jednotlivým rozhodnutím, ale hru přímo ovlivňuje. Ve chvíli, kdy se hra zapne, mezi hráči vypukne diskuze a debata nad důkazy, které mohli nalézt na stránkách hry. (Krahulec, 2019b) Po uplynutí 30 minut si hráči mohou otevřít dopis od mladého detektiva, jehož prostřednictvím dostávají další důležité důkazy. Hráči ovšem dostanou jenom jednu ze dvou verzí dopisu, které jsou připraveny. Jednotlivé verze dopisu zásadně mění konec hry, protože v případě jedné verze dopisu je Mr. Johnson ve skutečnosti opravdu vrah, pokud hráči dostanou druhou verzi dopisu, ocitnou se ve verzi hry, kdy je Mr. Johnson nevinný.

Hráči mohou po 45 minutách definitivně rozhodnout o osudu Mr. Johnsona, tento časový limit je také pevně daný a hru nelze skončit dříve. Z toho jasně vyplývají celkově čtyři možné konce hry, pro každou verzi dva. Vždy je možné rozhodnout nesprávně, například odsoudit nevinného člověka, nebo naopak propustit vraha. Výsledek se hráči dozví po skončení hry formou videa.

Hra není časově ohraničena, ovšem hráč by v ní měl nejdéle strávit 50 minut. Čas, který je po celou dobu hry zobrazen, se proto po 50 minutách hraní zbarví do oranžova, aby hráči naznačil, že by měl hru ukončit. Po hodině a čtvrt se časovač zbarví do červena, aby dal hráči jasně najevo, že již by opravdu měl hru ukončit. Nikdy ho však násilně časovač neposune. Po skončení hry se časovač zastaví a hráč může vidět, jak dlouho mu hra trvala.

### 10.2 Grimfield 1665

Hra Grimfield 1665 je co se týče mechaniky hry o něco propracovanější, jak lze vidět na diagramu hry (Příloha č. 2–3). Tato hra se dělí na jednotlivá kola. Rozhodnutí učiněná v daném kole pak přináší jednotlivé výsledky, jak pro dané kolo, tak pro celou hru.

Hra hraje s dvěma důležitými proměnnými, které jsou pro hráče během hry skryté. Jednou z nich je počet obyvatel města Grimfield: město má na začátku hry 1000 obyvatel, ovšem mor jich vždy 200 zabije automaticky. Druhou proměnnou

je tzv. „plague danger“, neboli nebezpečí šíření moru. To je na začátku hry na nule a může se během hry zvedat při učinění některých rozhodnutí.

Hráči jsou v každém kole postaveni před jasně daný problém, který město momentálně prožívá. Kola jsou pojmenovaná podle počtu dnů od vypuknutí moru (celkově si město během hry prožije asi 400 dní). Hráči mají jasně dané možnosti, kterými mohou daný problém v jednotlivých kolech vyřešit. Například v první kole se setkávají s problémem, že v městě zůstal uvězněný obchodník a rád by odešel. Městská rada (hráči) má na výběr buď mu umožnit odchod, nebo ho donutit zůstat do konce trvání moru. Hráči učiní jedno ze dvou rozhodnutí, přitom při každém se jim dostane jiného výsledku. V případě obchodníka je tomu tak, že pokud ho nechají odejít, tak poděkuje a opustí město. V případě, že ho rada donutí zůstat, rozzuří se a brzo na to je nakažen morem. To jsou ovšem jen texty, které dávají hráčům odezvu na jednotlivé rozhodnutí v kole.

Co ovšem hráči nevidí, jsou dopady jejich rozhodnutí na jednotlivé dvě již dříve zmíněné proměnné, což je v diagramu znázorněno pomocí barev jednotlivých šipek. Oranžová barva znamená, že učinění tohoto rozhodnutí zvýší nebezpečí šíření moru (plague danger). Červené šipky znamenají, že v městě zemře určitý počet obyvatel, dané číslo je pak zapsáno stejnou barvou u šipky.

Zatímco nebezpečí šíření moru se vždy zvedá o jedna, počty mrtvých se mohou různit. Jedno rozhodnutí ve hře (v 6. kole), dokonce způsobuje obojí. V případě obchodník se tedy stane, že pokud je puštěn ven z města, zvedne se „plague danger“.

Na začátku tohoto popisu mechanik jsem řekl, že hra není časově limitovaná, i tak existuje předpoklad, za jaký čas by měl hráč ukončit kolo a za jaký čas by měl ukončit celou hru.

Ve hře hráč vidí dva časovače, jeden ukazuje čas strávený v daném kole a druhý je celkový čas hry. Na zahrání jednoho kola by měl hráč mít přibližně 5 minut, proto se čas v časovači zbarví po pěti minutách oranžově, aby hráče upozornil. Po sedmi minutách se pak časovač zbarví na červeně, aby dal jasně najevo, že by hráči měli pokračovat. Podobně tomu je u času celé hry, tam se časovač zbarví po hodině na oranžovo a po hodině a čtvrt se barva změní na červeně. Nikdy však hráče hra po uplynutí nějakého času násilně nepřesune.

Problematika času vede ke zvláštním možnostem v devátém a desátém kole hry, kde mají hráči možnost dané kolo přeskočit a neučinit tak (v daném kole) žádné rozhodnutí. Tato možnost je ve hře proto, aby hráč mohl celou hru ukončit dříve, pokud se v některých kolech zasekl na příliš dlouhý čas.

Po jedenáctém kole přichází konec hry, kdy se hráč dozví, kolik lidí ve městě přežilo. Konečná videa existují pouze dvě, v jednom případě se mor rozšíří do okolních měst a vesnic; v druhém případě se mor nerozšíří. Tento výsledek se rozhodne podle

stupně „plague danger“ na konci hry. Je-li roven 4 a více, je mor rozšířen. Pokud je stupeň menší, tak se mor nerozšíří.



## **11 Zpracování rozcestníkového webu**

Webový rozcestník je z velké části vytvořen s pomocí šablony. (Themesine, 2019) Tato šablona je napsaná v čistém HTML s použitím CSS a JS. Využívá přitom JQuery a Bootstrap framework. Šablona musela být upravena, aby vyhovovala potřebám rozcestníkového webu. Vlastní grafiku zpracoval externí grafik Jakub Vaniš, který se takto dohodl s konzultantem práce Mgr. Krahulcem. Veškeré kódy v HTML, CSS a JS vytvořil modifikací šablony autor bakalářské práce.

### **11.1 Responzivita**

Velká část práce na webu spočívala v tom, aby byl ve všem responzivní a přizpůsobil se různým obrazovkám. V některých ohledech byl v šabloně špatně použit Bootstrap framework, který sám o sobě umožňuje velmi dobře ulehčit práci a právě fungování responzivity.

Kromě dílčích úprav bylo třeba upravit horní část stránky, kde se nachází malé okno s informacemi. U něj bylo nutné zajistit, aby se při zmenšení displeje jeho pozice změnila.

### **11.2 Dlaždice s hrami**

Šablona obsahovala dlaždicový systém, stačilo jej tedy jen přizpůsobit. Dlaždice byly zachovány, změnila se jejich grafika. K tomu byla přidána vyskakovací okna, která se objeví při stisknutí tlačítka na dlaždici. V těchto oknech je vždy krátká anotace dané hry a vespod tlačítko s odkazem na hru.

## 12 Souborová struktura webových aplikací

Aby bylo pochopeno, jak webové aplikace fungují, je zapotřebí rozumět tomu, jakým způsobem jsou aplikace a celkově web pro hru koncipovány. Oproti klasickému webu, u kterého si dokážeme představit klasický koncept několik HTML souborů, každý pro jednu stránku, k tomu složku s obrázky a jeden CSS soubor či JS.

PHP však pracuje jinak a Nette, ve kterém jsou aplikace napsané, používají jinou koncepci. Ve své práci používám koncept, který byl použit u aplikace CEO, která byla vytvořena v rámci projektu Agama. (Krahulec, 2016) Celá koncepce je dobře vidět v Příloze č. 4, kde je vidět souborový strom pro web People v. Johnson. Tento souborový strom a další byly zpracovány pomocí programu TreeSize. (Jamssoftware: TreeSize, 2019)

V kořenové složce najdeme SQL soubor, kde je připravená databáze, která je nutná pro fungování aplikace. Tento soubor pouze obsahuje SQL kód, který při vložení do relačního databázového systému jako PhpMyAdmin dokáže vygenerovat databázi s potřebnými tabulkami.

V podsložce „www“ najdeme HTML index a v jednotlivých podsložkách důležité soubory pro web. V „pics“ najdeme obrázky, v „js“ zase JavaScriptové skripty, dále pak v „css“ jednotlivé kaskádové styly. To je klasické rozložení známé ze struktury webové stránky.

Zde se zaměřuji na podsložku „app“. V ní se nachází několik důležitých podsložek. Ve složce „config“ najdeme dva soubory, jeden konfiguruje připojení k databázi a druhý konfiguruje samotnou aplikaci (je v něm například seznam modulů a formulářů, které web používá).

V podsložce „presenters“ najdeme PHP soubory, které slouží k několika věcem. Zaprvé jsou spojením mezi moduly a formuláři a HTML šablonami (s použitím Nette latte). Zároveň provádí hlavní zobrazovací funkce či metody, které se dějí při uživatelské interakci se stránkou. Důležité jsou „BasePresenter“ a „SecureBasePresenter“, které se používají na většině stránek. V následné podsložce najdeme šablony, o kterých se mluví výše, rozdělené do složek, často podle názvů presenterů, které jsou s nimi spojeny. Tyto šablony jsou psané v HTML, ale je přitom použito Nette latte; často jsou v nich použity PHP proměnné. Do těchto proměnných je možné z presenterů posílat informace či data, která se pak na stránce promítnou. Lze také například přímo do šablony napsat PHP podmínku či jinou funkci. To vše umožňuje Nette latte. Důležitá šablona je „@layout.latte“, která slouží jako univerzální šablona pro všechny stránky, podobně jako tomu bylo u předchozích dvou základních presenterů.

Další dvě důležité podsložky jsou „model“ a „forms“. Ve složce „model“ najdeme modely, ve kterých jsou metody, jež komunikují s relační databází. Jednoduše, zde

jsou k nalezení SQL příkazy. Ve „forms“ jsou jednotlivé „FormFactory“, neboli továrny na vytváření formulářů. Formuláře tvoří jádro obou webových aplikací.

Hlavní komunikace probíhá mezi presentery a formuláři s modely. Uživatel se data dostávají přímo ze serveru do šablon. Díky tomu například nemůže uživatel zasahovat do kódu a nehrozí, že by jej nějak narušil.

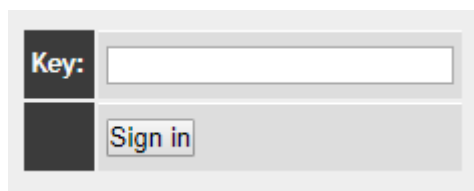
## 13 Zpracování webové aplikace pro hru People v. Johnson

Mechaniku samotné hry „People v. Johnson“ popisují v kapitole věnující se mechanikám her a na diagramu (Příloha č. 1). V této části se věnuji tomu, jak byla tato mechanika přenesena do prostředí vývoje webové aplikace. Některé již funkční části byly převzaty z webové aplikace pro hru CEO, již zpracovanou. Stejně jako určitá systematika vytváření her. (Krahulec, 2016)

### 13.1 Přihlašovací systém

Přihlašovací systém do hry byl vytvořen v rámci hry CEO. Jak takový systém vytvořit, popisuje i dokumentace k frameworku Nette. (Nette, 2019)

V praktickém směru jde o nutnost vstoupit do hry pomocí zadání kódu, který dostane hráč/žák od učitele. Tento kód slouží i pro pozdější vstup, když se chce žák podívat na výsledky, či pro vstup dalšího zařízení ke hře. Hru jako takovou lze spustit na více zařízeních / ve více prohlížečích, ovšem platí velmi jednoduché pravidlo, a to, že vždy rozhodnutí ve hře činí ten, kdo první klikne (učiní rozhodnutí ve hře). Systém pracuje s mezipamětí a cookies, pokud je uživatel přihlášen a stránku zavře a znovu otevře (ve stejném prohlížeči), tak může rovnou kliknout na možnost pokračování ve hře a nezasílat opět kód. Prohlížeč si, podobně jako u jiných webových stránek, uloží klíč, který jste zadali.

The image shows a simple login form. On the left, there is a dark grey vertical bar. To its right, the word 'Key:' is written in white. Further right is a white rectangular input field. Below the input field is a dark grey horizontal bar. On the left side of this bar is a dark grey square. To its right, the text 'Sign in' is written in white.

Ilustrace 3: Přihlašovací formulář

Důležitá přitom je komunikace s databází, jejíž strukturu najdete v přílohách práce (Příloha č. 5). Relační model byl vytvořen pomocí databázového systému Access z nabídky editorů Office. (Office: Access, 2019)

Přihlašování pracuje s tabulkou „p\_access\_keys“, ve které je ve sloupci „key\_hash“ uloženo heslo, které uživatel musí zadat. ID samotného klíče se pak používá jako cizí klíč v obou ostatních tabulkách v sloupcích „KEY“.

### 13.2 Generátor klíčů

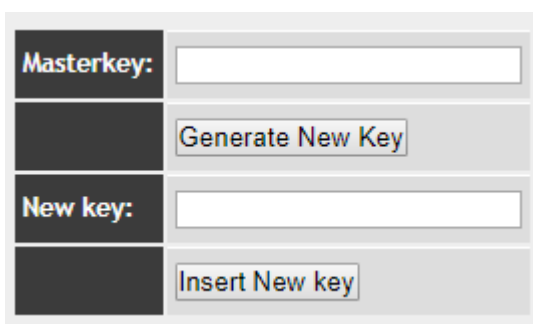
Pro zjednodušení administrativního přístupu ke správě klíčů jsem vytvořil tzv. „Generátor klíčů“, který tuto správu umožňuje bez nutnosti spravovat samotné tabulky v databázovém systému.

Tato jednoduchá aplikace je žákům/hráčům skryta, lze se k ní dostat pouze zadáním přesného URL. Stačí místo například „/homepage“ na konci URL zadat „/generator“.

Jedná se o jednoduchý formulář, který funguje jen pokud nejdříve zadáte tzv. „Masterkey“, který je kontrolou pro případ, že by aplikaci objevil někdo nežádoucí. Masterkey je pro obě hry jasně stanovený a je zapsaný uvnitř samotného PHP kódu.

Pokud je Masterkey zadán správně, je možné s tímto generátorem provádět dvě činnosti. První možností je nechat vygenerovat nový klíč kliknutím na „Generate New Key“, čímž se vygeneruje nový klíč sestávající se z náhodného řetězce znaků. Lze také vložit konkrétní klíč, a to vepsáním do volné kolonky a stisknutím „Insert New key“. V obou případech se nový klíč vloží do tabulky „p\_acces\_keys“ v databázi.

Proces vytváření nového klíče je ošetřen metodou, která zjišťuje, zda stejný klíč už neexistuje, pokud ano, oznámí to uživateli a klíč nevytvorí.



Ilustrace 4: Formulář pro vytváření klíčů

### 13.3 Časovač

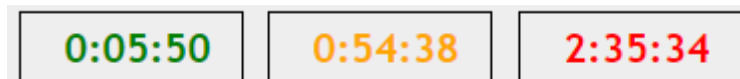
Důležitý pro hru je časovač, jak je velmi dobře vidět na diagramu mechaniky hry (Příloha č. 1). Časovač funguje uvnitř programu, pracuje s tabulkou „p\_timer“ v relační databázi, kam se po prvním vstupu do hry (prvním zadání klíče) nahraje do sloupce „START\_TIME“ aktuální čas, což je čas, kdy hra začala. Dál je třeba, aby se ve hře vypočítával aktuální čas v časovači. Tento proces zajišťuje JS, který běží po celou dobu hry, bere čas z tabulky a nechává si vždy nahrávat do jiné proměnné aktuální čas. „START\_TIME“ pak odečítá od aktuálního času a výsledek nahrává do časovače. Tento proces JS neustále opakuje, a proto vzniká iluze načítajícího se času. Proces je pochopitelně složitější, protože se v něm převádí jednotky času.

Přebarvování časovače podle toho, kolik času uplynulo, také zajišťuje JS pomocí funkce, která kontroluje uplynulý čas v proměnné „diff“ kam se v milisekundách ukládá rozdíl aktuálního a počátečního času. Pokud je čas vyšší než padesát minut a zároveň nižší než hodina a čtvrt, tak přiřadí „timeru“ novou class (třídu) „orange“, která je definována v CSS a pouze obarvuje text „timeru“ na oranžovo. Po hodině a čtvrt se přiřazuje třída „red“, která text obarvuje na červenou.

Hra samotná nejde ukončit dřív než za 45 minut a po 30 minutách má dát možnost odhalit další důkaz, který porotě přijde formou dopisu. Obě tyto funkce ošetřuje JS. Odhalení dopisu je rozděleno do tlačítka, která má odhalit dopis, a skrytého textu

dopisu. JS toto tlačítko primárně nastaví jako zakázané (disable), neboli ho uživatel nemůže zmáčknout. Stejně jako u obarvení je zde jednoduchá podmínka, která nahlíží do proměnné „diff“ a čeká, než uplyne 30 minut; pak jen umožní uživateli tlačítko stlačit, přes CSS je nastavené grafické ošetření, aby uživatel tuto změnu poznal. Ukončení po 45 minutách je ošetřeno podobně, změna je pouze v tom, že obě tlačítka pro rozsudek jsou součástí PHP formuláře a obě formulář odesílají. JS pouze nedovolí formulář odeslat dříve než po uplynutí 45 minut, na což uživatele upozorní vyskakovacím oknem.

Po ukončení hry a přechodu k výslednému videu je nutné čas zastavit. Proto se po odeslání formuláře, čímž se přechází k výsledku hry, použije metoda, která opět запиše aktuální čas do tabulky „p\_timer“, a to do sloupce „START\_TIME“. Tato tabulka má ještě sloupec „STAGE“. Při prvním záznamu neboli při startu hry se do tohoto sloupce nahraje „0“. Při konci hry, kdy se do tabulky nahraje nový záznam, je již místo nuly v sloupci „STAGE“ číslo 1. Tím pádem máme na konci hry dva záznamy v tabulce se stejnou hodnotou v sloupci „KEY“, neboť tam je uložen klíč hráče (jeho ID), ale zbytek hodnot v dalších sloupcích je různých. JS má nastaveno, že vždy bere čas počátku hry a odečítá ho od aktuálního času. Při výběru aktuálního času je třeba, aby se nejdříve splnila podmínka, že v tabulce „p\_timer“ není záznam, který by obsahoval ID uživatele, a zároveň byla ve sloupci „STAGE“ na řádce hodnota 1. Po ukončení hry, takový záznam existuje, a tak se podmínka nesplní; v takovém případě se místo aktuálního času vezme čas, který byl při ukončení hry, a od něj se odečte čas začátku. Výsledný čas se pak nahraje do „timeru“, a tak se po konci hry vytvoří iluze, že se čas zastavil.



Ilustrace 5: Časovač v různých fázích

### 13.4 Nový důkaz v průběhu hry

Během hry přichází porotě důkaz, který zásadně ovlivní celý výsledek. Tento důkaz existuje ve dvou verzích. Z předešlé kapitoly víme, že důkaz je pouze ukrytý text na stránce, který se po stisknutí tlačítka odhalí. Program je nastavený tak, že každá hra je rozhodnutá už po jejím začátku. Důležitá je v tomto ohledu tabulka „p\_guilty“, kam se ukládají důležité informace o hře.

Tato tabulka si první záznam vytvoří při prvním vstupu do hry. Aby se další záznamy netvořily při jakémkoliv dalším vložení klíče, je volána metoda „CheckKey“, která zjišťuje, jestli v této tabulce existuje záznam, který má ve sloupci „KEY“ zadané ID hráče. Pokud takový záznam není, tak se vytvoří a pokud již je, tak se nic nestane. Při vytvoření záznamu se do řádku v sloupci „JUDGE“ náhodně vloží 1 nebo 0, což je ošetřeno přes funkci. Zároveň se do „STAGE“ запиše 1, ale to je důležité kvůli jiné části programu.

V šabloně stránky je definované tlačítko pro odhalení dopisu, i samotný text dopisu. Ovšem v šabloně je jednoduchá podmínka, jež závisí na proměnné, kterou dostane šablona od PHP, které si ji vezme právě z tabulky a sloupce „JUDGE“. Podmínka určuje, který ze dvou textů dopisů se uživateli pak zobrazí.



Ilustrace 6: Tlačítko na odhalení dopisu

### 13.5 Rozsudek

Poté, co časovač dovolí hráči učinit rozsudek neboli odeslat formulář, zavolá se metoda, která opět zapisuje do tabulky „p\_guilty“. Tentokrát do sloupce „RESULT“ zapíše číslo od jedné do čtyř, podle toho, jakou verzi uživatel hraje a jaký rozsudek byl vynesena porotou (na jaké tlačítko bylo kliknuto).

Formulář samotný obsahuje pouze dvě tlačítka, která ho odesílají, každé z nich má přitom jiné jméno. Tudíž první věc, která se stane po odeslání, je, že se zkontroluje, jaké bylo jméno potvrzovacího tlačítka, a podle něho se rozděljuje podmínka na dvě větve. Tyto dvě větve se pak ještě jednou samy dělí na dva výsledky, podle toho, jaká hodnota je v sloupci „JUDGE“. Takto dostaneme čtyři konečné větve, kdy každá zadává své číslo do sloupce „RESULT“. Odeslání formuláře zároveň uživatele přesune na stránku s výsledkem hry, kde je v šabloně připravené okno pro YouTube video; odkaz k němu je definovaný skrz PHP proměnnou. PHP zjišťuje, jaká hodnota je v řádku sloupce „RESULT“, a podle toho posílá v proměnné odkaz na video k danému výsledku. Současně se během odeslání formuláře nahraje do sloupce „STAGE“ hodnota 2.



Ilustrace 7: Pohled uživatele

## 13.6 Kontrola postupu a přesměrování

Aplikace je nastavena tak, aby teoreticky šlo hrát ve více hráčích, kdy ostatní mohou sledovat průběh hry. Zde aplikace pracuje se zmíněným záznamem „STAGE“. Když se načítá první fáze hry, je-li u daného hráče hodnota „STAGE“ dva, rovnou jej pošle k výsledku. Díky této operaci je možné si prohlédnout výsledky hry i po delším času od ukončení hry samotné.

Zároveň by někoho mohlo napadnout, že by napsal příslušné URL k výsledné stránce během hry, rovnou skočil na výsledek a tím hru obelstil. To ovšem nelze, protože na výsledné stránce je kontrolována hodnota „STAGE“ také, a v takovém případě pošle hráče zpět do první fáze.

Jak je zmíněno výše, pokud aplikaci máme spuštěnou na více počítačích zároveň, vždy rozhoduje první, kdo učinil rozhodnutí (kliknul). Pokud někdo jako první klikne na rozsudek, tak i když další hráč za ním také klikne na jiný, tak bude pouze přesměrován na výsledek, podle rozhodnutí prvního.

## 13.7 Statistika případů

V šabloně, která je použita pro všechny stránky, je vložen například pravý blok, ve kterém kromě informací od autora najdeme dole i statistiku případů. Ta počítá, kolikrát se někdo pokusil případ vyřešit, kolik z těchto pokusů bylo úspěšných a kolik ne.

Jedná se opět o případ, kdy PHP posílá hodnoty do proměnných v šabloně. K tomuto existují dvě metody, jedna pouze prochází tabulku „p\_guilty“ a počítá každý záznam, u kterého je hodnota ve sloupci „STAGE“ 2. Ta počítá celkový počet dohraných her. Druhá metoda počítá stejným způsobem záznam, jen kromě kritéria ohledně hodnoty ve sloupci „STAGE“ ještě nahlíží do sloupce „RESULT“ a vybírá jen ty záznamy, kdy je hodnota tohoto sloupce 1 nebo 4 (to jsou úspěšné konce). Díky nim známe počet úspěšných případů. Počet neúspěšných pak úsporně získáme pomocí odečtu počtu úspěšných řešení od celkového počtu ukončených případů. Tyto hodnoty se posílají jako proměnné do šablony.

### Statistics

Juries that tried to crack  
the case so far: 2

Succeeded: 1

Failed: 1

*Ilustrace 8: Statistika případů*



## 14 Zpracování webové aplikace pro hru Grimfield 1665

Stejně jako u webové aplikace pro hru „People v. Johnson“, i zde byla použita struktura a některé části kódu z webové aplikace pro hru CEO. (Krahulec, 2016) Některé části jsou zcela totožné s předešlou hrou, tudíž je zde neopakují.

Stejně funguje Přihlašovací systém a Generátor klíčů, pouze při zápisu klíčů se používá tabulka „d\_access\_keys“, kterou můžete vidět na náhledu do relační databáze (Příloha č. 6.). Relační model byl vytvořen pomocí databázového systému Access z nabídky editorů Office. (Office: Access, 2019)

### 14.1 Systém kol

Tato hra je velmi specifická tím, že je rozdělena na jednotlivá kola, ve kterých hráč vždy musí učinit rozhodnutí, následně si může přečíst výsledek svého rozhodnutí a přejít do dalšího kola, kde se proces opakuje. Jak je zmíněné v kapitole o Mechanice této hry, je možné některá kola z časových důvodů přeskočit.

Dál popisují, jak tento systém vypadá. Vždy existuje šablona pro jednotlivé kolo a ve stejné složce je šablona pro stránku s výsledkem, tzv. „outcome page“. Postup kol tedy znamená přechod mezi jednotlivými šablonami.

Day 10

Stage: 0:00:04      Game: 0:00:08

One of the boys from the grammar school might be infected. Some boys from the local grammar school do not live in Grimfield, they would like to return to their families. They are scared, but they look healthy.

Will you let them return home?

Let them go home.  
 They must stay in Grimfield.

Submit

Ilustrace 9: Ukázka průběhu kola

### 14.2 Časovač

Tato hra má jeden velký rozdíl oproti hře předchozí, a to, že v ní jsou časovače dva. Jeden určuje čas pro celou hru a jeden čas na dané kolo. Časovač pro celou hru funguje stejně jako u „People v. Johnson“, počítá čas od prvního vloženého klíče pomocí odečítání od aktuálního času, pracuje s tabulkou „d\_timer“.

Časovač pro kola funguje hlavně na faktu, že ve chvíli, kdy se zapisuje výsledek kola do tabulky „d\_stage\_results“, se do sloupce „TIME\_STARTED“ zapíše aktuální čas.

„SecureBase Presenter“, který se stará o posílání údajů z tabulek do šablony, kde časovače jsou, pošle časovači kola právě tuto hodnotu, zapsanou při zapisování výsledku minulého kola. Čas nového kola se tudíž začne počítat od konce kola posledního. U prvního kola je ošetřeno, aby byl čas brán z tabulky „d\_timer“, kde se počítá čas celé hry.

Obarvování časovačů opět zajišťuje JS, kde existují dvoje pravidla pro dva timery. Jeden s indexem nula a druhý s indexem jedna. Přitom jeden kontroluje časovač celé hry a druhý časovač kola. Každý obarvuje po uplynutí jiného času.

Časovače je třeba v některých fázích skrýt, a to na stránce, kde máme výsledek minulého kola a když jsme na konci hry. Tam nás pouze zajímá za jak dlouho byla celá hra odehrána. To je ošetřeno přes podmínku ve šabloně s časovači. V „SecureBase Presenteru“ se vždy řeší, jaký záznam si má časovač kola vybrat z tabulky „d\_stage\_result“ při jednotlivých kolech. To zjišťuje tím, že si nejdříve skrz metodu „GetStage“ zjistí, v jaké části hry hráč je. To je zapsané v tabulce „d\_status\_of\_town“, kde se vždy při posunu z jedné části do druhé číslo v sloupci „STAGE“ zvýší o jedna. To ovšem platí i pro chvíli, kdy hráč přechází ze stránky s rozhodnutím na stránku s výsledkem. V prvním kole je hodnota „STAGE“ 1 a ve chvíli, kdy hráč přejde na stránku s výsledkem z kola, tak je již hodnota 2. Ve druhém kole je hodnota 3 a na stránce s výsledkem druhého kola je hodnota 4. Dokážeme tudíž poznat, že hráč je na stránce s výsledkem, a ne na stránce daného kola, kde se teprve rozhoduje. Následně je tedy možné, aby ve chvíli, kdy je hráč na stránce s výsledkem, poslat do šablony proměnnou, která splní podmínku a nechá zmizet časovače. Podobně je to nastaveno pro zmizení jednoho časovače na konci hry.

Na konci hry ještě přichází problém s tím, že jelikož je na stránce pouze jeden časovač, jeho index je automaticky nula. Nulový index měl časovač kola, to znamená, že se časovač obarvuje podle pravidel určených pro kolo. My chceme, aby se obarvoval vzhledem k mechanice obarvení pro celou hru. Zde je v JS jednoduchá metoda, která zjistí text z URL, a následuje podmínka, která říká, že pokud je v URL „result“ (tak je označená konečná stránka s výsledky hry), tak platí pro časovač s indexem nula podmínky, jako by byl časovač pro celkovou hru.

Stage: <b>0:05:10</b>	Game: <b>0:13:04</b>
--------------------------	-------------------------

Ilustrace 10: Časovač

### 14.3 Stav města

Jak jste si mohli dočíst v kapitole o mechanice této hry, jsou zde důležité dvě proměnné. Populace města a „nebezpečí moru“ (plague danger). Ve chvíli, kdy někdo zadá klíč a například se vygeneruje začáteční čas pro daný klíč, stejně tak

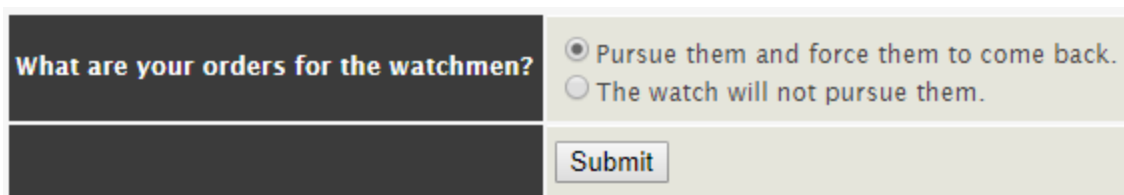
se vytvoří záznam do tabulky „d\_status\_of\_town“. Tam se, kromě zapisování „STAGE“, který je při prvním vstupu nastaven na 1 a následně se zvyšuj, do tabulky vloží 800 ve sloupci „PEOPLE“ a nula v sloupci „PLAGUE“. To nastavuje prvotní stav populace a moru. Lidí původně ve městě je 1000, ale mor vždy automaticky zabije 200 lidí, a tak je v pořádku nastavit hodnotu hned na začátku na 800.

#### 14.4 Průběh kola

V šabloně stránky každého kola je napsaný pouze nadpis a text, který osvětluje problém, ve kterém se nyní město nachází. To je pevně zapsané v šabloně. Poté se na každé stránce vytváří formulář pro dané kolo. Formuláře a stránky jsou označovány vždy podle posloupnosti „First“, „Second“, atd. Formulář v sobě má vždy jen otázku jako text a radiobutton na možná rozhodnutí. Aby se zabránilo možnosti odeslat formulář bez označené možnosti, tak se vždy automaticky označí první možnost. V klasickém případě má formulář pouze tlačítko na odeslání výsledku.

Uživatel může překlíkat mezi jednotlivými možnostmi pomocí radiobuttonu dle libosti. Když odešle formulář pomocí tlačítka, tak se vyhodnotí jeho rozhodnutí. Ve hře některá rozhodnutí zabijí populaci a jiná zvyšují nebezpečí moru, někdy dokonce obojí. Na toto ovlivňování byly vytvořeny dvě metody, jedna „MorePlague“, která vždy v tabulce „d\_status\_of\_town“ v záznamu daného uživatele zvýší hodnotu ve sloupci „PLAGUE“ o jedna. Druhá metoda, která se v některých kolech používá, je „KillPeople“, ta vždy očekává nějaké číslo, o které pak v tabulce „d\_status\_of\_town“ sníží v záznamu uživatele hodnotu ve sloupci „PEOPLE“.

Kromě této variabilní části se vždy zvýší hodnota v tabulce „d\_status\_of\_town“ v záznamu uživatele ve sloupci „STAGE“ o jedna, protože se přechází do další části. Zároveň se výsledek uloží do tabulky „d\_stage\_results“, kde se do sloupce „STAGE“ nahraje číslo daného kola, to znamená, že pro první kolo je to číslo 1, pro druhé 2 atd. Oproti sloupci „STAGE“ v druhé tabulce tento nepočítá stránky s výsledky daných kol. Kromě čísla kola je zapsán výsledek daného kola v sloupci „RESULT“ a nakonec je do sloupce „TIME\_STARTED“ uložen aktuální čas.



What are your orders for the watchmen?

Pursue them and force them to come back.

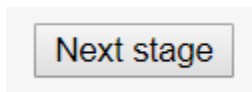
The watch will not pursue them.

Submit

Ilustrace 11: Formulář pro rozhodnutí kola

Stránku s výsledkem daného kola najdeme ve stejné složce pod názvem „outcome“. V této šabloně je podmínka, která podle určité proměnné ukazuje určitý text. Presenter dané stránky si nejdříve pomocí metody „getResult“ zjistí výsledek kola a tuto proměnnou pošle do šablony, kde se podle toho zobrazí správný text k danému výsledku. Na všech „outcome“ stránkách po kolech je vždy stejný „SubmitForm“, který obsahuje pouze jedno tlačítko, které má hráče posunout

do dalšího kola. Kromě přesměrování hráče na stránku dalšího kola, ještě zvyšuje hodnotu „STAGE“ v „d\_status\_of\_town“ tabulce.

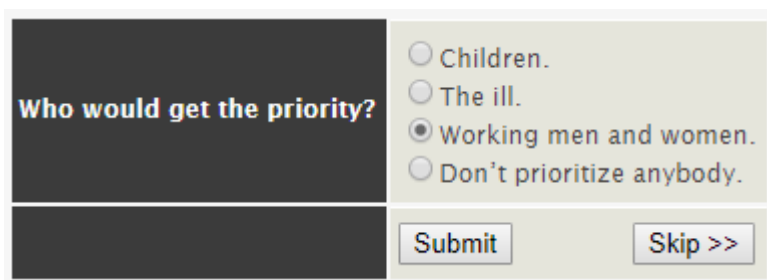


Ilustrace 12: Tlačítko pro pokračování do dalšího kola

## 14.5 Možnost přeskočení kol

Jelikož si během hry hráč hlídá nejen čas na jednotlivá kola, ale i celkový čas hry, je možné, že k dlouhým přemýšlení nad jednotlivými rozhodnutími ztratí moc času, a bude potřebovat zrychlit. K tomuto lze dvě kola před posledním kolem přeskočit. Tato kola se liší v tom, že v jejich formuláři je ještě jedno tlačítko na odeslání formuláře, a to „Skip“.

Tento formulář funguje jako ostatní, jenom před jeho vyhodnocení se nejdříve, podobně jako u hry People v. Johnson, řeší název tlačítka, kterým byl formulář odeslán. Pokud se jedná o tlačítko „Skip“, tak se do tabulky „d\_stage\_results“ do kolonky „RESULT“ zapíše 0. To potom na „outcome page“ pouze jen napíše, že kolo bylo přeskočeno.

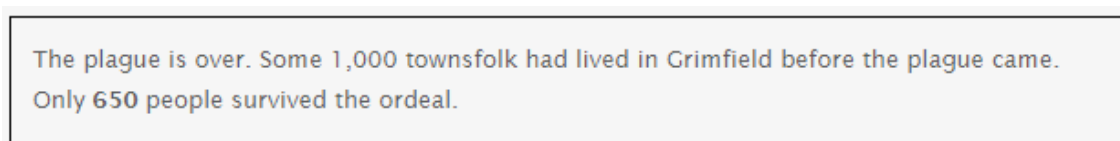
A screenshot of a web form. On the left, a dark gray box contains the text "Who would get the priority?". To the right, there are four radio button options: "Children.", "The ill.", "Working men and women." (which is selected), and "Don't prioritize anybody.". Below the options are two buttons: "Submit" and "Skip >>".

Ilustrace 13: Formulář s možností přeskočit kolo

## 14.6 Výsledek hry

Po dokončení hry se hráč dostane na stránku s výsledkem. V první řadě vidí text o počtu obyvatelstva na konci hry, kam se vloží proměnná, kterou získá aplikace pomocí metody „GetPeople“. Ta jen najde hodnotu v tabulce „d\_status\_of\_town“ v záznamu uživatele v sloupci „PEOPLE“.

Druhá část konce je výsledné video, tam pouze záleží na tom, jaká je hodnota proměnné „plague danger“ (nebezpečí moru). Podobně jako u People v. Johnson se vloží URL videa z YouTube, podle toho, jestli hodnota, kterou vrátí metoda „GetPlague“, je rovna, nebo vyšší než 4.



Ilustrace 14: Výsledek počtu obyvatel

## 14.7 Kontrola postupu a přesměrování

Ve všech částech hry se kontroluje, jestli je uživatel ve správné části. Ověřuje se to na hodnotě „STAGE“ v tabulce „d\_status\_of\_town“. Pokud hodnota nesouhlasí s částí hry, kde se nyní hráč nachází, se zjistí pomocí metody „GetStage“. Tak začne proces přesměrování, kde existuje pole (array), ve kterém jsou uloženy jednotlivé adresy částí. Adresy jsou v poli seřazeny chronologicky, takže pokud se použije hodnota, kterou zjistím skrz „GetStage“ jako index, měla by být nalezena správná část v poli. Příklad: Hráč je v druhém kole a hodnota v STAGE je tedy 3. Ted' napíše do URL „first“ a zkusí se tím vrátit do prvního kola. Jenomže tam se provede kontrola, která pomocí GetStage dostane hodnotu 3. Kontrola očekává 1, takže podmínka není splněna. Je tedy přesměrován na hodnotu v poli s indexem 3. Na indexu 3 je hodnota „Second:default“ a na tuto adresu je uživatel přesměrován.

Jediná výjimka je, pokud je hodnota „Result:default“, protože to znamená, že hra již byla ukončena, a tak je uživatel stejně přesměrován, jen dostane jiné hlášení, ve kterém se říká, že hra již byla ukončena.

Stále platí pravidlo že ten, kdo udělá první rozhodnutí (první klikne), rozhoduje za všechny, kteří pod stejným klíčem nahlížíjí do hry. Ti, i když následně zvolí jinou možnost, jsou přesměrováni s tím, že rozhodnutí již bylo učiněno.

## 14.8 Statistika obyvatel

Stejně jako u hry „People v. Johnson“ i zde je do postranního sloupce vložena statistika. Tentokrát se jedná o statistiku průměrného počtu obyvatel. Proměnnou, kterou poté posílám do šablony, získáme jednoduchou metodou, která pracuje s SQL funkcí Avg(), která počítá průměrnou hodnotu. Stačí nastavit, aby byly vybrány pouze ty záznamy z tabulky „d\_status\_of\_town“, kde je hodnota „STAGE“ 23, což znamená, že je hráč na konci hry (11 kol + 11 stránek s výsledky + konečná stránka).



Ilustrace 15: Statistika obyvatel

## 15 Ukázky kódu

V této části ukazují na několika exemplářích kódu, jak dané funkce jednotlivých aplikací funguje.

### 15.1 Přesměrování u hry Grimfield 1665

Kód najdete v příloze práce (Příloha č. 7). Zde si podrobně rozebereme, jak daná část programu funguje a jak je zajištěno přesměrování. Tato část kódu byla vzata z „FirstPresenter“. Funkce zajišťuje, že se do dané části nedostane uživatel, který již je ve hře na jiném místě. Program si nejdříve do proměnné „stage“ nahraje pomocí metody „GetStage“ aktuální hodnotu v tabulce, kde je zapsáno, v jaké části hry se nachází.

Jako první se řeší podmínka, jestli se stage nerovná 1, tak pokračuj. Pokud je hodnota proměnné stage 1, tak je všechno v pořádku a uživatel se nachází ve správné části hry. Pokud se hodnota nerovná 1, přichází na řadu pole „stageslibary“, kde se nachází několik hodnot. Jedná se hodnoty, které odkazují na jednotlivé části hry. Na prvním místě v poli jsou prázdné uvozovky, protože chceme pole začínat na hodnotě 1 a ne 0. Toto pole je důležité pro utváření nové proměnné „newstage“, kam se nahraje hodnota ze stageslibary s indexem vyjádřeném hodnotou stage. Například pokud je hodnota stage 5, do newstage se načte hodnota ze stagelibary s indexem 5, což je „Third:default“.

Následuje podmínka, která ověří, jestli je hodnota v newstage „Result:default“, pokud by tomu tak bylo, znamená to, že je hra ukončena, pokud ne, tak hra stále běží. Rozdíl je pouze ve znění „flashMessage“, která se uživateli zobrazí. Pokud je hodnota „Result:default“ je uživatel na tuto stránku přesměrován, pokud není, je přesměrován na stránku, která je uložena v hodnotě proměnné newstage.

### 15.2 Odeslání formuláře (ukončení kola) ve hře Grimfield

Zde si rozebereme, jak v kódu vypadá vyhodnocení ukončení kola ve hře Grimfield (Příloha č. 8). Funkce se vyvolá po odeslání dotazníku. Jako první použije metodu „MoveNextStage“, což přepíše hodnotu STAGE v tabulkách, kde se monitoruje postup uživatele skrz aplikaci. Následně pomocí metody „getName“, získá název tlačítka, kterým byl formulář odeslán a uloží tento název do proměnné „submit“. Jedná se totiž o kolo, které je možné přeskočit, a tak pokud by byl název tlačítka „skip“, pak uživatel stiskl tlačítko pro přeskočení. Proto první podmínka kontroluje, jestli je hodnota v proměnné submit „send“, pokud ne, tak uživatel stiskl přeskakovací tlačítko. V tom případě se pomocí metody „InsertResult“ zapíše do tabulek výsledek, který je v tomto případě 0.

V případě, že je hodnota proměnné submit „send“, tak hráč provedl rozhodnutí a je nutné pokračovat dál. Jelikož má formulář hned 4 radiobuttony, je možné vybírat ze 4 možností. Zde přichází série podmínek, které ověřují, jestli v hodnotě

„result“ není jedna z hodnot (one, two, three, nebo four). V každém výsledku se zapisuje výsledek pomocí metody „InsertResult“ a v jednom případě se ještě volá metoda „KillPeople“, která mění hodnotu „PEOPLE“ v tabulkách v tomto případě ji sníží o 30.

### **15.3 Statistika případu u hry People v. Johnson**

V příloze (Příloha č. 9) naleznete dvě části kódu, ta první je z „@layout“ šablony, která určuje šablonu pro všechny stránky daného webu. Druhá část je funkce z „BasePresenteru“, který také operuje s většinou částí webu. Na této části kódu ukazují, jak funguje komunikace šablony a presenteru. V horní části se nachází zápis ze šablony, kde je použito HTML pro zapsání textu do postranního sloupce stránky. Kromě toho jsou v závorkách klasické PHP proměnné (allgames, wingames, lostgames). Tyto proměnné získá šablona právě z presenteru. Když se podíváme do jeho funkce, vidíme, že presenter si nejdříve nahraje informace z několika metod (getNumberOfGames a getNumberOfWinGames) a z těchto metod nahrává hodnoty do proměnných (allgames a wingames). Proměnnou lostgames si pak vytvoří sama šablona, pomocí dvou předchozích metod a jejich vzájemného odečtení.

Následuje proces, kdy jsou tyto proměnné posílané do šablony ke stejnojmenným proměnným. Takto snadno komunikuje PHP s HTML (tedy v tomto případě s Nette latte) šablonou.

## 16 Testování obou aplikací

Obě webové aplikace byly podrobeny testování. Testování proběhlo na Masarykově univerzitě v rámci předmětů vyučovaných konzultantem práce a autorem celého projektu Agama, Mgr. Krahulcem.

Pro obě hry byl vytvořen dotazník zpětné vazby pomocí Google Forms. (Google: Forms, 2019) Pro obě aplikace byly zvoleny stejné otázky. Vzor dotazníku a některé grafy výsledků jsou k nalezení v přílohách této práce. (Příloha č. 10–12)

Podrobnější výsledky zapsané v tabulkách, lze nalézt v elektronické příloze práce.

### 16.1 Zpětná vazba na aplikaci pro hru People v. Johnson

Testování aplikace pro hru People v. Johnson proběhlo ve dnech 18. března 2019 a 25. března 2019 přitom jednou se jednalo o skupinu 13 osob a podruhé pouze o 5 osob. Věková hranice uživatelů byla nad 20 let s tím, že se pohybovala průměrně okolo 22–23 let. V první skupině přitom byli dominantní Češi, zatímco ve druhé byla většina cizinců.

Hlavní zhodnocení přináší dvě jednoduché číselné škály, kdy měli testeři (studenti) celkově hodnotit aplikaci a její grafický vzhled na pětistupňové škále, kdy 1 znamená nejlepší hodnocení a 5 nejhorší (jako ve škole).

Když se podíváme na grafy výsledků těchto otázek (Příloha č. 11), můžeme vidět relativně vysoké procento i u odpovědí 4 a 5. Po hlubším náhledu do výsledku dotazníku a přečtení následných textových odpovědí jsem došel k závěru, že si uživatelé špatně přečetli zadání a hodnotili obráceným způsobem, jak jsou zvyklí u řady hodnocení hvězdičkami z různých webových anket. Když přihlédneme k této okolnosti, vychází kladné výsledky, hodnocení je v drtivé většině naprosto kladné, či jen drobnými výhradami. Průměrné hodnocení se vztahuje pouze ke grafickému zpracování aplikace.

Poslední dvě otázky dotazníku se orientují na písemné vyjádření k aplikaci, a to jak k možným výhradám, tak k i možnosti zlepšení. Tabulku s výsledky a těmito textovými záznamy najdete na elektronické příloze práce. Většina odpovědí hodnotí aplikaci kladně a nemá k ní připomínky. Objevují se tu často odpovědi, které jsou směřované k celému webu, a ne konkrétně k aplikaci samotné. U jednoho uživatele se objevuje neporozumění funkci aplikace a u jiného zas, že si nevšiml tlačítka pro opětovný vstup do aplikace a místo toho opakovaně zadával kód. Ke grafické stránce se pak objevuje několik názorů, že by mohla být aplikace lépe graficky zpracována a mohla by být barevnější, pestřejší, např. s použitím více obrázků.

Celkově byla aplikace zhodnocena kladně a zdá se být pro svůj účel vhodná a splňující všechny potřeby dané hry.



## 16.2 Zpětná vazba na aplikaci pro hru Grimfield 1665

Testování aplikace pro hru Grimfield 1665 proběhlo ve dnech 26. března 2019 a 1. dubna 2019 přitom jednou se jednalo o skupinu čítající 6 osob a podruhé 9. Věková hranice uživatelů byla nad 20 let s tím, že se pohybovala průměrně mezi 22–25 let. V obou skupinách byli cizinci a Češi značně smíchaní. Zajímavé je, že se testování účastnily dvě výrazně starší osoby, jedné bylo 43 a druhé 28.

Stejně jako u předchozí zpětné vazby zde máme jasné hodnocení ze dvou číselných škál. Výsledek můžeme vidět na grafu v příloze práce (Příloha č. 12) a tabulku s přesným zněním všech odpovědí najdeme v elektronické příloze. Opět se zde stalo, že ne všichni testeři si přečetli instrukce a škálu pochopili obráceně. Celou aplikaci hodnotí nadpoloviční většina hodnotou 1, což se dá považovat za velký úspěch (musíme přihlídnout k tomu, že většina hodnocení 5 má vlastně být 1). Strohost aplikace a formulářů v nich se ukazuje na druhém grafu jako příčina skutečnosti, že hodnocení grafické stránky se pohybuje nejvíce okolo hodnot 2 až 3.

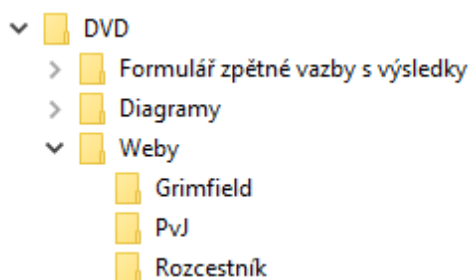
Bližší informace zjistíme, když nahlédneme do odpovědí na otázky, kde mohli testeři napsat, co je na aplikaci podle nich špatně nebo jak ji vylepšit. Často se zde vyskytuje požadavek na nějaké obrázky, které by dokreslovaly atmosféru a doprovázely strohý text. Často se vyskytuje požadavek na více možností k rozhodnutí, dvě se často zdají příliš svazující. To ovšem není problém aplikace, ale samotné mechaniky hry. Hlavním argumentem je vylepšení grafiky a s ním související zařazení obrázků.

Celkově si tedy myslím, že by této aplikaci prospělo zlepšení grafického designu a přidání obrázků, aby hra byla atraktivnější. Jinak se nezdá, že by testeři přišli na nějaký výrazný problém znemožňující funkčnost hry. Tudíž hodnotím testování jako úspěšné.

## 17 Instrukce k elektronické příloze práce

Elektronická příloha k práci obsahuje několik důležitých dat. Můžeme zde nalézt diagramy mechanik obou her, tak aby bylo možné prohlédnout si je v lepší kvalitě. Dále se zde nachází vzor obou formulářů zpětných vazeb na hodnocení aplikací pro obě hry. Důležitější jsou dvě tabulky obsahující záznamy z vyplnění obou formulářů, kde se dá podrobně studovat zpětná vazba.

Nejdůležitější částí této práce je ta praktická. Najdeme zde rozcestníkový web i oba weby, na kterých se nachází aplikace touto prací vytvořené. Připomínám, že pro možnost vyzkoušet si obě aplikace je potřeba zprovoznit na zařízení lokální server a relační databázi, ta je pro obě aplikace v SQL kódu přiložena.



Ilustrace 16: Souborový strom elektronické přílohy

## Závěr

Shrnutí a zamyšlení nad praktickou částí práce přináší kapitola se zpětnou vazbou. Zde zdůrazňuji jen několik nejpodstatnějších myšlenek.

Jako hlavní výsledek práce vznikly dvě nové webové aplikace a jeden webový rozcestník. Jejich funkčnost byla ozkoušena pomocí testování, kterým prošly obě aplikace. Závěrem lze říct, že zpracovanou webovou stránku a aplikace lze hodnotit jako funkční a připravené na nasazení do praxe v rámci projektu Agama. Cíle praktické části práce byly naplněny.

V teoretické části práce nabízí základní vhled do tématu didaktických her, gamifikace a webových aplikací používaných v edukaci. Práce obsahuje také základní přehled a popis nástrojů, pomocí kterých zpracována byla praktická část, a které jsou tím pádem vhodné k tvoření podobných webových aplikací.

Teoretickou část práce by bylo možné rozšířit a uvést více zdrojů; takové rozšíření by ovšem bylo nad rámec zadání této bakalářské práce, jejímž cílem je podat základní ucelený přehled problematiky. Pro splnění tohoto cíle je rozsah a úroveň podaných informací dostačující.

Hlavní přínos práce spočívá v její praktické části, která přispívá k možnosti úspěšného nasazení projektu Agama v pedagogickém procesu.

## Seznam použité literatury a zdrojů

*Bitbucket* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://bitbucket.org>>.

*Bootstrap* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://getbootstrap.com>>.

DEVEDZIC, Vladan. *Semantic Web and education*. New York: Springer, 2006. ISBN 978-0-387-35416-3.

*Draw.io* [online]. c2019, [cit.2019-03-10]. Dostupné z: <<https://www.draw.io>>.

*Git* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://git-scm.com>>.

*Google: Forms* [online]. c2019, [cit.2019-03-25]. Dostupné z: <[https://www.google.com/intl/en\\_EN/forms](https://www.google.com/intl/en_EN/forms)>.

*Jamsoftware: TreeSize* [online]. c2019, [cit.2019-03-26]. Dostupné z: <[https://www.jam-software.com/treesize\\_free](https://www.jam-software.com/treesize_free)>.

*Jetbrains: PhpStorm* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://www.jetbrains.com/phpstorm>>.

*jQuery* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://jquery.com>>.

KIRYAKOVA, Gabriela, ANGELOVA, Nadezhda a YORDANOVA, Lina. *Gamifiction in education*. Stara Zagora, 2013. Trakia University.

KRAHULEC, Ondřej. *Synchronous Online Simulation Games for Development of Speaking Skills in EFL*. Brno, 2016. Diploma Thesis. Masarykova univerzita. Vedoucí práce Mgr. Jana Zerzová, M.A., Ph.D.

KRAHULEC, Ondřej. *Grimfield* [online]. c2019, [cit.2019-03-07]. Dostupné z: <<http://agama.ped.muni.cz/grimfield>>.

KRAHULEC, Ondřej. *People v. Johnson* [online]. c2019, [cit.2019-03-07]. Dostupné z: <<http://agama.ped.muni.cz/pvj>>.

MANAS, Adele. *Gamifikace a její praktické využití v rámci školní výuky prostřednictvím tabletů*. Praha, 2015. Diplomová práce. Univerzita Karlova v Praze. Vedoucí práce Mgr. Radim Wolák.

*MySQL* [online]. c2019, [cit.2019-03-07]. Dostupné z: <<https://www.mysql.com>>.

*Nette: latte* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://latte.nette.org/cs>>.

NOVÁKOVÁ, Jiřina. *Aktivizující metody výuky*. Praha: Univerzita Karlova, Pedagogická fakulta, 2014. ISBN 978-80-7290-649-9.

*Office: Acces* [online]. c2019, [cit.2019-03-26]. Dostupné z: <<https://products.office.com/cs-cz/access>>.

*Php* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<http://php.net>>.

*PhpMyAdmin* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://www.phpmyadmin.net>>.

SKOŘEPOVÁ, Kateřina. *Typologie her a jejich konceptualizace*. Praha, 2011. Bakalářská práce. Univerzita Karlova v Praze. Vedoucí práce PhDr. Dana Bittnerová, CSc.

ŠAFARČÍKOVÁ, Simona. *Hry postav a situací: metodický list*. Plzeň: Labyrint, 2011.

*Themesine: Trace – Travel Agency Template* [online]. c2019, [cit.2019-02-01]. Dostupné z: <<https://www.themesine.com/downloads/trace-travel-agency-template>>.

*W3C: HTML and CSS* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://www.w3.org/standards/webdesign/htmlcss>>.

*W3C: JavaScript Web APIs* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<https://www.w3.org/standards/webdesign/script>>.

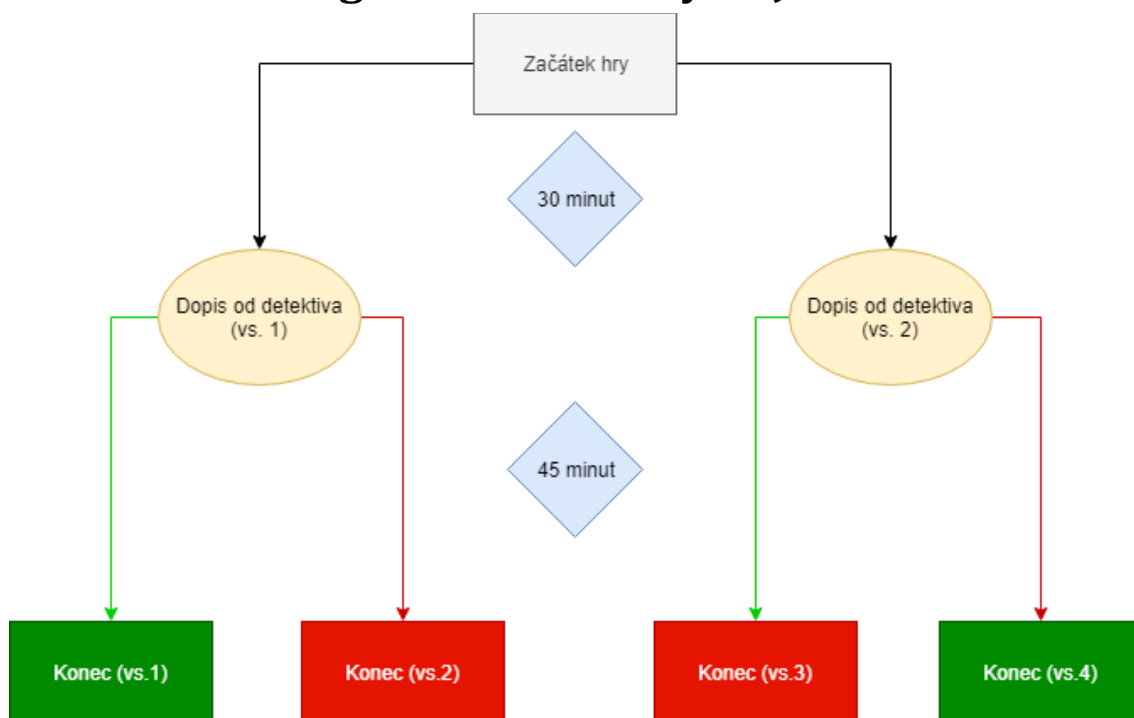
*W3schools* [online]. c2019, [cit.2019-03-25]. Dostupné z: <<http://www.w3schools.com>>.

*WampServer* [online]. c2019, [cit.2019-02-28]. Dostupné z: <<http://www.wampserver.com>>.

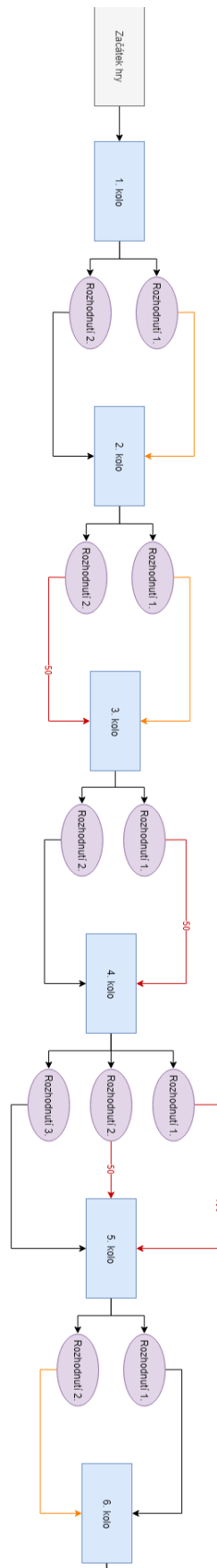
## Seznam příloh

Příloha č. 1 – Diagram mechaniky PvJ .....	47
Příloha č. 2 – Diagram mechaniky Grimfield.....	48
Příloha č. 3 – Diagram mechaniky Grimfield (2) .....	49
Příloha č. 4 – Souborový strom PvJ.....	50
Příloha č. 5 – Relační model databáze pro PvJ.....	51
Příloha č. 6 – Relační model databáze pro Grimfield .....	52
Příloha č. 7 – Ukázka kódu (Přesměrování) .....	53
Příloha č. 8 – Ukázka kódu (Konec kola).....	54
Příloha č. 9 – Ukázka kódu (Statistika případů) .....	55
Příloha č. 10 – Vzor formuláře zpětné vazby .....	56
Příloha č. 11 – Grafy výsledků zpětné vazby PvJ .....	57
Příloha č. 12 – Grafy výsledků zpětné vazby Grimfield.....	58

## Příloha č. 1 – Diagram mechaniky PvJ

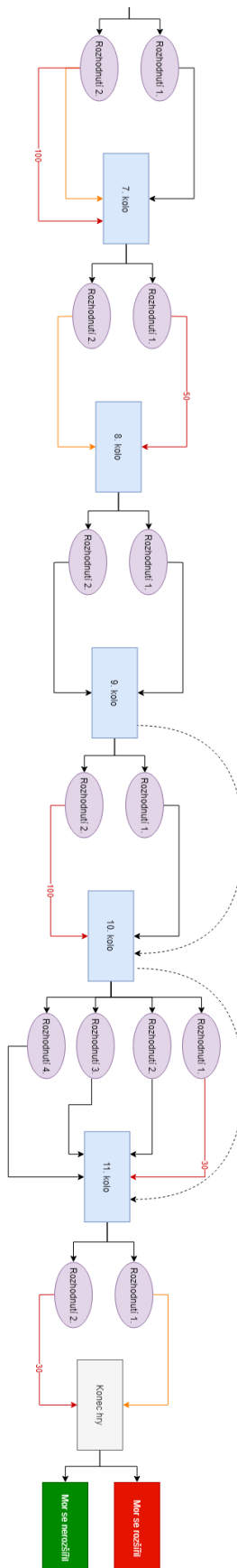


## Příloha č. 2 – Diagram mechaniky Grimfield

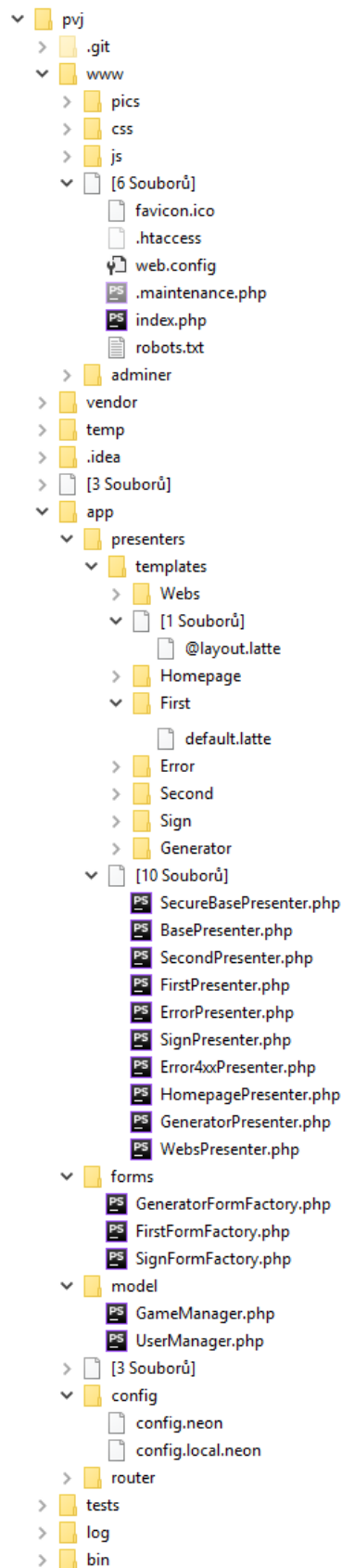




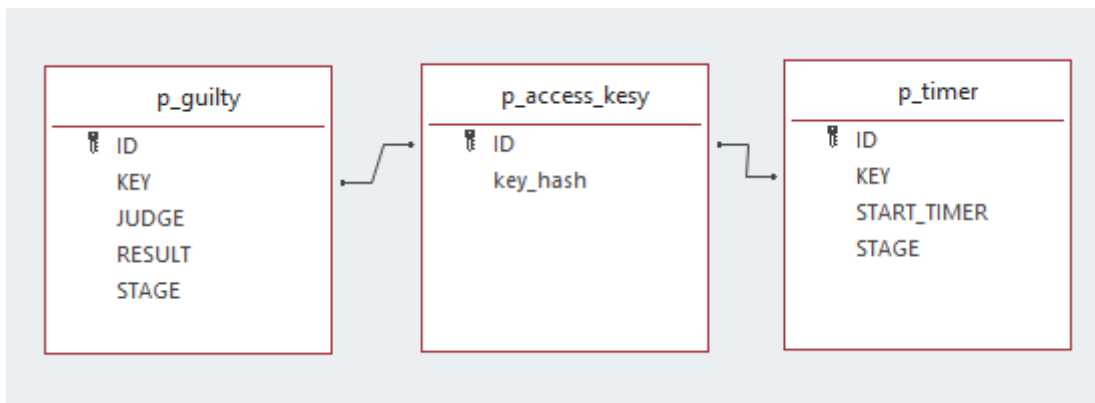
## Příloha č. 3 – Diagram mechaniky Grimfield (2)



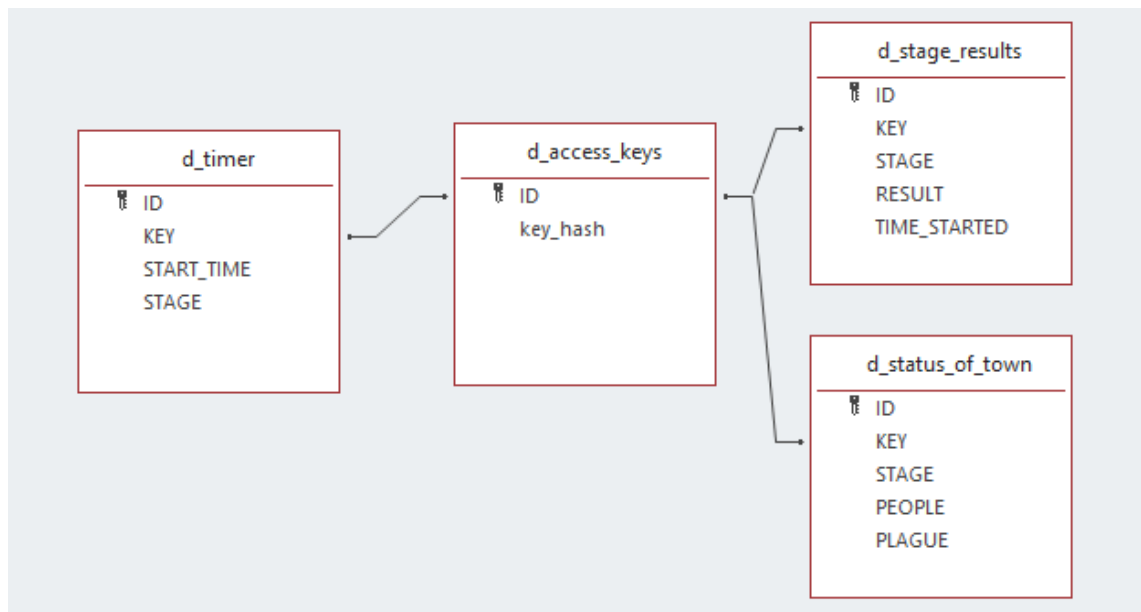
## Příloha č. 4 – Souborový strom Pvj



## Příloha č. 5 – Relační model databáze pro Pvj



## Příloha č. 6 – Relační model databáze pro Grimfield



## Příloha č. 7 – Ukázka kódu (Přesměrování)

```
public function actionDefault()
{
    $stage = $this->GameM->GetStage($this->user->id);

    if ($stage != 1) {
        $stageslibrary = array("", "First:default", "First:outcome",
"Second:default", "Second:outcome", "Third:default",
"Third:outcome", "Fourth:default", "Fourth:outcome",
"Fifth:default", "Fifth:outcome", "Sixth:default", "Sixth:outcome",
"Seventh:default", "Seventh:outcome", "Eighth:default",
"Eighth:outcome", "Ninth:default", "Ninth:outcome",
"Tenth:default", "Tenth:outcome", "Eleventh:default",
"Eleventh:outcome", "Result:default");
        $newstage = $stageslibrary[$stage];

        if ($newstage == "Result:default") {
            $this->flashMessage("The game has already ended. You
have been redirected to the summary page.");
            $this->redirect('Result:default');
        } else {
            $this->flashMessage("The game is already in a different
phase. You've been redirected to the right part.");
            $this->redirect($newstage);
        }
    }
}
```

## Příloha č. 8 – Ukázka kódu (Konec kola)

```
public function formSucceeded(Form $form, $values)
{
    $this->GameM->MoveNextStage(20, $this->user->id);
    $submit = $form->isSubmitted()->getName();
    if ($submit == 'send') {
        if ($values['result'] == 'one') {
            $this->GameM->KillPeople($this->user->id, 30);
            $this->GameM->InsertResult($this->user->id, 10, 1);
        } else if ($values['result'] == 'two') {
            $this->GameM->InsertResult($this->user->id, 10, 2);
        } else if ($values['result'] == 'three') {
            $this->GameM->InsertResult($this->user->id, 10, 3);
        } else {
            $this->GameM->InsertResult($this->user->id, 10, 4);
        }
    } else {
        $this->GameM->InsertResult($this->user->id, 10, 0);
    }
}
```

## Příloha č. 9 – Ukázka kódu (Statistika případů)

```
<div class="sidebar_item">
  <h3>Statistics</h3>
  <div id="cases">
    <p>Juries that tried to crack the case so far:
    {$allgames}</p>
    <p>Suceeded: {$wingames}</p>
    <p>Failed: {$lostgames}</p>
  </div>

  public function beforeRender()
  {
    $allgames = $this->GameM->getNumberOfGames();
    $wingames = $this->GameM->getNumberOfWinGames();
    $lostgames = ($allgames-$wingames);

    $this->template->lostgames = $lostgames;
    $this->template->allgames = $allgames;
    $this->template->wingames = $wingames;
  }
}
```

# Příloha č. 10 – Vzor formuláře zpětné vazby

## Feedback – web app PvJ

This survey was made to get some relevant feedback regarding the web application for the game called People v. Johnson. Please, answer truthfully. The results will be used for a bachelor thesis about this app.

Thank you for your answers

\*Povinné pole

### 1. Gender \*

Označte jen jednu elipsu.

- Male  
 Female

### 2. Age \*

\_\_\_\_\_

### 3. Nationality \*

Označte jen jednu elipsu.

- Czech  
 Foreigner

### 4. How would you rate the whole app? \*

1 – app is completely alright; 5 - app is  
Označte jen jednu elipsu.

- 1  
 2  
 3  
 4  
 5

### 5. How would you rate the graphic design? \*

1 - It's alright for this type of game; 5 - It's totally ugly  
Označte jen jednu elipsu.

- 1  
 2  
 3  
 4  
 5

### 6. Was there any problem with the app? \*

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

### 7. Do you have any suggestions for upgrading the app?

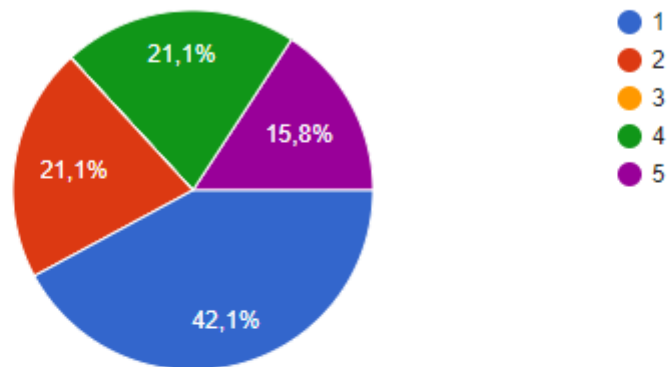
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_



## Příloha č. 11 – Grafy výsledků zpětné vazby PvJ

How would you rate the whole app?

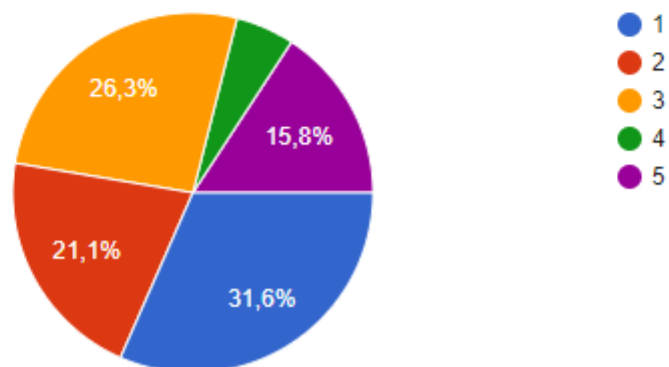
19 odpovědí



1 – app is completely alright; 5 - app is unusable

How would you rate the graphic design?

19 odpovědí

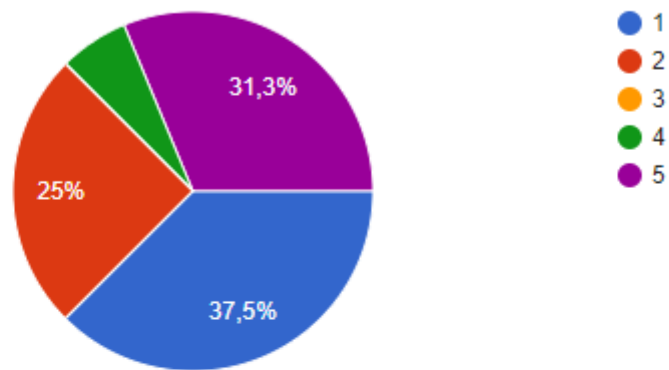


1 - It's alright for this type of game; 5 - It's totally ugly

## Příloha č. 12 – Grafy výsledků zpětné vazby Grimfield

How would you rate the whole app?

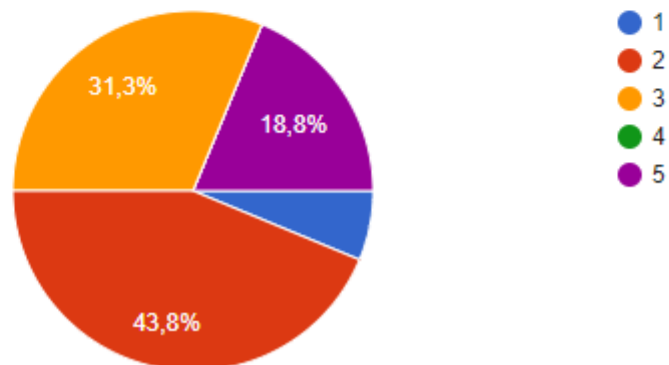
16 odpovědí



1 – app is completely alright; 5 - app is unusable

How would you rate the graphic design?

16 odpovědí



1 - It's alright for this type of game; 5 - It's totally ugly