

**Gymnázium, Praha 6, Arabská 14**

Arabská 14, Praha 6, 160 00

## **Ročníková práce**



2020

Martin Vondrášek

**Gymnázium, Praha 6, Arabská 14**

Arabská 14, Praha 6, 160 00

## **Ročníková práce**

**Předmět:** Programování

**Téma:** City guesser

**Autor:** Martin Vondrášek

**Třída:** 4. E

**Školní rok:** 2019/2020

**Vedoucí poradce:** Mgr. Jan Lána

**Třídní učitel:** Mgr. Jana Urzová, Ph.D.

# 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.

Dne: ..... Podpis: .....

## Anotace

Tato ročníková práce je věnována vytvoření hry, ve které je cílem najít přibližnou polohu nějakého města. Aplikace by měla ukázat, o kolik kilometrů jsme se netrefili. Kvíz bude obsahovat mapu, na které budete moci pohybovat značkami. Program by měl být napsán v JavaScriptu. Mapy budou dány do programu pomocí API. Výsledky budou dány do statistik.

## Anotace v anglickém jazyce

This term work is about creating a game in which the goal is to find the location of some cities. The application should show how different was our guess. The quiz will include a map on which you can move with the markers. The program should be written in JavaScript. Maps will be put into the program using the API. The results will be put into statistics.

# Obsah

1. Úvod.....	5
2. Ovládání hry.....	6
3. CodeSandbox.....	7
4. MapTiler a Mapbox GL JS.....	8
5. Databáze.....	9
5.1 JSON .....	9
5.2 Java.....	9
6. Výpočty.....	10
6.1 Vzdálenost.....	10
6.2 Body.....	11
7. Funkce tlačítek .....	12
8. Vzhled programu.....	13
8.1 CSS.....	13
8.2 Malování.....	13
9. Závěr .....	14
10. Seznam obrázků .....	15
11. Zdroje .....	16
12. Příloha .....	17

# 1. Úvod

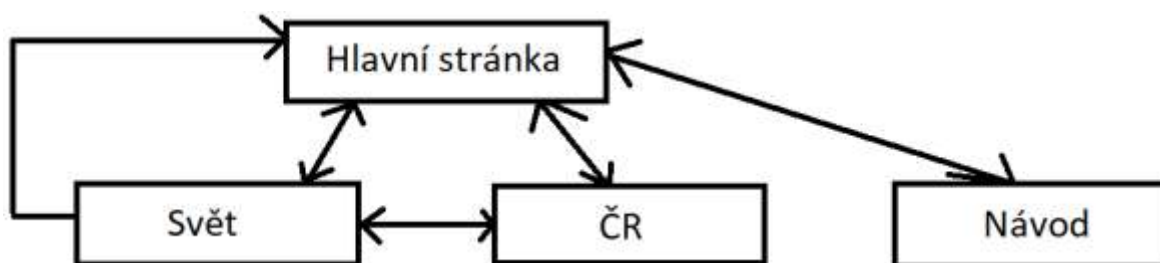
Cílem mé ročníkové práce je vytvořit program, který bude moc být používán k procvičování polohy některých měst. Studenti mají v dnešní době problémy v zeměpise, především v orientaci na mapě, proto by mohl tento program problém vyřešit.

K tématu projektu mě přivedla práce již z druhého ročníku gymnázia, kde i zde jsem pracoval s mapami. Ta pojednávala o aplikacích s mapami a umožňovala najít čtenáři pro něj tu pravou. Mou inspirací je také hra GeoGuessr, v níž je hlavním cílem najít polohu. Hledaná pozice však není město, ale náhodná pozice z Google street view. Domnívám se, že se moje znalosti s Hypertext Markup Language (html), JavaScriptem (JS) a JavaScript Object Notation (JSON) zlepší a budu je moc využít v mé budoucí kariéře. V práci je vytvořena také grafika pomocí kaskádových stylů (CSS) a obrázků, které byly vytvořeny v malování a jsou součástí programu. K dokončení práce mě motivuje fakt, že touto aplikací budu moci pomoci lidem s problémy v zeměpise a ukončení mé středoškolské výuky. Celá dokumentace je doplněna obrázky, jež vám pomohou v orientaci v kódu, který jsem naprogramoval.

Během své práce jsem narazil na značnou část problémů, které jsem byl nucen překonat. Řešit je bylo potřeba z různých stran pohledu a zvolit různé postupy k řešení. Dělo se to hlavně těsně před dokončením práce. Problematika stále přibývala, jako například dostat hodnotu z jedné html stránky na druhou. Rozhodl jsem se proto zrychlit a svůj projekt dotáhnout do cíle.

## 2. Ovládání hry

V této kapitole si důkladně popíšeme ovládání hry. Na hlavní stránce se nachází tři tlačítka. První dvě slouží do vstupu do hry. Pod nejspodnějším tlačítkem s názvem GUIDE se nachází podobný text tohoto ražení, kde se může každý hráč dozvědět více o pokynech k aplikaci. „Game - Česko“ obsahuje hru, kde je cílem najít nějaké z 50 českých měst. Oproti v tlačítku „Game - Capitals“ se nacházejí hlavní města celého světa, kde máte celkem možnost narazit na 243 měst. Teď již k samotné hře. Po kliknutí na jedno z těchto dvou tlačítek s názvem Game vás program přesměruje na slepou mapu, na které jsou pouze vidět hranice států, silnice a vodstvo. Tyto prvky mohou pomoci uživateli, v lepší orientaci a v zapamatovatelnosti. Veškeré názvy měst, vesnic, silnic, řek, moří, atd. jsou schovány. Uprostřed stránky v horní části se nachází panel, ve kterém je ukryt název města, který je potřeba najít. V levém horním rohu se nachází počet vašich nasbíraných bodů a v pravém tlačítko, které vás přesměruje zpět do hlavního menu. V pravém dolním koutu se nacházejí 3 tlačítka, která jsou umístěna nad sebou. Horní dvě přepínají v průběhu hry města česká a hlavní města celého světa. Toto nastavení však platí až od dalšího města, aby si nemohl uživatel vybírat mezi dvěma městy během každého kola. Poslední tlačítko slouží k uhodnutí pozice města. Na mapě se nachází oranžový bod, se kterým lze pohybovat. Hráč by ho měl nastavit do polohy, kde si myslí, že se nachází město v rámečku nahoře uprostřed. Po kliknutí na tlačítko „Make guess!!“ se v levém dolním rohu objeví, o kolik se daný uživatel spletl. Tlačítkem „Next!!“, které se objevilo v pravém dolním rohu, se dostaneme k dalšímu městu. Každý hráč má celkem 10 pokusů, aby předvedl, co umí. Na konci se uživateli ukáže finální skóre a může si zahrát znovu.

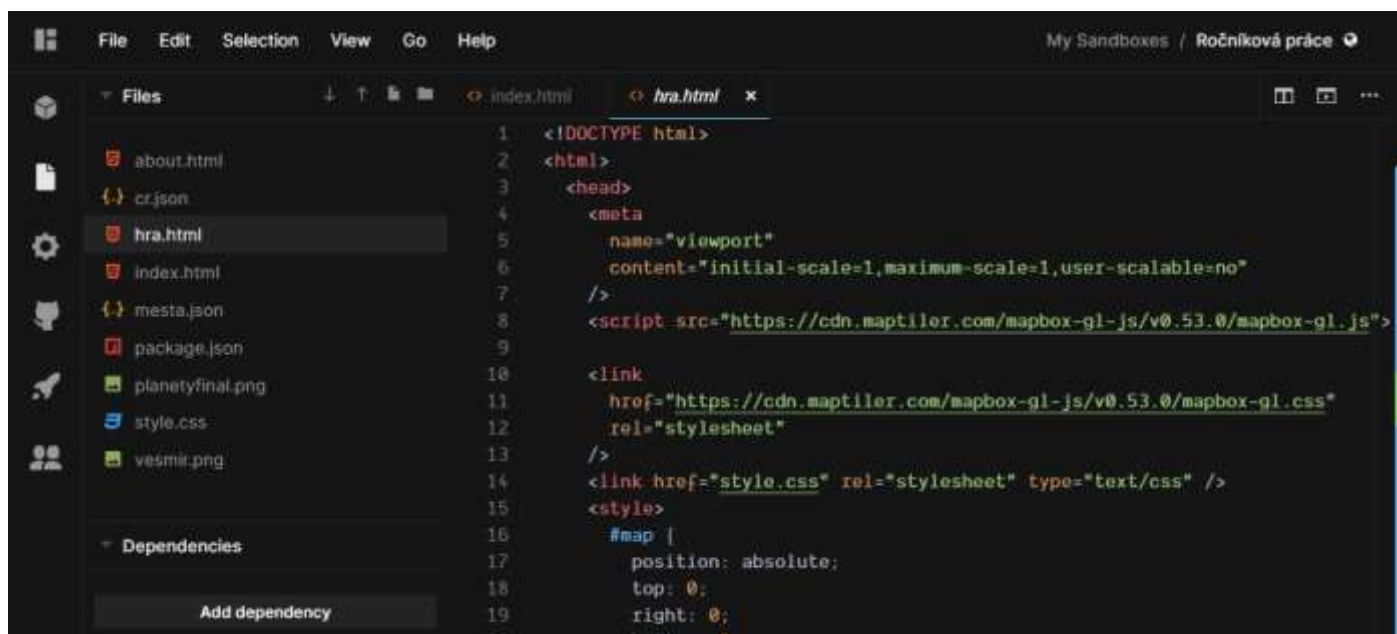


Obrázek 1 Schéma programu



### 3. CodeSandbox

CodeSandbox je online editor s prototypovými nástroji, který nám umožňuje rychle vytvářet, ukládat a sdílet webové aplikace. S tímto editorem jsem se setkal ve 3. Ročníku v hodinách webových aplikací. CodeSandbox mi poskytl rychlé testování mého programu, otevřít můj kód naprosto kdekoli a snadné ukládání práce na GitHub. Většina práce byla naprogramována v tomto editoru, ke zbytku byla použita Java, nebo jiný online editor. Pokud se v programu nachází chyba, CodeSandbox ji nahlásí. Výhodou je, že se nemusí nic stahovat, vše probíhá přes internet. Editor zvýrazňuje jednotlivé části kódu různými barvami, pro uživatele je to velmi přehledné. K přihlášení na webovou stránku vám stačí pouze email nebo účet na GitHub, služby jsou propojené.



Obrázek 2 vizualizace CodeSandboxu, Codesandbox (28. 03. 2020)

## 4. MapTiler a Mapbox GL JS

MapTiler je platforma s mapami, která uživateli umožňuje rychle publikovat online zvětšovatelné mapy pro webové aplikace, mobilní zařízení a 3D vizualizace. Tento online software byl v práci využit k vygenerování mapy. MapTiler umožňuje uživateli nekonečnou řadu možností, stejně jako Google Maps API, jen s tím rozdílem, že je zdarma. Na stránce jsem si vytvořil novou mapu, kterou jsem si následně přizpůsobil mému vkusu a potřebám. Odstranil jsem názvy měst, upravil jsem barvu krajiny, vody, cest a správních celků. Celý projekt jsem následně uložil, názvy je možné i převést do českého jazyka, avšak v mém programu není tato možnost potřeba. Poté jsem svou mapu zkopíroval do svého projektu pomocí mapovací JavaScript knihovny Mapbox GL JS, která je postavena na WebGL. Kód je vygenerován sám MapTilerem, do vlastního programu stačí pouze přetáhnout. Tento postup mi dále umožnil snadnou práci při použití souřadnic. Pomocí knihovny Mapbox GL JS jsem do svého programu mohl přidat značky, se kterými lze pohybovat a nastavit jim pozice. Souřadnice jednotlivých značek jsem dále mohl použít například k výpočtu vzdáleností. Práce s knihovnou byla zprvu složitější, pochopení principu fungování trvalo nějakou chvíli. Zpětně je práce celkem jednoduchá.



Obrázek 3 Možnosti MapTileru, MapTiler (28. 03. 2020)


## 5. Databáze

### 5.1 JSON

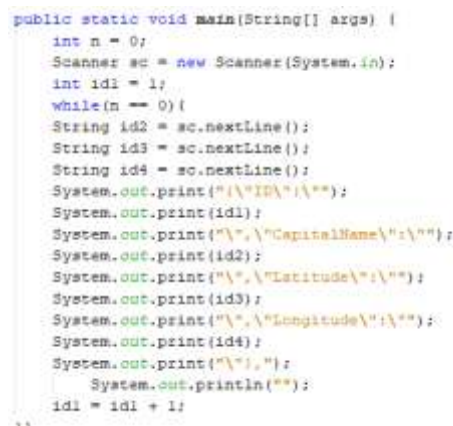
JavaScript Object Notation, v zkratce JSON, je způsob zápisu dat, který umí pojmout pole hodnot, objekty a hodnoty. Výstupem se stává řetězec, který je tvořen složenými závorkami a uvozovkami. Kódování JSON je UTF-8, a proto může uživatel používat českou klávesnici a další speciální znaky. Tento zápis dat byl v ročníkové práci použit k vytvoření databáze. Do programu byla přidána v následujícím tvaru - { "ID": "1", "CapitalName": "Břeclav", "Latitude": "48.75897", "Longitude": "16.88203" }. ID je pouze orientační pro pozorovatele, JSON sám čísluje jednotlivé položky od 0, já jsem je však ještě navíc seřadil od 1, ale dále nejsou hodnoty používány. CapitalName nese název po městech českých i světových. Latitude je zeměpisná šířka a Longitude je zeměpisná délka. V JavaScriptu jsem šahal do databáze pomocí příkazu fetch. Hra bude pouze fungovat v online editoru, protože databáze po stáhnutí nemá odkaz http nebo https.

### 5.2 Java

Určitě se nikdo nechce dělat ručně s databází, proto jsem naprogramoval jednoduchý kód v Javě, který nám úkol velmi usnadní. Vstupem jsou tři řádky v tomto pořadí – město, zeměpisná šířka a zeměpisná délka. Do programu vkládáme najednou, ID se doplňuje vzestupně samo. Aplikace za nás doplní automaticky potřebné závorky, uvozovky, dvojtečky, čárky a text. Výsledek poté stačí vložit pouze do souboru s koncovkou .json.

A screenshot of a code editor showing a JSON array. The first object has the following properties: "ID" with value "1", "CapitalName" with value "Břeclav", "Latitude" with value "48.75897", and "Longitude" with value "16.88203". The code is color-coded: strings are in red, numbers in green, and punctuation in white.

Obrázek 5 Databáze, CodeSandbox (28. 03. 2020)

A screenshot of Java code in a code editor. The code defines a main method that takes an array of arguments. It uses a Scanner to read input from the command line. It then generates a JSON object by concatenating strings for ID, CapitalName, Latitude, and Longitude. The ID is incremented for each iteration. The code is color-coded: keywords in blue, strings in red, and other tokens in white.

Obrázek 4 Generace databáze pomocí Javy, Java (28. 03. 2020)

## 6. Výpočty

### 6.1 Vzdálenost

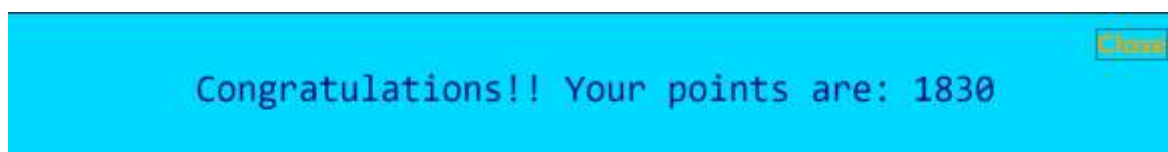
Výpočet vzdálenosti je velmi složitá. Tento postup bych sám bohužel nevymyslel, nápad je zkopírován z internetu, zde v práci je zdroj přímo na stránky. Nejdříve potřebujeme zjistit zeměpisnou šířku a zeměpisnou délku obou měřených bodů. V mém programu získám jednu pozici pomocí databáze, druhou pomocí knihovny Mapbox GL JS. Další údaj, který je nutný, je poloměr Země, což je 6371 km. Vzdálenost pak vypočítáme pomocí složitého vzorce zvaného „Haversine“. Tento vzorec je určen takto:  $Vzdálenost = R * (\sin^2(\varphi_2/2 - \varphi_1/2) + \cos(\varphi_1) * \cos(\varphi_2) * \sin^2(\lambda_2/2 - \lambda_1/2))$ , kde R je poloměr země,  $\varphi_1$  a  $\varphi_2$  jsou zeměpisné šířky v radiánech,  $\lambda_1$  a  $\lambda_2$  jsou zeměpisné délky v radiánech. Hodnoty v radiánech získáme vynásobením čísla ( $\varphi_1$ ,  $\varphi_2$ ,  $\lambda_1$  a  $\lambda_2$ ) a čísla  $\pi$ , výsledek tohoto vydělíme ještě 180. První využití tohoto vzorce se datuje na začátek 19. století. Ve svém programu je i zajištěn převod na metry, pokud je vzdálenost menší než 1 kilometr.

```
function distance() {  
  //změření vzdálenosti  
  var lngLat1 = marker.getLngLat();  
  var lngLat2 = marker2.getLngLat();  
  var lat1 = lngLat1.lat;  
  var lon1 = lngLat1.lng;  
  var lat2 = lngLat2.lat;  
  var lon2 = lngLat2.lng;  
  var R = 6371; // km  
  var dLat = ((lat2 - lat1) * Math.PI) / 180;  
  var dLon = ((lon2 - lon1) * Math.PI) / 180;  
  var a =  
    Math.pow(Math.sin(dLat / 2), 2) +  
    Math.cos((lat1 * Math.PI) / 180) *  
    Math.cos((lat2 * Math.PI) / 180) *  
    Math.pow(Math.sin(dLon / 2), 2);  
  var d = R * (2 * Math.atan2(Math.sqrt(a), Math.sqrt(1 - a)));  
  var dis = Math.round(d);  
  if (d > 1) d = Math.round(d) + "km";  
  else if (d <= 1) d = Math.round(d * 1000) + "m";  
}
```

Obrázek 6 Funkce, která měří vzdálenost, CodeSandbox (28. 03. 2020)

## 6.2 Body

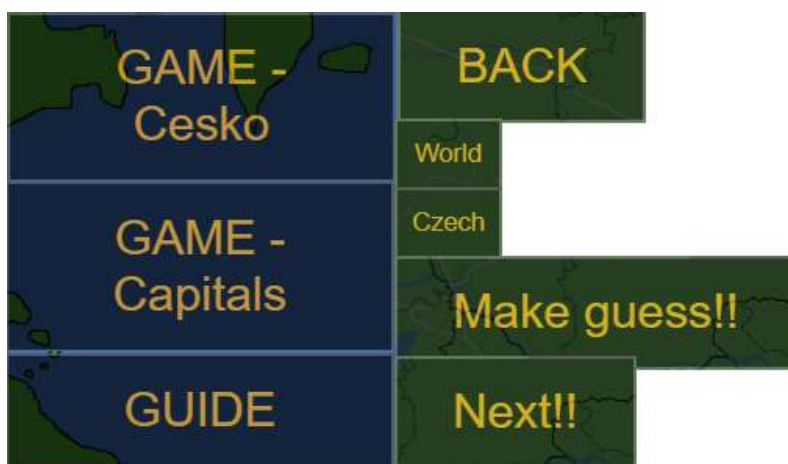
Další hodnotou, která je hojně využívána, je skóre hráče. Maximální body, které může uživatel získat, jsou 2000. Z každého kola je maximálně 200 bodů, kol je 10, hráč může však hru hrát opakovaně. Bodování pro svět a pro Česko je různé. Najít na mapě české město je velmi snazší, než najít nějaké hlavní město, o kterém jste dodnes vůbec nevěděli, že existuje. Proto jsem vytvořil speciální bodování pro obě možnosti. U České republiky dostane hráč plný počet bodů, když je vzdálenost menší než 8 kilometrů. Dále se za každý kilometr ztrácí 10 bodů, což znamená, že vzdálenost k získání bodů musí být menší než 28 km. Naopak u měst celého světa je plný počet bodů do 15 kilometrů, za 16 kilometrů a více se odečítá každý bod navíc. Každé město je trochu jinak velké, proto jsem nastavil plné body do konstantních vzdáleností, které odpovídají poloměru velkých měst. Česká města jsou menší, a i přestože dáte na kraj města značku, program vám přidělí 200 bodů. Stejně je to i u hlavních měst, kde je hranice posunuta více nahoru. Splést se je velmi snadné, je nutná pozornost. Vaše skóre naleznete v levé horní části obrazovky, na konci hry se vám zobrazí, ve vyskakovacím okénku, vaše finální skóre.



Obrázek 7 Možné finální skóre, City guesser (28. 03. 2020)

## 7. Funkce tlačítek

V programu se nachází několik typů tlačítek, které dělají různé funkce. Dají se rozdělit na 3 typy. Prvním typem je klasické tlačítko, které nás někam přesměruje. Jsou pojmenovány například – „BACK“, „GAME-...“, „GUIDE“. Obvykle obsahují jenom odkaz, kromě tlačítek „GAME-...“. Jsou to dvě tlačítka, které ještě uloží hodnotu 1 (pro Česko), nebo 2 (pro světová hlavní města). Tyto dvě čísla se uloží do lokální paměti. Jednu hodnotu získávám při kliknutí na hru, kde se rozhoduje, jestli hraju hru s českými městy nebo světovými. Mezi druhý typ patří tlačítka „Make Guess!!“ a „Next!!“. První z nich přidá na mapu značku pomocí knihovny Mapbox GL JS a databáze, další funkce změří vzdálenost dvou bodů (viz. kapitola 6.1). „NEXT!!“ náhodně vybere další město z databáze, hlídá konec hry a přepíná scénu mezi jednotlivými pokusy (město, skóre a skrývá vzdálenost). Posledním typem tlačítek je přepínací tlačítko, to nám umožňuje v průběhu hry přepínat mezi českou a světovou verzí. Nacházejí se ve hře nad tlačítky druhého typu.



Obrázek 8 Vzhled tlačítek, City guesser (28. 03. 2020)



## 8. Vzhled programu

### 8.1 CSS

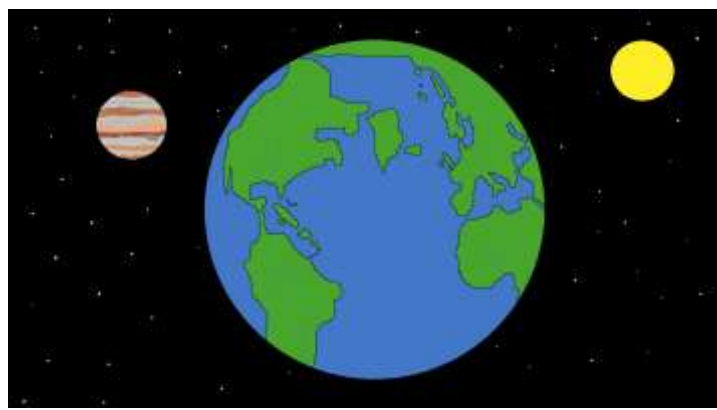
Kaskádové styly, ve zkratce CSS, jsou v ročníkové práci využity v hojném počtu. Zajišťují celou grafiku, výsledek potom nevypadá jenom jako pár černobílých tlačítek a textu. V CSS můžeme nastavovat font textu, velikost, barvu, odsazení. Všechno není jenom kolem textu, pomocí stylů můžeme i upravit vzhled tlačítek, jejich průhlednost, tloušťku okrajů, pozici na obrazovce a spoustu dalších vychytávek. Všechno z uvedených věcí je obsaženo v souboru.

### 8.2 Malování

Každý program by měl být trochu vzhledný, a proto jsem využil programu malování a do pozadí jsem vložil obrázek vesmíru se Zemí, Sluncem, Jupiterem a hvězdami. Vesmír je opravdu nekonečný, obrázek byl bohužel samotný menší než rozlišení obrazovky, a tak jsem vytvořil nekonečný cyklus samostatného vesmíru.

```
button4 {  
  background-color: ■ black;  
  color: ■ orange;  
  padding: 15px 32px;  
  text-align: center;  
  font-size: 28px;  
  position: absolute;  
  right: 34%;  
  left: 34%;  
  top: 60%;  
  opacity: 70%;  
  border: 2px solid ■ #555555;  
}  
button4:hover {  
  background-color: ■ orange;  
  color: ■ purple;  
}
```

Obrázek 10 část kódu CSS,  
CodeSandbox (28. 03. 2020)



Obrázek 9 Obrázek pozadí

## 9. Závěr

Celý projekt a jeho zpracování bylo náročné na čas, na množství informací, ale především na to, jak tyto objemy dat zapsat do ročníkové práce. Abych odevzdal své dílo v řádném termínu, musel jsem si svou práci zorganizovat a naplánovat. Při psaní a programování se čas od času nedaří a dlouhý čas strávený u počítače tomu moc nepomáhá.

Za velmi důležité pokládám rozhodnutí výběru tématu, které mě zajímá a u kterého bych se neunudil. Během zpracovávání tématu jsem využil znalosti, které jsem získal ve škole. Jednalo se zvláště o předměty, jako jsou webové aplikace, zeměpis a český jazyk. Celá práce mi přinesla hodně nových a cenných informací, které byly k dokončení práce potřeba. Čerpal jsem také ze zkušeností nabytých z předchozích ročníkových projektů. Během mého úsilí dokončit práci, jsem narazil na jediný problém, který by mi znemožnil dovést mé dílo do očekávaného cíle. Ani po hodinách strávených u problému se mi nepodařilo naprogramovat server, se kterým by bylo možné ukládání skóre všech hráčů. Při vytváření obrázků a při úpravě celého dokumentu jsem využíval svoji představivost a náležitá kritéria. Nejtěžší částí celé práce bylo napsání a upravení dokumentace. Dovést ji do takové podoby, aby odpovídala znalostem studentovi čtvrtého ročníku čtyřletého gymnázia. S výsledným zpracováním jsem spokojen, naplnil téměř moje očekávání.

Cílem mé ročníkové práce bylo vytvořit program, který by mohl být využitý jako pomůcka při výuce zeměpisu. Tento cíl se mi podařilo splnit. Už se těším na další zkušenosti s webovými stránkami a s další prací v programovacích jazycích.



## 10. Seznam obrázků

Obrázek 1 Schéma programu .....	6
Obrázek 2 vizualizace CodeSandboxu, Codesandbox (28. 03. 2020).....	7
Obrázek 3 Možnosti MapTileru, MapTiler (28. 03. 2020).....	8
Obrázek 5 Generace databáze pomocí Javy, Java (28. 03. 2020) .....	9
Obrázek 4 Databáze, CodeSandbox (28. 03. 2020).....	9
Obrázek 6 Funkce, která měří vzdálenost, CodeSandbox (28. 03. 2020) .....	10
Obrázek 7 Možné finální skóre, City guesser (28. 03. 2020).....	11
Obrázek 8 Vzhled tlačítek, City guesser (28. 03. 2020).....	12
Obrázek 10 Obrázek pozadí .....	13
Obrázek 9 část kódu CSS, CodeSandbox (28. 03. 2020) .....	13

## 11. Zdroje

The\_coder. Výpočet vzdálenosti pomocí zeměpisné šířky a délky. *Snipplr* [online]. 2010 [cit. 2020-04-28]. Dostupné z: <https://snipplr.com/view/25479/calculate-distance-between-two-points-with-latitude-and-longitude-coordinates>

ČERVENÝ, Pavel. Měření vzdálenosti na mapách. *Metodický portál RVP* [online]. Česko, 2007 [cit. 2020-04-28]. Dostupné z: <https://clanky.rvp.cz/clanek/k/ZL/1638/JAK-SPRAVNE-MERIT-VZDALENOSTI-NA-MAPACH.html/>

Wikipedia contributors. Haversine formula [online]. Wikipedia, The Free Encyclopedia; 2020 Apr 26, [cit. 2020-04-28]. Dostupné z: [https://en.wikipedia.org/w/index.php?title=Haversine\\_formula&oldid=953270281](https://en.wikipedia.org/w/index.php?title=Haversine_formula&oldid=953270281).

How To Create a Modal Box. *W3Schools* [online]. Refsnes Data [cit. 2020-04-28]. Dostupné z: [https://www.w3schools.com/howto/howto\\_css\\_modals.asp](https://www.w3schools.com/howto/howto_css_modals.asp)

List of Countries and Capitals. *Techslides* [online]. Data Visualization, 2013 [cit. 2020-04-28]. Dostupné z: <http://techslides.com/list-of-countries-and-capitals>

List of Countries and Capitals. *MapTiler* [online]. MapTiler, 2020 [cit. 2020-04-28]. Dostupné z: <https://www.maptiler.com/>

## 12. Příloha

Odkaz na funkční program: <https://e16yv.csb.app/>

Odkaz na Codesandbox: <https://codesandbox.io/s/rocnikova-prace-e16yv>