STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**

Vojtěch Věchet

Mladá Boleslav 2024

STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**

**Autor: Vojtěch Věchet  
Studijní obor: 18-20-M/01 Informační technologie  
Vedoucí práce: Jan Urban**

Mladá Boleslav 2024

# Obsah

[Obsah 3](#_Toc174220954)

[1 Úvod 7](#_Toc174220955)

[2 Obsah práce 8](#_Toc174220956)

[2.1 Visual Studio Code 8](#_Toc174220957)

[2.2 HTML 8](#_Toc174220958)

[2.3 Programovací jazyky 8](#_Toc174220959)

[2.3.1 CSS 8](#_Toc174220960)

[2.3.2 JavaScript 9](#_Toc174220961)

[2.4 Electron 9](#_Toc174220962)

[2.5 Node.js 9](#_Toc174220963)

[2.6 Local Storage 9](#_Toc174220964)

[2.7 Problematika 10](#_Toc174220965)

[2.7.1 Určení přestupného roku 10](#_Toc174220966)

[2.7.2 Připravení pro zobrazeni kalendáře 11](#_Toc174220967)

[2.7.3 Vypsaní dní do kalendáře 11](#_Toc174220968)

[2.7.4 Vypsaní kalendáře 12](#_Toc174220969)

[2.7.5 Zvýrazněni aktuálního dne 12](#_Toc174220970)

[2.7.6 Posouvaní mezi měsíci 13](#_Toc174220971)

[2.7.7 Výběr dne 13](#_Toc174220972)

[2.7.8 Přidáni nove události 13](#_Toc174220973)

[2.7.9 Aktualizace kalendáře 14](#_Toc174220974)

[2.7.10 Zobrazení událostí 14](#_Toc174220975)

[2.7.11 Označení dnů s událostmi 14](#_Toc174220976)

[2.7.12 Vzhled 15](#_Toc174220977)

[3 Závěr 17](#_Toc174220978)

[4 Přílohy 18](#_Toc174220979)

[4.1 Seznam obrázku 18](#_Toc174220980)

[4.2 Seznam zdrojových kodů 18](#_Toc174220981)

[4.3 Zdroje 18](#_Toc174220982)

**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

Cílem této práce je vyvinout chytrý kalendář který se může chlubit mnoho funkcemi, včetně zobrazeni aktuálního dne, posouvaní mezi měsíci a roky pomoci jednoduchého posouvaní a také funkci pro ukládaní zaznamenaných událostí.

Toto téma jsem si vybral, protože kalendář je jedním z nástrojů pro správu casu, který umožňuje uživatelům plánovat a sledovat své události, aktivity a schůzky. S příchodem nových technologii je, čím dal víc populární si zapisovat události do mobilu nebo počítače.

Project jsem vytvářel v Electronu za pomocí jazyků HTML, CSS a JavaScript. Uživatel klikne na datum v kalendáři a poté na tlačítko vyvořit událost a událost se zapíše do Local Storage. Následně se zobrazí událost a v kalendáři se označí barevně datum.

# Obsah práce

## Visual Studio Code

Při vytváření kalendáře byl použit editor Visual Studio Code, známý také jako VSC. Tento editor je jedním z nejoblíbenějších a nejznámějších mezi programátory. Oblíbený je díky množství rozšíření pro práci s kódem a nápovědě při psaní kódu.

## HTML

Hypertext Markup Language známý jako HTML je jazyk pro tvorbu webových stránek. HTML se používá pro rozložení jednotlivých prvku na stránce, skládá se ze dvou hlavních částí tělo a hlavička. V tělu se zapisuje jednotlivé značky (tag) které se používají pro jednotlivé prvky, jako odstavec, nadpis, obrázek, odkazy apod. V hlavičce se nachází informace stránky jako je název stránky. také se nachází propojeni s CSS.

## Programovací jazyky

V Kalendáři se objevuji dva základní programovací jazyky CSS, JavaScript

### CSS

Cascading Style Sheets je známy jako CSS který se používá společně s HTML, kterému dává vzhled stránky.

### JavaScript

JavaScript (JS) [[1]](#footnote-1)je skriptovací jazyk učený pro tvorbu moderních dynamických webů. Byl představen v devadesátých letech 20. století, jako reakce na klasické statické internetové stránky a otevřel prostor pro interaktivní webové aplikace i příjemnější uživatelské rozhraní doplněné animacemi a 2D i 3D grafikou. JavaScript je prioritně určen ke zhotovování klientských částí aplikací, s příchodem Node.js a podobných technologií však dokáže plnohodnotně figurovat i na straně serveru.

## Electron

Za pomoci Elektronu se vytvářejí desktopové aplikace za pomocí JavaScriptu, HTML a CSS. Díky Cromia a Node.js vám Elektron umožňuje vytvářet aplikace pro více platforem, které fungují na Windows, macOS a Linux.

## Node.js

Node.js umožňuje spuštění JavaScript mimo webové prohlížeče.

## Local Storage

Local Storage je funkcionalita webovými prohlížecí v rozhraní Javascriptu. Jedná se o uložiště dat, které je k dispozici v rámci webového prohlížeče. Data jsou uložena pod klíčem a je přirazena hodnota. V tomto případě Klíč pro local storage je datum dne, pro kterou je událost naplánovaná. Hodnota je pak rozdělena čárkami na tři části den čas konaní události, barevne označení a název.

## Problematika

Během práce na ročníkovém projektu se objevilo několik problémů, které bylo nutné vyřešit, nebo potřeba projekt částečné upravit či přepracovat

### Určení přestupného roku

Pro správnou funkci kalendáře je nezbytné, aby funkce nejprve ověřila, zda aktuální rok je rokem přestupným. Obecné pravidlo říká, že přestupný rok je každý rok, který je dělitelný čtyřmi, což znamená, že pokud je rok při dělení čtyřmi beze zbytku, má únor 29 dní. Avšak toto pravidlo má výjimky, které jsou nezbytné pro udržení přesnosti kalendáře v dlouhodobém horizontu.

Pokud je rok dělitelný 100, tedy pokud se při dělení 100 nevyskytne žádný zbytek, rok by neměl být přestupný. Tento krok by mohl způsobit, že by se únor v daném roce vrátil zpět k 28 dnům, což by na první pohled mohlo kalendář narušit. Ale pravidlo má ještě jednu výjimku, která tento problém řeší. Pokud je totiž rok dělitelný nejen 100, ale také 400, rok se opět považuje za přestupný. Jakmile funkce určí, že se jedná o přestupný rok, je dalším krokem úprava pole, které obsahuje počet dní v jednotlivých měsících.

Tím se zajistí, že únor bude mít 29 dní, což je zásadní pro správné zobrazení kalendáře. Pokud funkce zjistí, že rok není přestupný, únor zůstává s 28 dny.



Zdrojový kód č. 1 – pole pro počet dni v měsíci

Obsah obrázku text, Multimediální software, software, snímek obrazovky

Popis byl vytvořen automaticky

Zdrojový kód č. 2 - funkce pro výpočet přestupného

### Připravení pro zobrazeni kalendáře

Po zjištěni, zda se jedna o přestupu rok či nikoliv je důležitá tabulka do které se budou dny v měsíci vypisovat se musí nejprve připravit. To znamená vymazání všech existujících hodnot a nastavit barvu textu na bílou, jinak by zůstala barva z poznámky z minulého měsíce. Vymazání hodnot zahrnuje odstranění jakéhokoliv textu z jednotlivých buněk v tabulce, aby se zabránilo zobrazení starých dat z minulého měsíce. Jinak by se vypsal kalendář a zůstali by před nim nebo za nim hodnoty z minulého měsíce.

### Vypsaní dní do kalendáře

Kalendář se vypisuje do předem připravené tabulky v HTML, kde je každé políčko označeno datasetem (označením). Pro výpis hodnot do políček slouží funkce, která nejprve zjistí, zda je daný rok přestupný. Pokud ano, v poli, kde je zapsán počet dní v měsíci, se změní počet dní v únoru z 28 na 29. Následně funkce vymaže všechny hodnoty z políček a změní barvu textu na bílou. Poté začíná samotný výpis hodnot do políček, kde pomocí cyklu a prvního dne v měsíci vypíše příslušné hodnoty.



Zdrojový kód č.3 – funkce pro vymazání hodnot a změna barvy textu

### Vypsaní kalendáře

Jakmile je tabulka připravena pro zobrazení, funkce začne vypisovat dny aktuálního měsíce do jednotlivých políček tabulky. Proces začíná tím, že funkce pomocí metody getDay() zjistí, na který den v týdnu připadá první den v měsíci. Získáme číslo od 0 do 6, které odpovídá dni v týdnu (0 – pondělí, 1 - úterý atd.). Na základě této informace funkce správně umístí první den měsíce do odpovídající buňky v tabulce. Pomocí cyklu se pak pro každý den v měsíci vypisuje číslo dne do správné buňky.

Kromě čísel dnů funkce také zkontroluje, zda existují události naplánované na jednotlivé dny. Tyto události jsou uloženy v Local Storage. Funkce načte události a přiřadí je k odpovídajícím dnům v kalendáři. Každá událost má své barevné označení. Den, který má událost, je pak označen barevným označením (prvním uloženou událostí v daném dni). Díky tomu mohou uživatelé snadno vidět, na který den mají naplánovanou událost a které dny mají volno.

Pokud na daný den žádná událost není, do odstavce v HTML se vypíše text: „Dneska nemáte žádnou událost."

### Zvýrazněni aktuálního dne

Dalším prvkem je zvýrazněni aktuálního dne. Jakmile jsou všechny dny měsíce vypsány, funkce provede kontrolu, zda některé toliko odpovídá aktuálnímu dni. Pokud ano, toto políčko je zvýrazněné červenou barvou. Díky teto pomoci si uživatel snadno zjisti dnešní datum a rychleji se bude orientovat v kalendáři

### Posouvaní mezi měsíci

Kalendář zahrnuje pohybové prvky, které umožňují uživatelům přecházet mezi jednotlivými měsíci. Tlačítka pro přechod na předchozí a následující měsíc umožňují snadný pohyb mezi měsíci. Pokud se uživatel chce přesunout na předchozí měsíc, musí kliknout na šipku, která míří doleva. Funkce zkontroluje, zda je aktuální měsíc leden. Pokud ano, rok se sníží o jeden a měsíc se nastaví na prosinec. Pokud není leden, sníží se pouze měsíc o jeden.

Naopak, když uživatel klikne na tlačítko pro následující měsíc (šipka doprava), funkce zvýší měsíc o jeden a pokud je aktuální měsíc prosinec, zvýší se rok o jeden a měsíc se nastaví na leden. Po každé změně se kalendář vymaže a nahrají se nové hodnoty do kalendáře.

### Výběr dne

Uživatel má možnost kliknout na libovolné políčko v tabulce, které odpovídá konkrétnímu dni. Funkce, která reaguje na tento klik, vezme hodnotu z data setu tohoto políčka, což je specifická informace o dni, který uživatel vybral. Funkce pak zkontroluje, zda je v políčku již nějaká hodnota. Pokud ano, uloží se datum do Local Storage a funkce se zaměří na aktualizaci zobrazených informací. Včetně tohoto kroku se aktivují další funkce, které umožní zobrazení událostí plánovaných pro vybraný den.

### Přidáni nove události

Pro přidání nové události je k dispozici tlačítko "vytvoření události". Po jeho stisknutí se zobrazí formulář, který umožňuje uživatelům zadat detaily nové události. Tento formulář může zahrnovat pole pro název události, čas konání a barevné označení. Jakmile uživatel vyplní všechny potřebné údaje a klikne na tlačítko pro vytvoření události, data se uloží do Local Storage. Pokud uživatel nezadá název události, funkce zobrazí upozornění, které informuje uživatele, že název je povinný. Tento krok zajišťuje, že všechny události mají potřebné informace, které jsou nezbytné pro jejich správné zobrazení.

### Aktualizace kalendáře

Po uložení nové události je třeba aktualizovat kalendář. Funkce skryje formulář pro přidání události a zobrazí zbytek stránky. Následně se kalendář znovu vykreslí s novými daty. Funkce načte události z Local Storage a aktualizuje jednotlivé buňky v kalendáři podle nových údajů. Pokud již existují události pro daný den, nové události se přidají k těm, které už byly uloženy. Tento proces zajišťuje, že kalendář vždy zobrazuje aktuální a správné informace.

### Zobrazení událostí

Pro zobrazení událostí naplánovaných na konkrétní den funkce načítá data z Local Storage a kontroluje, zda existují záznamy pro daný den. Pokud jsou události nalezeny, funkce je rozdělí na jednotlivé části a zobrazí je v odstavci na stránce. Tato část stránky může obsahovat seznam událostí spolu s jejich detaily, jako je název, čas a barva. Pokud pro daný den nejsou žádné události, zobrazí se zpráva "dnes nemáte žádnou událost", aby uživatel věděl, že pro tento den nejsou naplánovány žádné akce.

### Označení dnů s událostmi

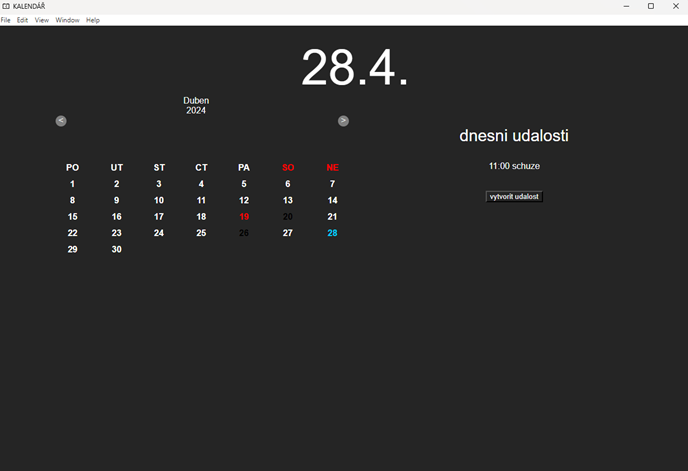
Pro zlepšení přehlednosti funkce prochází všechna políčka v kalendáři a porovnává je s dny, pro které jsou naplánovány události. Pokud se políčko shoduje s dnem, kdy je naplánována událost, toto políčko se označí speciální barvou. To usnadňuje uživatelům rychlou identifikaci dnů s událostmi, což zvyšuje uživatelský komfort a přehlednost kalendáře. Označení dnů může zahrnovat změnu barvy pozadí nebo přidání malého označení, které vizuálně vyjadřuje přítomnost události.

### Vzhled

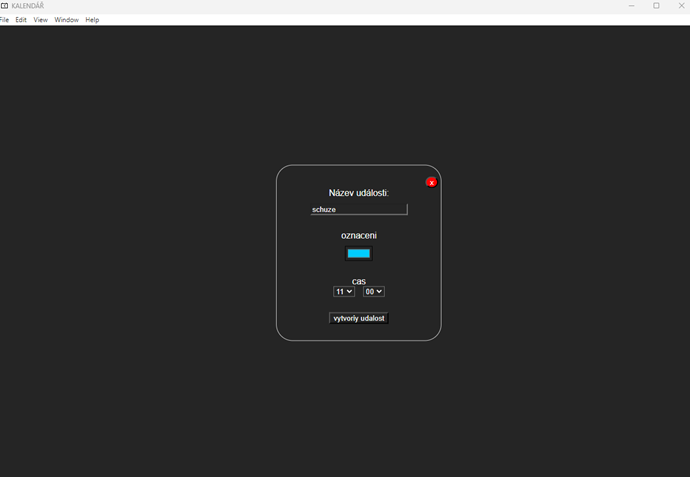
Vzhled aplikace je navržen tak, aby byl co nejjednodušší a snadno pochopitelný pro uživatele všech věkových kategorií, od malých dětí až po seniory. Uprostřed horní části obrazovky se nachází datum aktuálně označeného dne, které je jasně viditelné. Na levé straně obrazovky se nacházejí tlačítka pro posunování kalendáře do předchozího nebo následujícího měsíce. Mezi těmito tlačítky se nachází zobrazení aktuálního měsíce a roku, ve kterém se kalendář nachází, což umožňuje uživateli snadnou orientaci v čase.

Pod touto oblastí se nachází samotný kalendář, kde jsou jednotlivé dny jasně označeny. Dny, které obsahují naplánované události, jsou barevně zvýrazněny, což umožňuje uživateli rychle a snadno identifikovat, ve kterých dnech jsou naplánované aktivity. Každý den je rozdělen do jednotlivých políček, která reprezentují dny v měsíci.

Na pravé straně obrazovky, vedle kalendáře, se nachází seznam naplánovaných událostí. V tomto seznamu se zobrazuje čas a název každé události, což umožňuje uživateli rychlý přehled o jeho denním plánu. Pod tímto seznamem je umístěno tlačítko pro vytvoření nové události. Po kliknutí na toto tlačítko se zobrazí formulář, který umožňuje uživateli zadat název události, vybrat její čas a přiřadit jí specifické označení. Tento formulář je navržen tak, aby byl intuitivní a snadno použitelný, a to i pro uživatele, kteří nemají mnoho zkušeností s digitálními technologiemi. Tímto způsobem je zajištěno, že každý uživatel, bez ohledu na svůj věk nebo technické dovednosti, bude schopen efektivně používat kalendářovou aplikaci pro plánování svých každodenních aktivit.



Obrázek č.1 – hlavní okno kalendáře [Vlastní zdroj]



Obrázek č.2 – okno pro vytvoření události [Vlastní zdroj]

# Závěr

Smyslem této práce bylo vytvořit Kalendář, do kterého se můžou zadávat různé události a schůzky.

Nejtišší bylo vymyslet samotný zobrazeni kalendáře. Kde po spoustu neúspěšných pokusech se nakonec podařilo vyřešit problém a kalendáře začal ukazovat správné měsíce. Dále přišlo posouvaní mezi myšicemi což bylo asi nejlehčí na sprovozeni v této práci. Následně bylo obtížné vymyslet samotné ukládaní kde bylo více možností ale nakonec bylo zvoleno uložení do Local Storage. Poté přišla obtížnější část, kde se muselo vymyslet, jak se data z Local Storage budou zobrazovat v kalendáři. Následně vypsání událostí s názvem a časem bylo lehké.

Chtěl by tímto poděkovat svému vedoucímu ročníkové práce, Janu Urbanovi, za ochotu a podporu.

# Přílohy

## Seznam obrázku

Obráze č.1: screenshot Kalendáře 16

Obráze č.2: screenshot Kalendáře 16

## Seznam zdrojových kodů

Zdrojový kód č.1: pole pro počet dní v měsíci 16

Zdrojový kód č.2: funkce pro výpočet přestupného 16

Zdrojový kód č.3: funkce pro vymazání hodnot a změna barvy textu 16

## Zdroje

Co je JavaScript? Online. Rascasone. Dostupné z: https://www.rascasone.com/cs/blog/co-je-javascript-pro-zacatecniky. [cit. 2024-08-10].

*JavaScript Tutorial*. Online. Rascasone. Dostupné z: <https://www.w3schools.com/js/default.asp>. [cit. 2024-08-10].

*What is Electron?* Online. Electron. Dostupné z: <https://www.electronjs.org/docs/latest/>. [cit. 2024-08-10].

1. Co je JavaScript? Online. Rascasone. Dostupné z: https://www.rascasone.com/cs/blog/co-je-javascript-pro-zacatecniky. [cit. 2024-08-10]. [↑](#footnote-ref-1)