STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**

Pavel Šípek

Mladá Boleslav 2022

STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**

**Autor: Pavel Šípek  
Studijní obor: 18-20-M/01 Informační technologie  
Vedoucí práce: Jan Till**

Mladá Boleslav 2022

# Obsah

[Obsah 3](#_Toc102323513)

[1 Úvod 6](#_Toc102323514)

[2 Obsah práce 7](#_Toc102323515)

[2.1 HTML 7](#_Toc102323516)

[2.2 CSS 7](#_Toc102323517)

[2.3 JavaScript 8](#_Toc102323518)

[2.4 Vývojové prostředí 8](#_Toc102323519)

[2.5 Postup při tvoření kalendáře 9](#_Toc102323520)

[2.5.1 Pole 9](#_Toc102323521)

[2.5.2 Funkce 9](#_Toc102323522)

[2.5.3 Přepínání dnů a roků 11](#_Toc102323523)

[2.5.4 Zvýraznění uložené informace 11](#_Toc102323524)

[2.6 LocalStorage (web storage) 12](#_Toc102323525)

[2.7 Design/Vzhled stránky 13](#_Toc102323526)

[2.7.1 Obrázky 13](#_Toc102323527)

[3 Závěr 14](#_Toc102323528)

[4 Přílohy 15](#_Toc102323529)

[4.1 Seznam obrázků 15](#_Toc102323530)

[4.2 Zdroje 15](#_Toc102323531)

**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:

# Úvod

Smyslem této práce bylo vytvořit funkční kalendář, jako je třeba ve Windows. Kalendář by měl ukazovat základní funkce jako je aktuální den a možnost přepínat mezi roky a měsíci. Také by měl umožňovat ukládání informací k jednotlivým dnům.

Toto téma jsem si vybral, protože mi přišlo jako velmi zajímavé vytvořit něco co každý den používá mnoho lidí a je to taková nezbytná součást života. Kalendář používá prakticky každý a ani si neuvědomuje jeho důležitost. Tento projekt jsem vytvářel pomocí jazyků HTML, CSS a JavaScript. Ukládaná data jsem se rozhodl ukládat do LocalStorage. Tento kód by šel použít i v dalších projektech jako je třeba zásilkový portál nebo v kině.

# Obsah práce

## HTML

HTML[[1]](#footnote-1) HTML (Hypertext Markup Language) je jazyk používaný pro tvorbu webových stránek. Dříve se jazyk HTML používal i ke stylování, ale dnes se už místo HTML ke stylování používá CSS. V HTML se tvoří struktura webu, to znamená rozmístění jednotlivých prvků po stránce. HTML se rozděluje do dvou hlavních částí, a to do hlavičky a těla. V hlavičce jsou uvedeny informace popisující stránku. Jsou zde zapsány důležité informace jako je název stránky v kartě na internetovém prohlížeči. Do hlavičky se píší také propojení s CSS a odkazy na různé knihovny.

## CSS

CSS (Cascading Style Sheets) je kaskádový styl, který dodává styl webovým stránkám. V kaskádových stylech funguje dědičnost. Používá se v kombinaci s HTML. CSS odděluje styl od struktury a obsahu.

## JavaScript

JavaScript[[2]](#footnote-2) je objektově orientovaný programovací jazyk, který se využívá při tvorbě dynamických webových stránek. JavaScript je hlavně určen k vytvoření klientské části aplikace nebo stránky. Uživatel pošle požadavek na server a ten zpět zašle script společně s HTML a následně zpracované na prohlížeči. Díky tomu je aplikace vytvořená přes JavaScript rychlejší než aplikace vytvořené například přes PHP. Pomocí Node.js a jiným podobným technologiím může JavaScript fungovat i na serverech. Programy napsané v JavaScriptu se nazývají skripty a jsou propojeny s HTML kódem. Na rozdíl od Javy se není JavaScript tak těžký na pochopení, a i úplný začátečník se může naučit jeho základy. Do JS se také dají přidat různé knihovny jako je třeba React. JavaScript se také rozšířil i na mobilní telefony, a i díky technologii React Native.

## Vývojové prostředí

WebStorm[[3]](#footnote-3) je IDE (Integrated Development Enviroment), která se používá především pro vývoj JavaScriptu, ale umí i jiní jazyky. Pro uživatele správně používajícího WebStorm nebo jiné vývojové prostření může práci mnoho usnadnit. WebStorm také umí spolupracovat s mnoha různými aplikacemi a knihovnami např. (GitHub).



Obrázek : Logo WebStorm[https://www.wikidata.org/wiki/Q4053021]

## Postup při tvoření kalendáře

Kalendář se skládá ze tří základních tříd HTML, CSS, JavaScript a několika obrázků.

### Pole

Pole umožňuje mít uloženo v jedné proměnné více hodnot. Pole mohou být i dvourozměrné.

V tomto projektu je pole využito pro uložení názvů měsíců a pro uložení počtu dnů v měsících k dalšímu použití.

**const** month\_names = ['Leden', 'Únor', 'Březen', 'Duben', 'Květen', 'Červen', 'Červenec', 'Srpen', 'Září', 'Říjen', 'Listopad', 'Prosinec']

### Funkce

#### LeapYear

Funkce pro vypočítání přestupného roku.

#### FebDays

Tato funkce vybírá, zda bude mít únor 28 nebo 29 dní. Spolupracuje s funkcí LeapYear.

#### GenerateCalendar

Funkce generuje do kalendáře dny. K tomu využívá pole, ve kterém jsou uloženy počty dní v jednotlivých měsících a další pole, ve kterém jsou uloženy názvy měsíců. Funkce počítá i s tím, že každý měsíc nezačíná ve stejný den. Tak si umí dopočítat a doplnit zbývající dny prázdnými políčky.

#### OpenModal

Tato funkce otevírá okno pro možnost ukládání informací/poznámek k jednotlivým dnům.

**const** openModal = (date) => {

clicked = date;

**const** eventForDay = events.find(e => e.date === clicked);

**if** (eventForDay) {

document.getElementById('eventText').innerText = eventForDay.title;

deleteEventModal.style.display = 'block';

} **else** {

newEventModal.style.display = 'block';

}

backDrop.style.display = 'block';

}

#### closeModal

Funkce closeModal umožňuje zavřít okno pro ukládání.

#### SaveEvent

Tato funkce v tomto případě spustí požadavek pro ukládání vašich informací/poznámek.

#### DeleteEvent

Funkce maže věci, které jste si uložili.

#### initButtons

Funkce umožňuje fungovat tlačítkům (saveEvent, closeEvent, deleteEvent). A běží nepřetržitě.

#### Dark\_mode\_toggle

Tato funkce přepíná mezi světlým a tmavým režimem kalendáře. Funkce přepíná tlačítko, které je nastavené v CSS a světlé barvy a ikonu slunce a tmavé barvy s ikonou měsíce.

### Přepínání dnů a roků

Přepínání funguje pomocí querySelectorů. Selector prev-month a next-month přepíná měsíce. Tyto selektory jsou ošetřeny podmínkou, když uživatel dojede na první nebo poslední tak se načte automaticky následující měsíc. Dále jsou zde použity dva selektory a to prev-year a next-year, které přičítají nebo odečítají roky.

document.querySelector("#prev-month").onclick = () => {

--curr\_month.value

**if** (curr\_month.value < 0)

curr\_month.value = 11;

generateCalendar(curr\_month.value, curr\_year.value)

};

document.querySelector("#next-month").onclick = () => {

++curr\_month.value

**if** (curr\_month.value >= 12)

curr\_month.value = 0;

generateCalendar(curr\_month.value, curr\_year.value)

};

### Zvýraznění uložené informace

Pokud proměnná eventForDayCheck najde novou událost potom se v nedefinované podmínce přidá nová událost a označí se stylem, který je nastavený v CSS.

## LocalStorage (web storage)

Local Storage[[4]](#footnote-4) je HTML API umožňující webovým aplikacím ukládat data do webového prohlížeče na straně uživatele. Do tohoto uložiště jde data ukládat dočasně nebo trvale. Na webových prohlížečích jsou dva typy míst, kam můžeme ukládat data (local storage, session storage) tyto dva typy se liší životností a rozsahem. Zprvu by mohlo vypadat, že web storage a cookies jsou úplně stejné věci, ale není to tak web storage má mnoho výhod oproti cookies. Web storage je podporováno většinou moderních webových prohlížečů.

Rozdíly mezi web storage a cookies. Cookies slouží také k ukládání dat, která jsou přenášena protokolem HTTP. Uživatel si na mnoha webových stránkách může nastavit, jaké cookies se mu budou ukládat a poskytovat webovým stránkám a aplikacím jeho preference anebo-naopak neukazovat věci, o které nemá zájem. Také je velký rozdíl v kapacitě uložiště web storage má k dispozici 5-10 MB za to cookies má k dispozici pouze 4 KB. Důvod, proč mají cookies tak malou kapacitu je, že se přenášejí každým HTTP požadavkem. Web storage má data uchována v klientově prohlížeči, odkud s nimi dále může operovat JavaScript.

Jak je použit local Storage v tomto projektu.

V tomto projektu local storage slouží jako místo pro ukládání informací/poznámek, které si uživatel napíše k jednotlivým dnům a následně uloží pomocí tlačítka k tomu určenému. Uživatel také může uložená data smazat.

**let** events = localStorage.**getItem**('events') ? JSON.**parse**(localStorage.**getItem**('events')) : [];

## Design/Vzhled stránky

Tento projekt je designován v moderním stylu. Jsou zde použity jednoduché tvary. Kalendář je přizpůsoben co nejjednoduššímu použití. Na stránce je také možnost přepínat mezi světlým a tmavým režimem. Je zde použit font písma Plus Jakarta Sans z knihovny fontů od Google Fonts[[5]](#footnote-5). Skoro každá komponenta v HTML co je zde použita má svoji vlastní třídu nebo id, přes které je nastylována v CSS.

### Obrázky

Obrázky použité v tomto projektu jsou unikátní. Jsou vytvořené pomocí programu Paint.NET.

Paint.NET[[6]](#footnote-6) je software vytvořený pro úpravu fotografií a obrázků. Program je dostupný pro počítače s operačním systémem Windows. V programu je k dispozici mnoho funkcí, speciálních efektů a nástrojů pro širokou paletu uživatelů. Tento program je plně zdarma a je srovnatelný s jinými programy jako jsou GIMP a Adobe Photoshop.

# Závěr

Smyslem této práce bylo vytvořit funkční kalendář, do kterého si uživatel bude moci ukládat data. Pro vytvoření této stránky bylo zapotřebí vymyslet, jak to bude celé fungovat, protože bylo mnoho možností. Ze tří možností pro ukládání byla vybrána ta nejsympatičtější, a to local storage. Dále bylo nutné se naučit, jak data dostat na uložiště a zase je smazat. Také bylo zapotřebí vymyslet, jestli se políčka jednotlivých dnů budou generovat nebo budou natvrdo zapsána každé zvlášť. Nakonec byla zvolena možnost generování políček. A design byl od začátku jasný, udělat moderně vypadající a fungující kalendář. A to myslím, že se povedlo.

# Přílohy

## Seznam obrázků

[Obrázek 1: Logo WebStorm[https://www.wikidata.org/wiki/Q4053021] 8](#_Toc102319823)

## Zdroje

HTML. *Netinbag* [online]. [cit. 2022-04-30]. Dostupné z: https://www.netinbag.com/cs/internet/what-is-html.html

JavaScript. *Rascasone* [online]. [cit. 2022-04-30]. Dostupné z: https://www.rascasone.com/cs/blog/co-je-javascript-pro-zacatecniky

WebStorm. *JetBrains* [online]. [cit. 2022-04-30]. Dostupné z: https://www.jetbrains.com/webstorm/

Web storage. *Wikipedie* [online]. [cit. 2022-04-30]. Dostupné z: https://cs.wikipedia.org/wiki/Web\_storage

Google fonts. *Google fonts* [online]. [cit. 2022-04-30]. Dostupné z: https://fonts.google.com/specimen/Plus+Jakarta+Sans?query=Plus

Paint.NET. *Paint.NET* [online]. [cit. 2022-04-30]. Dostupné z: https://www.getpaint.net/

1. HTML. *Netinbag* [online]. [cit. 2022-04-30]. Dostupné z: https://www.netinbag.com/cs/internet/what-is-html.html [↑](#footnote-ref-1)
2. JavaScript. *Rascasone* [online]. [cit. 2022-04-30]. Dostupné z: https://www.rascasone.com/cs/blog/co-je-javascript-pro-zacatecniky [↑](#footnote-ref-2)
3. WebStorm. *JetBrains* [online]. [cit. 2022-04-30]. Dostupné z: https://www.jetbrains.com/webstorm/ [↑](#footnote-ref-3)
4. Web storage. *Wikipedie* [online]. [cit. 2022-04-30]. Dostupné z: https://cs.wikipedia.org/wiki/Web\_storage [↑](#footnote-ref-4)
5. Google fonts. *Google fonts* [online]. [cit. 2022-04-30]. Dostupné z: https://fonts.google.com/specimen/Plus+Jakarta+Sans?query=Plus [↑](#footnote-ref-5)
6. Paint.NET. *Paint.NET* [online]. [cit. 2022-04-30]. Dostupné z: https://www.getpaint.net/ [↑](#footnote-ref-6)