

STŘEDNÍ PRŮMYSLOVÁ ŠKOLA

MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**



Ondřej Naďo

Mladá Boleslav 2023

# STŘEDNÍ PRŮMYSLOVÁ ŠKOLA MLADÁ BOLESLAV

## ROČNÍKOVÁ PRÁCE

**Autor: Ondřej Naďo**

**Studijní obor: 18-20-M/01 Informační technologie**

**Vedoucí práce: Jan Peča**

**Mladá Boleslav 2023**

## Obsah

Obsah.....	3
1 Úvod .....	6
2 Programovací jazyky .....	7
2.1 JavaScript.....	7
2.2 HTML .....	7
2.2.1 Canvas.....	8
2.3 CSS .....	9
3 Obsah práce.....	10
3.1 Hráč .....	10
3.2 Hlavní menu .....	11
3.3 Hrací pole .....	12
3.3.1 Čudlík .....	12
3.3.2 Ukazatel úrovní.....	12
3.3.3 Ukazatel smrtí.....	13

3.3.4	Pozadí hracího pole .....	13
3.4	Nepřátelé.....	14
3.5	Bariéry .....	15
3.6	Pohyb objektů .....	16
3.6.1	Pohyb hráče .....	16
3.6.2	Pohyb nepřítele .....	17
4	Závěr .....	18
5	Přílohy.....	19
5.1	Seznam obrázků .....	19
5.2	Zdroje.....	19

### **Prohlášení**

Prohlašuji, že jsem svou ročníkovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

Nemám závažný důvod proti zpřístupňování této ročníkové práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V Mladé Boleslavi dne ..... podpis: .....



# 1 Úvod

Smyslem této práce je vytvořit textový dokument o hře The World's Hardest Game. Dokument by měl seznámit čtenáře s problematikou práce a představit hru, která vyžaduje od hráče strategické myšlení a dobrou reakční schopnost. Hra představuje výzvu pro hráče, ale taky příjemnou zábavu pro odreagování.

The World's Hardest Game využívá dvourozměrné herní prostředí s přednastavenou pevně danou herní kamerou, se kterou se nedá nijak manipulovat, ani se nijak nehýbe. Základním principem hry je dostat kostičku ze startovní a zároveň bezpečné zóny do druhé bezpečné zóny. Bezpečné zóny jsou zeleně označené úseky hry. Po dosažení hranice bezpečné zóny se spustí další level, nebo skončí hra. Pohyb kostičkou je pomocí kláves WASD. Nesmíte se dotknout pohybujících se nepřátel. Dotykem nepřítele není konec hry, ale kostička se vrátí zpátky na startovní pozici a musíte začít zase od začátku úrovně.

Hra byla vytvořena v rámci ročníkového projektu. Jedná se o jeden z mých prvních větších projektů, který měl ověřit mou schopnost vytvořit aplikaci dle daného zadání. The World's Hardest Game jsem si vybral za téma kvůli zajímavé náplni hry a její zvladatelnosti provedení.



## 2 Programovací jazyky

Značná část práce je vytvořena v programovacím jazyce JavaScript, který byl vybrán na začátku zadání práce a při vybrání tématu, z důvodů jeho jednoduchosti a čitelnosti. Práce je doplněna dalšíma dvěma programovacíma jazyky. Těmi jsou CSS a HTML. Tyto tři jazyky spolu skvěle spolupracují a díky hypertextovým odkazům jsou spolu propojeny.

### 2.1 JavaScript

JavaScript je objektově orientovaný programovací jazyk, který je zapisován ve formě kódu, ten vykonává určitý úkol. Dá se zapisovat přímo do HTML, což je jeho velká výhoda a dělá ho to jednoduchým. Jeho velké a nejpodstatnější využití je při vytváření webových stránek. Při vývoji této aplikace se hodí, díky jeho skvělé a jednoduché práci s objekty<sup>1</sup>. JavaScript běží na straně klienta, takže jsou všechny aplikace spuštěny v prohlížeči u uživatele.



Obrázek 1 Logo JavaScriptu: <https://github.com/Shweta2024/JavaScript-Projects>

### 2.2 HTML

HTML je zkratka pro HyperText Markup Language, v překladu Hypertextový značkový jazyk. Používá se k vytváření obsahu webové stránky, obsahem mohou být různé

---

<sup>1</sup> Instance od třídy



tabulky, obrázky, texty a další prvky. Pomocí hypertextových odkazů propojuje jednotlivé webové stránky a programovací jazyky. V téhle aplikaci byl použit pro vytvoření její vizuální stránky, která byla nastýlována pomocí CSS.<sup>2</sup>



Obrázek 2 Logo HTML: [https://en.wikipedia.org/wiki/HTML#/media/File:HTML5\\_logo\\_and\\_wordmark.svg](https://en.wikipedia.org/wiki/HTML#/media/File:HTML5_logo_and_wordmark.svg)

### 2.2.1 Canvas

Canvas je HTML element<sup>3</sup>, který umožňuje vytvářet a renderovat grafiku pomocí programování v JavaScriptu. Canvas se používá pro tvorbu interaktivních grafických aplikací, jako jsou hry, animace a vizualizace dat. Canvas umožňuje kreslit obrazce, text, obrázky a další grafické prvky<sup>4</sup> na plátno, které následně lze upravovat a animovat. Canvas poskytuje metody pro kreslení úseček, oblouků, čtverců a mnoho dalších tvarů. Také lze vytvářet stíny a další efekty.

Tato práce je dost postavená na funkcích Canvasu. Využívá funkce vykreslování obrazců na plátno, jejich následné vymazávání a překreslování. Díky těmto funkcím funguje vizuální pohyb těles na plátně.

---

<sup>2</sup> Inspirace ze stránky: <https://www.strafelda.cz/html>

<sup>3</sup> Dokument psaný v programovacím jazyce

<sup>4</sup> Virtuální objekt, který slouží k vykreslení





## 2.3 CSS

CSS je zkratka Cascading Style Sheets, v překladu tabulky kaskádových stylů. Je to jazyk používaný v programování pro popis vzhledu a stylu webových stránek. CSS umožňuje programátorům definovat různé styly pro různé prvky webové stránky, jako jsou barvy, velikosti, polohy, písma, pozadí a další vlastnosti. Tato úprava je definována pomocí speciálních syntaxí, které se aplikují na HTML elementy. Díky CSS jsou webové stránky snadno čitelné a vizuálně přitažlivé.<sup>5</sup>



Obrázek 3 Logo CSS: [https://en.wikipedia.org/wiki/CSS#/media/File:CSS3\\_logo\\_and\\_wordmark.svg](https://en.wikipedia.org/wiki/CSS#/media/File:CSS3_logo_and_wordmark.svg)

---

<sup>5</sup>Inspirace ze stránky: <https://blog.shoptet.cz/co-je-nutne-vedet-o-css-1-dil>.



## 3 Obsah práce

Práce popisuje hru The World's Hardest Game, která je určená pro jednoho hráče. The World's Hardest Game se odehrává ve dvourozměrném virtuálním prostředí. Hráč ovládá červený čtvereček a jeho úkolem je dostat čtvereček ze startovního zeleného pole, do druhého zeleného pole, přitom nesmí být chycen nepřátelskými kroužky.

Hra obsahuje více než jednu úroveň a jsou navrženy tak, aby byly náročné a hráč musel využít své strategické a reakční schopnosti, aby je dokončil. Každá úroveň je rozdělena na části, ve kterých se musí hráč pohybovat tak, aby ho nechytily nepřátelské kroužky. Nepřátelské kroužky opakují svůj pohyb, který je předem daný, ale v každé úrovni se může lišit. Pokud byl hráč chycen nepřátelským kroužkem, vrátí ho to na startovní pozici a musí začít úroveň zase od začátku. Hráč se může pohybovat každým směrem, pohyb vykonává stisknutím kláves WASD, na kterých je pohyb přednastavený.

Po dohrání hry je pro hráče přirozené ji hrát zase znovu, aby ji dohráli s co nejmenším počtem smrtí a v co nejlepším čase. To dělá ze hry The World's Hardest Game velmi populární výzvu pro všechny hráče.

### 3.1 Hráč

Hráčem se v této práci rozumí červenému čtverečku s černým ohraničením, který má za úkol se dostat do zelené cílové zóny. Červená barva proto, že se tam stylově hodí a je nejvíce rozpoznatelná od okolí a od modrých nepřátel, navíc je to tak trochu symbolická barva pro tuto hru.

Hráč je tvořen přes objekt pomocí funkce v JavaScriptu a následně je vykreslován na canvas. Hráč má danou svoji výšku a šířku, která musí být vždycky stejná, jinak by nesplňoval podmínku čtverce. Dále má svoje hodnoty x a y, díky kterým se nastaví jeho poloha a nadále



se s ním dá díky těmto hodnotám pohybovat. Tyto názvy jsou dané kvůli systému souřadnic a člověk se v tom lépe vyzná. Osa x je vodorovná a tím pádem, když budete přidávat hodnotu, tak se bude čtvereček překreslovat doprava, naopak když budete ubírat hodnotu, tak doleva. Osa y je svislá což znamená, že když přidáváte hodnotu, tak se bude překreslovat čtvereček dolů a když ubírat, tak se překreslí nahoru. Osa y se v programování nechová jako v matematice, tam je to samozřejmě obráceně. Nulové hodnoty na obou osách jsou v levém horním rohu. Hráč má svoji vlastnost pro zadání barvy, ohraničení a tloušťky ohraničení.

Hráč má v kódu vytvořenou svoji samostatnou stránku, aby to bylo lépe přehledné a také díky jeho důležitosti a dohledatelnosti, jelikož se jeho hodnoty často měnily. Soubor se nazývá `player.js`, koncovka `js` se používá, aby bylo poznat s čím pracujete, takže to je zkratka JavaScriptu. Samotný hráč se musí importovat do `main.js`, neboli do stránky s hlavním kódem JavaScriptu a tam se všechny jeho hodnoty lehce nastaví a dále se s ním pracuje, jak je potřeba.



Obrázek 4: Hráč

## 3.2 Hlavní menu

Jako první hra nabídne obrazovku hlavního menu. V hlavním menu se nachází velký název hry *World's Hardest Game* a tři tlačítka. Tlačítka jsou jemně zaoblená a pomocí funkce `hover`<sup>6</sup> po najetí tlačítka myši mění barvu. Každé tlačítko má přidělený `onclick` event, díky kterému po kliknutí myši se přepne obrazovka na jinou, přidělenou obrazovku podle názvu tlačítka. Každé tlačítko má ve funkci v JavaScriptu přidělené, co má skrývat, a naopak co má volat, to zajistí že se skryjí věci, co v tu chvíli uživatel nepotřebuje a objeví se věci co požaduje. Po každém kliknutí na kterékoliv tlačítko se v levém horním rohu objeví tlačítko s nápisem `go back`<sup>7</sup>, které zajistí, aby uživatel mohl jít po kliknutí na toto tlačítko zase zpátky

---

<sup>6</sup> Funkce v CSS

<sup>7</sup> Krok zpět



do hlavního menu. Celé menu je stylizované pomocí css, a funkční díky funkcím a eventům v JavaScriptu.

### 3.3 Hrací pole

Hrací pole je samotný canvas, šířka a výška hracího pole je nastavená, aby se rovnala šířce a výšce canvasu. Hrací pole není na celou šířku a výšku obrazovky, ale jen na její část. Je zarovnané do středu obrazovky a je ohraničené černou čarou. Prostor, který vznikl kolem hracího pole, je vyplněn černou barvou. V prostoru se nachází, konkrétně v horní části čudlík pro odchod ze hry, ukazatel, v jaké se nacházíte úrovni a počítadlo smrtí. V hracím poli jsou zelené zóny a obrázek v pozadí.

#### 3.3.1 Čudlík

V levém horním rohu, v prostoru kolem hracího pole je umístěn čudlík s nápisem go back. Čudlík využívá onclick event, který dělá, že když se na něj klikne tak se něco stane, konkrétně se vrátíte zpět do hlavního menu hry. Ve funkci v JavaScriptu má přesně dáno, co se má po jeho stisknutí skrýt, a naopak co se má zase objevit. Je umístěn v levém horním rohu záměrně, aby to vypadalo že se vždycky vracíte o krok zpět, což také daný čudlík dělá.

#### 3.3.2 Ukazatel úrovní

Uprostřed obrazovky, úplně nahoře je ukazatel úrovní, který ukazuje, ve které úrovni se přesně nacházíte. Díky podmínce, která přičítá po každém dokončení úrovně jedničku, tak se mění číslo za nápisem level<sup>8</sup> podle toho, ve které úrovni se právě nacházíte. Problematikou bylo vložení textu do JavaScriptu, který se má pravidelně přičítat. Všechno se to vyřešilo díky innerHTML, který byl přidán do podmínky.

---

<sup>8</sup> úroveň



### 3.3.3 Ukazatel smrtí

Ukazatel smrtí funguje vlastně úplně na stejné bázi, jako ukazatel úrovní. Rozdíl v problematice mezi nimi byl ten, ukazatel smrtí začínal od nuly a ukazatel úrovní vždycky od jedničky. Každý z nich měl svou hodnotu danou, pomocí jejich proměnné<sup>9</sup>. Ta se samozřejmě u ukazatele smrtí nastavila, že se bude rovnat nule. Stejně jako u ukazatele úrovní se problematika vložení textu, který nabývá určitých hodnot, vyřešila pomocí innerHTML. Po každém doteku hráče s nepřítelem se přičítá jednička, problematika je řešená přes podmínku a velmi pomohlo, že oba objekty, jak hráč, tak nepřítel má svoje proměnné x a y, díky kterým je znatelné, na které pozici se v tu danou chvíli nachází.

### 3.3.4 Pozadí hracího pole

Na pozadí v hracím poli této práce se nachází obrázek šachovnicového pole. Obrázek byl zadarmo stažen z internetu a následně byl upraven. Upraven musel být z důvodu nepřehlednosti hry. Hráč ani nepřátelé nebyli pořádně vidět na hracím plátně. Z černobílého šachovnicového pole se stalo šedobílé a hra rázem prokoukla. Pro obrázky je v programu vytvořené samostatná složka img, do které se obrázky vkládají a celý program je díky tomu přehlednější. Samozřejmě se na to musí brát ohled při vytváření cesty neboli odkazu k obrázku, která je o jedno slovo delší.

Problematikou u vkládání obrázku do pozadí bylo, že se obrázek vkládal pořád až za pozadí canvasu, tudíž ho překreslovala nastavená bílá barva plátna. Byla řešena vytvořením proměnné backgroundImage (), která se rovnala novému obrázku. Vytvořením cesty, odkazu k obrázku, byla proměnná už kompletní. Dalším krokem muselo být vykreslení obrázku do pozadí canvasu. V tuto chvíli se obrázek vykresloval až pod pozadí canvasu a nebyl vidět. Pomocí metody clearRect, kterou práce s canvasem nabízí se pozadí canvasu vyčistilo a následně mohla být použita metoda pro vykreslení obrázku do canvasu. Na tenhle krok byla použita metoda drawImage () a následně už stačilo napsat do závorek metody, co za obrázek chceme vykreslit, na jakou pozici a jaké rozměry má být. Zjistit rozměry už nebylo

---

<sup>9</sup> Prvky, které nabírají různých hodnot



tak těžké, jelikož se musely rovnat rozměrům canvasu, které byly na začátku programu vloženy do proměnných `canvas.height`<sup>10</sup>, `canvas.width`<sup>11</sup>.

### 3.4 Nepřátelé

Nepřátelé jsou ve hře velmi důležitou součástí, jelikož když se dostane hráč do kontaktu s nepřítelem, tak automaticky ztrácí život a začíná v dané úrovni zase od začátku. Bez nepřátel by hráč mohl v klidu projet do cíle a hra by absolutně nemohla fungovat, a hlavně by to hráče nenaplnovalo a absolutně nemohlo bavit. Hra by ztratila i svůj podstatný název *World's Hardest Game*, svůj titul obtížnosti a strategické hry. Úkolem nepřítele je co nejvíce hráči hru ztížit.

Nepřítelem se ve hře *The World's Hardest Game* představuje modrý kroužek s černým ohraničením. Nepřítel má stejné rozměry jako hráč, ale díky jeho rozdílnému tvaru může vypadat lehce menší. Modrý kroužek se pohybuje všemi směry, podle toho, v jaké úrovni se zrovna nacházíte. Pohyb nepřítele je rychlejší oproti pohybu hráče, aby hra byla obtížnější. Pohyb modrého kroužku je v každé úrovni a u každého kroužku daný a je pořád stejný. Každý nepřítel je vzhledově stejný, ale může mít rozdílný směr pohybu.

Nepřítel se tvořil podobně jako hráč, také přes objekt v JavaScriptu, ale na rozdíl od hráče jeho tvorba byla těžší a problematičtější. Objekt má stejně jako hráč pozici `x` a pozici `y`. Na rozdíl od hráče mu stačí jenom jedna proměnná na rozměry a tou je poloměr, díky jeho kulatému tvaru. Díky metodě `arc`, ve které se musí nastavit, jak dlouhou čáru chceme, může vzniknout kruh. Na rozdíl od hráče má nepřítel daný pohyb, který se musí pořád opakovat, to dělalo problematiku nepřítele těžší a problematičtější. Na vyřešení problematiky je potřeba vytvořit proměnnou, do které se uloží rychlost na ose `x` a proměnnou, kde se uloží na ose `y`. Jako další byla potřeba nějaká proměnná, do které se bude ukládat směr pohybu, v této práci ji zastupovala proměnná `direction`<sup>12</sup>. Objekt měl v programu samostatný js soubor, takže se

---

<sup>10</sup> Výška canvasu

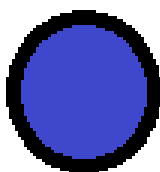
<sup>11</sup> Šířka canvasu

<sup>12</sup> Směr pohybu



musel importovat do hlavní stránky JavaScriptu, nepřítel se jako u hráče vykresloval na canvas.

Pozice a směr pohybu se u každého jednoho nepřítele může lišit, ale jenom z malé části. Kód se u každého jednoho objektu nemusel celý psát zase od začátku, ale stačilo ho vykopírovat a podle toho, kolik nepřátel v dané úrovni bylo potřeba, tak tolikrát se tam vložil, pak se přepsala pozice a směr pohybu.



Obrázek 5: Nepřítel

### 3.5 Bariéry

V této hře jsou dva druhy bariér. První druh bariéry tvoří ohraničení hracího pole, které způsobuje, že nic nemůže proniknout ven. Konkrétně, hráč nemůže pokračovat dál a jeho pohyb bariéra zastaví. Druhý typ bariéry tvoří celené pole společně s ohraničením hracího pole. Tahle bariéra způsobí, že nepřítel nesmí dovnitř zeleného pole a pryč z hracího pole, na rozdíl od prvního druhu bariéry nezastaví jeho pohyb, ale odrazí ho opět na druhou stranu. Tato bariéra způsobí stále stejný pohyb a do nekonečna opakovatelný pohyb nepřítele.

Problematika bariér se řešila přes rozdílné podmínky, které fungovaly na stejný způsob. U podmínek se muselo počítat s rozměry těles, a hlavně také s jejich ohraničením. Bez uvedení rozměrů do podmínek, by se tělesa odrážela a zastavovala o pár čísel na jiném místě a tím pádem by hra nefungovala pořádně a bezchybně a nebyla by úplně tak vizuálně krásná. Nesmí se zapomenout na tloušťku ohraničení těles, jelikož se musí započítat do rozměrů. Důležité byly proměnné  $x$  a  $y$ , díky kterým se dalo do podmínky nastavit například, až bude pozice  $x$  u hráče stejná, nebo větší než šířka canvasu, mínus rozměry hráče, tak se pozice  $x$  u hráče rovná šířce canvasu, tudíž hráč nemohl dál na pravé straně.



## 3.6 Pohyb objektů

Pohyb objektů se v této konkrétní práci rozděluje na dvě části. První druh pohybu se řadí ke čtverečku, který se ovládá pomocí znaků na klávesnici, konkrétně písmen WASD. Klávesa W slouží pro pohyb čtverce nahoru, klávesa S dolů, klávesa A doleva a klávesa D doprava. Může se to přirovnat k šipkám, ale je to naschvál zvoleno na tyto znaky, jelikož všechny klávesnice už šipky neobsahují, za to písmena většinou ano. A také je to více přizpůsobené hráčům. Většina her se ovládá klávesy WASD. Šipky jsou v tomhle ohledu zastaralé a nekomfortní. Druhý druh pohybu je pohyb nepřátel, který se pořád opakuje a nijak se neovládá, je předem nastavený. Pohyb nepřátel může začínat jak nahoru, dolů, tak doleva a doprava. Ve skutečnosti to je jedno, jelikož když by nepřítel, který má jet nahoru jel dolů, tak se hned automaticky odrazí od stěny a jede nahoru a má to úplně stejný výsledek, jako v opačném případě.

### 3.6.1 Pohyb hráče

Problematika pohybu byla v obou případech řešena přes podmínku. U hráče se vytvořily dvě proměnné,  $xr^{13}$  a  $xl^{14}$  pro pohyb po ose x a proměnné  $yu^{15}$  a  $yd^{16}$  pro pohyb po ose y. Například pro pohyb doprava musela znít podmínka následovně, když `e.code17 == klávese D`, tak  $xr$  = nějaké číslo, číslo byla rychlost, kterou chcete, aby se hráč pohyboval. Pro pohyb doleva musela být hodnota v záporu a samozřejmě se `e.code` musel rovnat klávese A, proměnná  $xl$  se pak musela rovnat číslu v záporu. To samé, akorát s rozdílnou klávesou a proměnnou bylo provedeno s pohybem nahoru a dolů, hodnota v záporu byla použita u pohybu nahoru. V tuhle chvíli by se hráč pohyboval pořád na tu stranu, kde byla zmáčknuta klávesa i po puštění klávesy. Proto se musela vytvořit druhá podmínka pro resetování pohybu. Ve zkratce, když je klávesa nahoře, tak se hráč přestane hýbat. Podmínka byla úplně stejná jako v předchozím případě, akorát se proměnná rovná nule.

---

<sup>13</sup> Pohyb po ose x doprava

<sup>14</sup> Pohyb po ose x doleva

<sup>15</sup> Pohyb po ose y nahoru

<sup>16</sup> Pohyb po ose y dolů

<sup>17</sup> Event v JavaScriptu





### 3.6.2 Pohyb nepřítele

U nepřítele byla podmínka pro pohyb jednodušší než u hráče. Na rozdíl u hráče byl pohyb promíchaný s bariérami, jelikož se musel objekt odrazit zase zpátky, po nárazu do bariéry. Tato problematika se řešila přes souřadnice bariér a souřadnice nepřítele.



## 4 Závěr

V závěru lze říci, že hra The World's Hardest Game je nesmírně náročná a vyžaduje velkou dávku trpělivosti a šikovnosti, které jsem ve vytváření jí, také použil. Její jednoduchá grafika dodává hře minimalistický styl, který však neubírá na zábavnosti a výzvě, kterou hra představuje. Hráči, kteří se rozhodnout si vyzkoušet tuto hru se mohou těšit na chvílky nekompromisního zábavného hraní a na vyzkoušení svých schopností a dovedností.

Při vytváření této práce jsem se setkal s velkým množstvím problematiky a s různými druhy. Asi největší problematikou, se kterou jsem se potkal při vytváření hry byla responzivita stránky hlavního menu. Problematika nebyla dlouho vyřešena a k jejímu vyřešení došlo vlastně až v samotném závěru práce. Kdy jsem došel k závěru, že se musí skoro celý kód v CSS překopat. Druhou největší problematikou, se kterou jsem se při vytváření této práce setkal byly kolize. Konkrétně kolize mezi hráčem a nepřítelem, které mi dlouhou dobu nefungovaly anebo fungovaly jenom na jedné straně.

V závěrečném shrnutí této práce bych určitě zmínil, že to je pro mě dobrá zkušenost a místy to byla i super zábava. Hra The World's Hardest Game splnila moje veškerá očekávání, ale lehce mě překvapila v náročnosti u dané problematiky, kdy jsem čekal že to bude místy větší hračka a sranda.



## 5 Přílohy

### 5.1 Seznam obrázků

Obrázek 1: Logo JavaScriptu [https://github.com/Shweta2024/JavaScript-Projects]	
.....	<b>Chyba! Záložka není definována.</b>
Obrázek 2: Logo HTML	
[https://en.wikipedia.org/wiki/HTML#/media/File:HTML5_logo_and_wordmark.svg] .....	8
Obrázek 3: Logo CSS	
[https://en.wikipedia.org/wiki/CSS#/media/File:CSS3_logo_and_wordmark.svg] .....	9
Obrázek 4: Hráč .....	11
Obrázek 5: Nepřítel .....	15

### 5.2 Zdroje

CSS. 9. Dostupné také z: <https://blog.shoptet.cz/co-je-nutne-vedet-o-css-1-dil>.

HTML. 7-8. Dostupné také z: <https://www.strafelda.cz/html>.