|  |
| --- |
| **Fakulta Informatiky a Managementu** |
| **Univerzita Hradec Králové** |



**Moderní metody stylování webových aplikací**

Forma studia: Prezenční 2023/2024

Obor: IM3

Ročník: 3. ročník

Vedoucí práce: Ing. Karel Malý, Ph.D.

Datum zpracování: 25. června 2024

Autor(ka): Nazarij Dovžanyn

Anotace:

DOVŽANYN, Nazarij. Moderní metody stylování webových aplikací. Hradec Králové: Fakulta informatiky a managementu Univerzity Hradec Králové, 2024. 22s. Seminární práce.

Seminární práce se zabývá porovnáním zvolených metodik stylování webových aplikací. Teoretická část bakalářské práce jejíž obsah a vzhled tato seminární práce nastiňuje se zaobírá historií celkového designu webu a webových aplikací a také popisuje její vývoj a trendy, které se v průběhu let od vzniku střídali. Obsahuje také vybrané metody stylování, většinou ty nejvíce známé z nejvyužívanějších okruhů stylování webu, a popisuje je.

Klíčová slova:

Web, aplikace, vzhled, rozložení, vývoj, trend

OBSAH

[1 Úvod 4](#_Toc175575701)

[2 Cíl seminární práce 4](#_Toc175575702)

[3 Web design 6](#_Toc175575703)

[3.1 Historie web designu 6](#_Toc175575704)

[3.1.1 Počátky web designu 6](#_Toc175575705)

[3.1.2 Evoluce vývoje webu 7](#_Toc175575706)

[3.1.3 Mobilní éra 8](#_Toc175575707)

[3.1.4 Budoucnost web designu 8](#_Toc175575708)

[3.2 Trendy web designu 8](#_Toc175575709)

[3.3 UI a UX 10](#_Toc175575710)

[3.3.1 UX 10](#_Toc175575711)

[3.3.2 UI 11](#_Toc175575712)

[4 Metodiky stylování 12](#_Toc175575713)

[4.1 CSS 12](#_Toc175575714)

[4.1.1 Verze CSS 13](#_Toc175575715)

[4.1.2 Struktura CSS 13](#_Toc175575716)

[4.2 Preprocesory 14](#_Toc175575717)

[4.2.1 SASS 14](#_Toc175575718)

[4.2.2 Less 15](#_Toc175575719)

[4.3 CSS-in-JS 15](#_Toc175575720)

[4.3.1 Emotion 16](#_Toc175575721)

[4.3.2 Styled Components 16](#_Toc175575722)

[4.4 Komponentní frameworky 16](#_Toc175575723)

[4.4.1 MUI 17](#_Toc175575724)

[4.4.2 Bootstrap 17](#_Toc175575725)

[4.4.3 Tailwind CSS 18](#_Toc175575726)

[5 ZÁVĚR 19](#_Toc175575727)

[6 SEZNAM ZDROJŮ 20](#_Toc175575728)

# Úvod

Stylování webových aplikací se za dvě desítky let změnili v mnoha směrech. Tato práce se rozděluje na teoretickou a praktickou část, v teoretické části bakalářská práce sleduje historii stylování webu a webových aplikací, která se za více jak dvě dekády rapidně měnili, popisuje počátky stylování a předky a také uživatelské požadavky, které měnili směr vývoje, a mimo jiné také sděluje trendy v průběhu let. Je zde zmíněn pojem uživatelské rozhraní. Teoretická část se zabývá také vývojem a popisem vybraných kategorií stylování pro vzhled základní aplikace jako je základního CSS, CSS preprocesory, CSS in JS a frameworky.

V praktické části se bakalářská práce zaměřuje na informace, zdrojový kód, využití, dokumentaci a náročnost využitých metod. Činí tak pomocí rozpracované základní webové aplikace, která se dále zvlášť upravuje do ucelené podoby pomocí jednotlivých metod stylování. Mezi implementacemi, které se k zobrazení a úpravě vzhledu zvolili, se bude porovnávat, jak byli jednotlivé metody náročné od těch dalších, nebo v čem vynikali a v čem jsou naopak slabší. V aplikaci se navrhli základní okruhy stavby webu, na kterých se tato odlišení a porovnání dají jednoduše určit. Téma je spojeno s pracovním prostředím a metody stylování jsou využívány na denní bázi.

# **Cíl seminární práce**

Smyslem bakalářské práce je odlišit od sebe zvolené metody stylování. Důraz se klade na určení, která z těchto metod je lepší v ohledech, ve kterých ostatní metody zaostávají, je jasné že jedna metoda nemůže být nejlepší než ostatní ve všech ohledech, a proto se porovnávají okruhy jako je práce s textem, s obrazem, funkcemi, proměnnými, celkovým vzhledem a responzivitou, tabulkami a formuláři nebo také výkonem. Každá kategorie stylování je specificky lepší v některých okruzích, a to se bude snažit práce pospat a jasně sdělit. K porovnání dojde také v čitelnosti napsaného kódu stylování nebo jak snadné bylo dohledat daný okruh stylování v dané metodě stylování, tedy jak přehlednou a rozsáhlou má metoda stylování dokumentaci a podporu.

K tomuto porovnání dojde z vypracování webové aplikace. Tato aplikace bude představena zprvu jako kostra pouze s žádaným obsahem bez sebemenšího stylování. A poté každou jednotlivou vybranou metodou stylování se tato kostra s obsahem upraví pro žádoucí vizuální vzhled. Tento vzhled, kterého se bude chtít dosáhnout, bude již stanoven a za využití každé jednotlivé metody se k této vizualizaci musí aplikace co nejvíce přiblížit. Jelikož není cílem, aby se využilo jednotlivých metod stylování pouze pro bezvýznamnou stavbu vzhledu za využití jejich nejsilnějších stránek, ale naopak stanovením vzhledů a okruhů využití, které musí aplikace obsahovat se využijí v daných metodách, jak vlastnosti, ve kterých vynikají tak se najdou vlastnosti, ve kterých zaostávají. V rámci stylizace webové aplikace k žádanému vzhledu dojde k vytvoření několika těchto aplikací, kde ve zdrojovém kódu bude využito pouze vybrané metodiky stylování.

# Web design

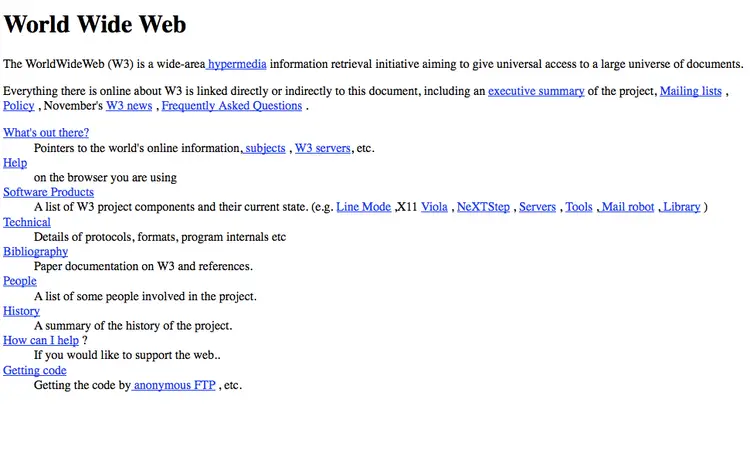
Stručný popis historie web designu a jak se v průběhu času vyvíjel. Zaměřuje se zde na přelomové okamžiky a stavební pilíře, které formovali design, na trendy v průběhu let a důležité pojmy uživatelského rozhraní a zážitku.

## Historie web designu

Historicky se webové stránky používali primárně jako nástroj k zobrazení informací. S nedostatečnými nástroji, stránky vypadali vizuálně nepřívětivé narozdíl od moderních webových stránek a aplikací.

### Počátky web designu

Prvním průkopníkem web designu byl Tim Berners-Lee. Je nazýván prvním web designerem a mimo jiné v roce 1990 vytvořil HTML tedy Hypertext Markup Language, což je značkovací jazyk pro tvorbu webu. Jeho stránka se dostala do sítě 6. srpna 1991, obsahovala pouze informace o sobě samotný a byla dostupná na výzkumném centru CERN ve Švýcarsku. První stránka a její vzhled se nelišil od obyčejného zobrazení textu na papíře, pouze černý text na bílém pozadí, avšak jediné, co měla stránka navíc byl modrý text, který vyznačoval odkaz.



Obr. 1 World wide web. Z knihy Digital Design: A history, ESKILSON.

‚*Web, jako zdroj pro sdílení vědeckých informací, byl pouze vizuálním zprostředkovatelem v tom nejpřesnějším slova smyslu*.‘ jak uvádí ESKILSON (2023:96). V této době, rok 1991 až 1994, se informace organizovali a stylovali v podobě tabulek, neobsahovali žádné barvy, obrázky ani grafy, pouze text. S přibývajícími webovými stránkami se začalo více zaměřovat na design jako třeba barevný text, barevné pozadí nebo i bannery. Vedla k tomu hlavně potřeba upoutat potencionální návštěvníky a uživatele.

V těchto prvopočátcích se začali objevovat první náznaky stylování jako například ViolaWWW od Pei-Yaun Wei v roce 1991, nebo využití nestandartních tagů v HTML pro design od Adreessena ale ani jeden z těchto příkladů nebyl natolik přelomový jako nadcházející Cascading Style Sheets, zkráceně CSS, od Håkon Wium Lie.

### Evoluce vývoje webu

V období 1995 až 2000 se vzhled webových stránek začal vyvíjet a zlepšoval tak uživatelům čas strávený na webu. V roce 1995 se vydala nová verze HTML 2.0 podporující grafy, formuláře, tabulky a mnohem více. Tento krok umožnil vývojářům větší svobodu při organizování stránek. V tomto období také na web dorazil JavaScript, který dodal interaktivnost a pohyb do statického vzhledu webu.

Dalším zásadním krokem bylo CSS, vydáno v roce 1996. Tento jazyk upřesnil stylování jednotlivých elementů v HTML jako barvu, rozložení, typografii a další. V této době také přišel do světa stylování Macromedia Flash, jenž přidal na web animace a další dynamický obsah. Animace však vyžadovali, aby web měl Flash rozšíření, což spolu s budoucím vzestupem mobilních zařízení, které Flash nepodporovali, vedlo k zániku a ukončení podpory pro Flash.

V dalším období, 2000 až 2007, se začínal rozvíjet responzivní vzhled webů. Vývojáři začali vytvářet více verzí webu pro různé prohlížeče a také zařízení, což s rychlým vývojem prohlížečů a zařízení, vyžadovalo spoustu času, jelikož se verze vyvíjeli zvlášť.

Více a více lidí vytvářeli weby pro podnikání, informativní funkci a blogování. S tímto se do popředí vznesli systémy pro správu obsahu neboli Content management system, zkráceně CMS. Tyto systémy přinesli hladší aktualizaci obsahu webů. Jedním z největších CMS byl WordPress, avšak v roce 2003 přišel MySpace a rok poté Facebook. Tyto dvě aplikace s sebou přinesli nový způsob web interakce, ve které se vyskytovali reklamy, webové widgety a nové interakce fotek.

### Mobilní éra

Mezi lety 2007 a 2010, se s rostoucí technologií, mobilní zařízení dostali do obliby uživatelů. V této době již byla přátelská responzivita webu na mobilní zařizení nezbytnou. V roce 2010 Ethan Marcotte publikoval Responzivní Web Design, který nastínil optimalizaci obsahu na základě displeje.

V průběhu času se design změnil z bohatého zpracování na minimalistický a jednoduchý. Toto zmenšilo čas načítání a stalo se přívětivějším a účinnějším vůči uživatelům. V přítomnosti je to vše zaměřeno na uživatele a na jeho pocit z návštěvy webu.

### Budoucnost web designu

Do webu ve velkým přichází AI, její využívání v designu a interakci s uživatelem je ovšem potřeba brát ohleduplně. AI může pomáhat s vytvořením responzivity a různými částmi kódu, tedy k nasměrování vývojáře k jedné cestě, jak vyřešit problém anebo nabídnutí mnoha řešení, jak daný problém vyřešit. Vyvíjení webu se tím může výrazně zrychlit. Ať už je budoucnost jakákoliv vždy se musí zaměřovat na uživatele, jeho pocit z webu a jeho bezpečnost, jelikož problém s manipulací citlivých dat je na webu již dlouhá léta.

## Trendy web designu

Trendy se v průběhu času drasticky měnili s příchodem nových technologií a nových jazyků. V počátcích webu bylo hlavní zaměření na funkčnost, avšak postupem času se více upřednostňovala využitelnost, responzivita a jednoduchost.

V devadesátých letech v době prvních webových stránek se na webu zobrazoval pouze čistý text i s několika obrázky. Převládalo tabulkové rozložení stránek, jinak známo jako grid-based layouts, do řádků a sloupečků s omezeným využitím barevného pozadí nebo textu, odlišného písma nebo velikostí písma. Popularitu získali GIFy, které stránce přidávali pohyb a interakci. Zaměřovalo se především na funkčnost, aby se informace dostali k uživateli. Stránky se načítali rychle a design byl o čitelnosti a přístupu, ne o estetice.

Na konci tisíciletí vzrostl zájem o webové stránky na komerční využití. V této době se mluvilo o dot-com trendu, tedy ‚.com‘. Design webu se zaměřoval více na vzhled a multimédia. Stránky začali obsahovat více obrázků, videí a audio záznamů. V některých případech měli stránky animaci či intro. Dalším trikem pro upoutání pozornosti uživatele se stali pop-upy čili vyskakovací okna, obsahující přihlášení newstelleru nebo výhodný nabídky. Tyto trendy však načítání webových stránek výrazně prodloužili a snížili tím uživatelům dojem ze stránky.

Po tomto velmi interaktivním a také pro podniky drahém trendu se vývojáři vrátili k jednodušším rozložením, čistšímu zobrazení a použitelnosti, a tyto charaktery se stali prioritou. Do popředí se také dostával více a více CSS, který vývojářům umožňoval oddělit obsah od vzhledové prezentace stránky, což vedlo k efektivnosti a flexibilnějším rozložením stránek. Trendů bylo více, ale mezi ty nejvyužívanější patřilo minimalistické rozložení obsahu do jednoho sloupce s jednoduchou navigací, více prázdného místa a méně rozptylujících elementů, Sans-serif font, text psaný přes fotografie a jiné.

V roce 2004 se do povědomí začal dostávat termín WEB 2.0. a AJAX. Stránky se stali více interaktivní s větší schopností reagovat na uživatelské podněty. AJAX což v překladu stálo pro Asynchronní JavaScript a XML povolil načítání a aktualizaci obsahu bez potřeby plně znovu načíst celou stránku. Toto umožnilo stránkám funkce jako automatická kompletace vyhledávání, nekonečné posouvání obsahu, táhnout a pustit, nebo mimo jiné i rychlejší načítání.

Exponenciálně stoupal i trend mobilních zařízení, avšak většina webových stránek nebyla optimalizovaná pro rozložení na menší obrazovky. Responzivní web design vše změnil a pomohl stránky vytvářet pro adaptaci na jakoukoliv obrazovku a zařízení. S tímto, WEB 2.0. a responzivní web design, pomáhal vytvářet stránky, které byli pro uživatele více dynamické a poutavější.

Kolem roku 2010 se web design vydal směrem plochého designu a minimalistického vzhledu. Pro plochý design je charakteristické zaměření na plochou a 2D grafiku, tučné barvy a minimalistické ornamenty, také jednoduchý a otevřený rozložením a dostatek prázdného místa, velké obrázky a široké nadpisy, silná vizuální hierarchie a čistá navigace. V popředí tohoto směru byl Windows 8 anebo Apple se změnou vzhledu IOS 7, a celý tento směr byl reakcí proti přehnanému vzhledu a lesklé grafice vedoucího směru. Stránky se tedy zlepšili jak ve svém vzhledu, díky minimalistickému a responzivnímu rozložení, a také ve svém výkonu a efektivnosti.

Neumorfizmus se dostal do širšího okruhu web designu kolem roku 2020. Tento trend se popisuje jako hladký a dimenzionální uživatelský rozhraní, který využívá fyziky a světla, aby vytvořil jemný a organický vzhled. Neumorfismus spoléhá na jemný spád, stíny a světla pro vytvoření hloubky. Zakulacené okraje a tvary ještě více podporují organickou estetiku, v barvách převažuje bílá a pastelová s šedými stíny. Po tolika letech plochého designu přináší neumorfismus dimenzionální a prostorový vzhled ale přesto zůstává čistý a jednoduchý.

Se všemi trendy po sobě, je jasné že design se postupně zaměřil především na uživatelskou přívětivost, efektivnost a funkčnost s jednoduchým ale chytlavým vzhledem.

## UI a UX

Tyto dvě zkratky úzce souvisí s webovým designem. Zkratka UX označuje user experience neboli uživatelský zážitek a UI je zkratka pro user interface tedy uživatelské rozhraní. Tyto dvě terminologie se od sebe významně liší.

Uživatelský zážitek dělá rozhraní užitečným naopak uživatelské rozhraní ho dělá vizuálně přitažlivým. Jako první ale přichází uživatelský zážitek a po něm následuje uživatelské rozhraní. Zprvu se pracuje na tom pro koho, jak a k čemu bude web sloužit a až po stanovení těchto cílů se začne kolem výsledků budovat uživatelské rozhraní. Důležitost je nyní kladena více na uživatelský zážitek než na rozhraní. I na dále to však jsou dva důležité body pro růst webu.

V blízké budoucnosti by se mohla tato oblast uživatelského zážitku a rozhraní posunout o krok dál s přicházejícím trendem virtuální nebo rozšířené reality. První nápady a směr rozvoje již nedávno představila společnost Apple se svými Apple Vision Pro.

### UX

‚*Víte, co dělá architekt? UX designér jsou v podstatě architekti, ale na místo designování fyzických struktur, designujeme digitální strukturu. A stejně jako architekt fyzicky nestaví struktury, které nedesignuje, my se také spoléháme na programátory a vývojáře, aby postavili digitální strukturu, kterou my navrhneme*.‘ PEREYRA (2023:4).

Uživatelský zážitek je přesně to, co název napovídá, je to zážitek uživatele z digitálního produktu především z jeho užívání, někdy se označuje také jako pocit. Je to oblast designu kdy se web snaží zapůsobit na koncového uživatele například tím, jak uživateli zjednoduší dosáhnutí jeho cíle. Tato oblast designu se snaží vytvořit bezpečnější, efektivnější, vhodnou a uspokojivou stránku pro uživatele.

K tomu, aby produkt měl skvělý uživatelský zážitek je třeba mnoho základních parametrů, mezi ty nejdůležitější ale patří například využitelnost webu, která musí rozumět svým základním funkcionalitám a jak jich lehce dosáhnout, dále třeba konzistence designu, kdy není uživatelsky přívětivé, když se design odlišuje od základního nastaveného na daném webu, anebo také předejít, aby design byl pro uživatele obtížný, matoucí nebo moc bohatý na elementy, jako další je dostupnost, která zajišťuje, že web bude dostupný největší škále uživatelů včetně skupiny lidí s postiženími, nesmí ale také chybět estetický a emoční aspekt, který uživatelům činí web více žádoucím a atraktivním.

Jedny z oblastí které v poslední době ovlivňují uživatelský zážitek je například vzestup chytrých reproduktorů a hlasových asistentů, které vedli k vývoji hlasového rozhraní a změnili pohled uživatele na to jak může interagovat s technologií, dalším je větší personalizace díky které se šijí aplikace přímo na individuální potřeby a chování jedinců, a mezi směry které v poslední době ovlivňují nemalou oblast informačních technologií je AI a strojové učení, které napomáhají k více adaptivnímu a prediktivnímu prostředí webu.

Za zmínku stojí kniha Univerzální principy UX, PEREYRA (2023), jež popisuje sto strategií pro vytvoření pozitivní interakce mezi člověkem a technologií. Mezi tyto principy patří méně je více, první a poslední věc je nejvíce pamatována, shoda s reálným světem, minimalizace vstupu formuláře, učení ze špatných příkladů, stavba logické struktury, vzestup temného motivu, zpracování více vstupů a spoustu dalších.

### UI

Uživatelské rozhraní je více vizuální oblast. UI bere ideologii UX a používá ji na rozhraní produktu jako je menu webu, rozložení stránky, formuláře a další. V této oblasti je především důležité, aby produkt byl nejvíce intuitivní, jak to jde, proto se musí každý vizuální a interaktivní element, se kterým se uživatel může setkat, pečlivě vybírat. Musí se myslet na detaily jako typografie, mezery, odsazení, responzivitu, ikony, tlačítka, obrázky anebo také barevné schéma.

Rozhraní je vytvářeno tak aby co nejrychleji zachytilo pozornost uživatele. Uživatelské rozhraní upravuje uživatelský zážitek tím, že směřuje a navádí pozornost uživatele k důležitým oblastem. Zásadní složkou při vytváření uživatelského rozhraní je testování, v tomto kroku se mimo jiné zkoušejí nové barevné palety a tóniny, fonty a rozložení.

Mezi trendy v uživatelském rozhraní se prosadil tmavý režim, který nyní nabízí snad každá webová stránka a aplikace, jehož účelem je šetřit jak zrak uživatele, tak i baterii přístroje, mezi další trendy patří také již zmíněný neumorfizmus a i mikro interakce, což jsou v podstatě malé animace nebo designové elementy, které reagují na uživatelskou akci a předávají tak okamžitou zpětnou vazbu.

# Metodiky stylování

Vybraná stylování, která jsou základními a nejvyužívanějšími způsoby stylování jsou zde krátce popsány. Stylování se rozdělilo na prostý CSS, preprocesory CSS, CSS-in-JS a frameworky. Do praktického projektu byl vybrán vždy jeden z dané kategorie, avšak zde je popsán jak tento vybraný typ tak i jeho alternativy v dané kategorii, pro zobrazení rozdílu.

## CSS

CSS je zkratkou pro Cascading Style Sheets, v překladu kaskádové styly, je opensource jazyk pro kontrolu vzhledu webových stránek napsaných v HTML, XHTML a XML. CSS se představilo v listopadu roku 1994 na web konferenci v Chicagu a bylo navrženo Håkon Wium Liem pod organizací World Wide Web Consortium, zkráceně W3C, a bylo vydáno v prosinci roku 1996.

‚*Konečně můžeme interagovat na webu s elegancí a stylem! Tohle je přirozený a dlouho očekávaný krok v rozšíření snadno použitelného webu*.‘ LILJEREN (2003: w3.org).

CSS by však bylo pouhým akademickým zájmem, kdyby nebylo prohlížeče, který CSS podporoval, což byl Microsoft’s Internet Explorer 3, vydaný v srpnu téhož roku. Postupně se přidávali další prohlížeče jako Netscape, Opera a další.

Tento jazyk se dělí do verzí, které vždy stanoví nové způsoby vzhledu a rozložení. V roce 1996 to bylo CSS1, v roce 1998 CSS2, v roce 2011 CSS2.1, právě probíhající verze je CSS3 a ve vývoji již je CSS4.

### Verze CSS

První ucelenou verzí byla CSS1 v roce 1996. Tato verze již od organizace W3C v současné době není podporovaná. První verze přinesla práci s textem jako jsou vlastnosti fontu, atributy, zarovnání a barva textu, vzdálenosti mezi písmeny, slovy a řádky, mimo jiné představila verze i okraje, vnitřní a vnější okraje anebo také podporu tabulek a obrázků.

Se zvýšeným využíváním stylování vydali v květnu roku 1998 verzi CSS2. Tato verze již není udržovaná. V této verzi se pak přidala podpora z-indexu, stíny nebo také absolutní, relativní a fixní polohování. V červnu roku 2011 se vydala, a nahradila tak předešlou verzi, ucelená verze CSS2.1, která obsahovala především opravu problémů z verze CSS2.

Další verze nevyšla jako jednotný návrh jako předešlé verze. Verze CSS3 byl rozdělen do 4 modulů a tyto moduly se vydávali mezi rokem 2011 a 2012 pro zlepšení podpory prohlížečů a rozšíření vlastností verzí CSS2 a CSS2.1. Verze rozšířila úpravu webových stránek o box modely, efekty textu, tranzice a animace, pozadí a také o rozložení do více sloupců stejně jako flexbox a grid rozložení.

Společnost W3C stále ještě vyvíjí CSS poslední verzi, která je CSS4, a postupně aktualizují vlastnosti, které s sebou přináší, jako proměnné, filtry a detailnější práce s posouváním obsahu.

### Struktura CSS

CSS lze vložit do HTML třemi způsoby, a to vnitřní stylování, vnější stylování a řádkové stylování. Při vnějším stylování je vytvořen externí soubor s příponou .css, který poté lze využít na více stránkách. Tento externí soubor se dále nalinkuje pomocí elementu <link> do hlavičky HTML souboru. Vnitřní stylování je způsob, při kterém se kód stylování nachází přímo v souboru HTML. Tento kód se v souboru zaobalí do elementu <style> který se nachází v hlavičce HTML souboru. Tato metoda není velmi efektivní v případě, kdy má webová aplikace více než jednu stránku, jelikož vývojář musí vkládat kód stylování do každé jedné stránky, a v případě kdy bude potřeba upravit jednu položku stylování je potřeba ji upravit na všech stránkách. Řádkové stylování se nachází přímo v daném stylovaném elementu, jako třeba odstave nebo nadpis, v souboru HTML. Tomuto způsobu je lepší se vyhýbat. Jako předchozí způsob je třeba měnit stylování na každém elementu na každé stránce webu. Dalším důvodem, proč se tomuto způsobu vyhýbat je, že míchá dohromady jak obsah HTML souboru, tak CSS prezentaci a stylování obsahu, což dělá kód méně přehledným a srozumitelným.

Struktura .css souboru je ze základu tvořena selektory a deklaracemi. Selektor je první částí, kterou začíná CSS pravidlo a určuje, na jaký prvek či element v HTML souboru se pravidlo nebo styl vztahuje. Selektor se dále může dělit na specifičtější třídu anebo identifikátor, kde třída ovlivňuje všechny elementy se stejnou danou třídou a identifikátor ovlivňuje jeden daný element s jedním daným identifikátorem. U selektoru je také dobré zmínit, jakmile v případě, kdy se v .css souboru nachází více stejných selektorů, a každý udává jiné pravidlo stylování pro daný selektor, tak poslední selektor z těchto stejných je ten, co bude platit, jelikož přepisuje pravidla určena v selektorech před ním, tomu se říká kaskádové pravidlo. Deklarace je složena z vlastnosti a její hodnoty přímo uvnitř daného selektoru. Vlastnost poukazuje na specifickou stylistickou vlastnost, kterou se vývojář snaží změnit a hodnota určuje, jak se daná vlastnost bude stylizovat. Jako jednoduchý příklad lze zmínit vlastnost jako barva a hodnota vlastnosti stanovena na modrou.

## Preprocesory

Klasické CSS je základem pro stylování ale někdy může být zdlouhavé a náchylné k chybám, a to hlavně při tvorbě velkých projektů. Preprocesory CSS jsou nástroje, které usnadňují stylování a rozšiřují prosté CSS o nové syntaxe, proměnné, funkce a další užitečné vlastnosti.

K výhodám využívání preprocesorů patří proměnné, které mohou ukládat hodnoty pro snadné změny, modularita pro lepší přehlednost a udržitelnost, funkce k provádění matematických operací a manipulaci s hodnotami nebo nesting kdy dochází k vnořování selektorů pro lepší čitelnost.

### SASS

SASS je zkratkou pro Syntactically Awesome Style Sheets. Je to preprocesor, který rozšiřuje klasické CSS. Pomáhá vývojářům psát mnohem udržitelnější, znovu využitelný a lépe zorganizovaný kód. SASS soubory se zapisují s připonou .scss nebo .sass.

SASS vytvořil Hampton Catlin v roce 2006, v časech, kdy se CSS kódy stávali mnohem komplexnější díky nárustu dynamických webových aplikací. Prvotní verze SASS byla inspirována a také napsána v programovacím jazyku Ruby. V roce 2010 Natalie Weizenbaum a Chris Eppstein převzali vývoj SASS, což vedlo k vytvoření SCSS syntaxe, která je v dnešní době nejvíce využívanou verzí SASS.

Mezi další CSS preprocesory patří dále zmíněný Less, ale také méně využívanější Stylus nebo PostCSS. Jako přiklad webů, které využívají SASS pro stylování, lze zmínit Boostrap jeden z nejpopulárnějších CSS frameworků, dříve používal Less ale od čtvrté verze začal využívat SASS, GitHub, Airbnb nebo Kickstarter.

### Less

Název Less stojí pro Leaner Style Sheets a jedná se o jazyk, který rozšiřuje kaskádové styly. Byl vytvořen Alexis Sellierem a poprvé se objevil v roce 2009. Less byl napsán v programovacím jazyce Ruby, ale později se kód přepsal do JavaScriptu. Less patří mezi dva z nejpoužívanějších rozšíření CSS, která pomáhají vývojářům s lepší kontrolou nad napsaným kódem a zpřístupňují funkce a proměnné, tím druhým je SASS, kterým bylo Less inspirováno.

## CSS-in-JS

CSS-in-JS je metoda stylování kde je k určení a správě CSS stylu využíván JavaScript. Tento přístup ke stylování umožňuje vývojářům psát CSS napřímo do JavaScript souborů, kde mohou využít plný potenciál JavaScript jazyka pro dynamické stylování, témata a komponenty.

Tato metoda vznikla jako odpověď na výzvy vývojářů, se kterými se museli setkávat při budování rozsáhlých webových aplikací, a to především v kontextu komponentních architektur jako React. Jeden z problémů, kterým vývojáři čelili byl souhrnný rozsah, kdy ve velkých aplikacích mohl CSS lehce uniknout napříč komponenty, což vedlo k neočekávanému vzhledu stylování. S těmito velkými aplikacemi také narůstal počet CSS souborů, o které bylo potřeba se starat a udržovat. Jeden z hlavních problémů, který vedl k využití CSS-in-JS je dynamické stylování, jehož chování záleží také na stavech.

Klíčovými vlastnostmi metody je komponentně zaměřené stylování, zaměřené na individuální komponenty zabraňující souhrnné střety stylování. Kromě dynamického stylování je zde také vykreslení na straně serveru neboli SSR, umožňující generování a použití na straně serveru, což zlepšuje počáteční načítací výkon webu. Jelikož je stylování definováno v JavaScriptu je mnohem jednoduší rozpoznat a odebrat nevyužívající se kód při budovacím proces. Na druhou stranu výsledná velikost projketu s využitím CSS-in-JS se může zvýšit, pokud není projekt dobře optimalizován. Vývojáři designu také mohou shledat tuto metodu neobvyklou a jako větší výzvu k osvojení.

Nejpoužívanějšími CSS-in-JS jsou nyní Emotion, Styled Components, JSS anebo Stitches.

### Emotion

Emotion je silná a flexibilní CSS-in-JS knihovna. Je populární především v React ekosystému ale může být využita s jakýmkoli JavaScript frameworkem. Emotion nabízí dvě primární cesty, jak stylovat jedna je ‚styled‘ API, která je podobná Styled Components a ta druhá je ‚css‘ API, nabízející dynamičtější stylování.

Emotion knihovna vytvořena Mitchellem Hamiltonem, uvolněna v roce 2017, v důsledku rostoucí potřeby CSS-in-JS řešení. Knihovna byla designována tak aby byla vysoce výkonná s co nejmenší zátěží. V průběhu času se knihovna vyvinula, aby obsahovala vykreslování na straně serveru a další.

### Styled Components

Styled Components využívají literály pro stylování komponentů způsobem, který přijde jako přirozený pro vývojáře obeznámený s CSS, ale přitom využívající silné stránky JavaScriptu. Tato metoda přináší vývojářům možnost vytvořit zapouzdřené komponenty, rozsáhlé stylování a vylepšuje udržitelnost a škálovatelnost velkých projektů.

May Stoiber a Glen Maddern vyvinuli Styled Components a vydali v roce 2016. Rychle si tato metoda získala oblibu s jeho intuitivním API, silnou integrací na React a s možností psát klasický CSS do JavaScript souboru.

## Komponentní frameworky

CSS frameworky jsou sbírky předem připravených stylů, které poskytují základní rozvržení, typografii, komponenty a další prvky pro rychlé stylování webových aplikací.

Mezi výhody frameworků se řadí zrychlení vývoje a celkového procesu tvorby webu díky předem připraveným komponentám, konzistence zajišťující jednotný vzhled a chování celého projektu, responzivita přizpůsobující webové stránky různým velikostem zařízení a obrazovek a v neposlední řadě komunita uživatelů spolu s rozsáhlou dokumentací.

### MUI

Jinak známo jako Material-UI, je populární komponentní knihovnou. Poskytuje vývojářům komplexní sadu komponentů a nástrojů pro rychlejší vývoj konzistentních a responsivních webů. MUI je postaveno tak aby bylo vysoce přizpůsobitelné a povolilo vývojářům, aby si upravili vzhled a chování nabízených komponentů pro potřeby jejich projektu.

MUI projekt začal v roce 2014 Olivierem Tassinarim a Hai Nguyenem. Z počátku byl využíván vývojáři Reactu a poskytoval jim sadu znovu využitelných komponentů, které dodržovali principy Google Material Design standardu. Časem se MUI vyvinulo z obyčejné komponentní knihovny na komplexní nástroj zahrnující širokou škálu komponentů, témat a služeb. V roce 2021 se projekt přejmenoval z Material-UI na MUI což reflektovalo rozšířené soustředění za hranice Material Design standardu.

Jedna z klíčových vlastností MUI je bohatá knihovna s velkou kolekcí již postavených React komponent, mezi které náleží například tlačítka, karty, dialogová okna, formuláře, navigační elementy a mnohem více. Vývojáři dále mohou vytvářet vlastní tematické okruhy pro svůj projekt anebo MUI nabízí již defaultně možnost responzivního rozložení webové aplikace. MUI je velmi přizpůsobitelné a přístupné pro uživatele. V neposlední řadě MUI také podporuje TypeScript.

MUI je primárně využíváno na React aplikacích a vývojáři integrují MUI komponenty do jejich projektu instalací MUI knihovny pomocí npm nebo yarn. Mezi další komponentní knihovny spadá například Ant Design, Chakra UI, Semantic UI anebo také Bootstrap. Stránky využívající MUI jsou Netflix, Spotify, Amazon, Nasa anebo Unity.

### Bootstrap

Bootstrap je CSS knihovna, umožňující jednodušší práci s CSS. Tato knihovna je volně dostupná na jejich oficiálních stránkách getbootstrap.com a také má volně dostupný zdrojový kód, tedy je opensource. Bootstrap byl poprvé vydán v roce 2011 a to pod vývojem Twitteru, který Bootstrap používal jako interní nástroj, jejími autory je Mark Otto a Jacob Thornton. Bootstrap je založena na HTML5 a CSS3, ale pracuje také dobře i s JavaScriptem. Tato knihovna obsahuje šablony pro rozsáhlý okruh webových elementů jako jsou formuláře, tlačítka, ikony, tabulky, navigační panely, odsazení nebo varovná okna. Pomocí této knihovny je mnohem snazší dosáhnout responzivního vzhledu a také obsahuje širokou škálu motivů a témat pro jednotlivé elementy. S využitím této knihovny mohou vývojáři ušetřit spoustu času. A navrch knihovna má výbornou kompatibilitu s mnoha staršími prohlížeči.

Mezi největší milníky ve vývoji Bootstrap se dají považovat tři verze. První z nich je Bootstrap 3, která je nejstabilnější verzí. Je stále podporovaná a spravována vývojovým týmem. Bootstrap 4 doplnil knihovnu o nové komponenty, rychlejší šablonu a přidal více responzivních prvků, avšak od této verze přestal být podporován prohlížeč Internet Explorer 9 a nižší. Poslední stabilní verzí je Bootstrap 5, v níž se jQuerry nahradilo standartním JavaScriptem a ukončila se podpora pro prohlížeč Internet Explorer 11 a nižší.

### Tailwind CSS

Framework zdůrazňující vysokou přizpůsobivost. Na rozdíl od klasických CSS frameworků, které poskytují před designované komponenty tlačítek, karet a dalších, Tailwind poskytuje užitečné třídy, které jsou malé, recyklovatelné, pro jeden účel a tyto třídy vývojář může skládat a kombinovat pro vytvoření potřebného designu.

Tailwind byl poprvé vydán v roce 2017 a pracovali na něm Adam Wathan, Jonathan Reinink, David Hemphill a Steve Schoger. Tento projekt byl vytvořen jako řešení na limitace a neefektivnost, se kterými se vývojáři setkávali při používání tradičních CSS frameworků. Cílem Tailwind bylo opustit od sémantických názvů tříd a BEM metodologie, do které spadala struktura blok, element a modifikátor, a více se zaměřit na pragmatický, jednoúčelový přístup.

Jednoúčelový přístup je postaven kolem konceptu účelových tříd, které jsou jednoduché a atomické jako třeba ‚text-center‘. Tyto třídy jsou vkládány přímo do HTML souboru pro vytvoření komplexního designu. Další vlastností Tailwind je snadné přizpůsobení, kdy vývojář může upravovat konfigurační soubor Tailwind pro nastavení defaultních hodnot pro stylování. Tailwind obsahuje vlastnosti takzvaného očištění, kdy dochází k odebrání nevyužívaného CSS, což výrazně zlepšuje výkon, což se může prokázat jako skvělá vlastnost ve velkých projektech. V neposlední řadě má Tailwind bohatý systém pluginů, které pomáhají rozšiřovat funkčnost frameworku o další pomocné účinky a komponenty jako jsou formuláře, typografie nebo poměr stran.

# ZÁVĚR

Tato práce zpracovala teoretickou část metod stylování zpracováním historie webových aplikací a stránek a jejich visuálního představení. Historicky se zaměřilo na vývoj v průběhu dekád, které milníky se překonali a ke kterým se naopak vrátilo. Trendy převládající v daných érách webových aplikací se rozdělili do jednotlivých sekcí a nastínila se i budoucnost jakou se můžou webové aplikace po visuálním a uživatelském směru vydat.

Mimo popis historie a trendů se zvolili určité kategorie stylování a popsali se nejvyužívanější technologie v těchto kategoriích. Z těchto kategorií se vybrala vždy jedna metoda, která se dále využije při zpracování praktické části, tedy ke zpracování webové aplikace, která se bude skládat a visuálně tvarovat ke stanovené podobě, avšak za pomocí různých metod.

Práce má za úkol v budoucnu při její kompletaci pomáhat při výběru z, v dnešní době, mnoha metod, využívaných pro stylování a designovou stavbu webových aplikací. Má napomoct vybrat si třeba ne jenom jednu metodu ale i kombinaci více využitých metod, dle jejich využitelnosti, nebo jednoduchosti naprogramování, či rozšířené dokumentaci. Výběr může záležet i na velikosti aplikace, podpoře pomocných technologií, počtu dostupných knihoven a možností. Zobrazí rozdíl mezi jednotlivými metody a tím i zjednoduší výběr personalizovaný dle potřeb.

# SE**ZNAM ZDROJŮ**

ESKILSON, Stephen, 2023. Digital Design: A history. New Jersey: Princeton University Press. ISBN 978-069-125-3244

BOTAZZI, Roberto, 2018. Digital Architecture Beyond Computers. London: Bloomsbury Visual Arts. ISBN 978-1-4742-5813-5

BEAIRD, Jason; GEORGE, James a WALKER, Alex, 2020. The Principles of Beautiful Web Design. Melbourne: SitePoint. ISBN 978-1-925836-36-3

MCNEIL, Patrick, 2013. The Web Designer’s Idea Book, Volume 3: Inspiration from the Best Web Design Trends, Themes and Styles. Ohio: HOW Books. ISBN 978-1-4403-2396-6

PEREYRA, Irene, 2023. Universal Principles of UX. Beverly: Rockport Publisher. ISBN 978-0-7603-7804-5

HOFFMAN, Jay. 2017. A look back at the history of CSS. Online. Dostupné z: <https://css-tricks.com/look-back-history-css/> [cit 2024-06-24]

BOS, Bert. 2016. A brief history of CSS until 2016. Online. Dostupné z: <https://www.w3.org/Style/CSS20/history.html> [cit 2024-06-24]

RAJ, Anupama. 2023. History of CSS: The evolution of web design. Online. Dostupné z: <https://www.almabetter.com/bytes/articles/history-of-css> [cit 2024-06-25]

COLEMAN, Ali. 2024. The evolution of Web Design Trends through the ages. Online. Dostupné z: <https://www.amigostudios.co/blog/from-web-1-to-web-3-the-evolution-of-web-design-trends-through-the-ages> [cit 2024-06-24]

SHUBHAM. 2024. How web design trends changed over the years. Online. Dostupné z: <https://www.geeksforgeeks.org/how-web-design-trends-changed-over-the-years/> [cit 2024-06-26]

SELLIER, Alexis. 2023. Less. Online. Dostupné z: [https://lesscss.org/#](https://lesscss.org/) [cit 2024-06-24]

LITTLE, Chantelle. 2021. The history of web design. Online. Dostupné z: <https://tillerdigital.com/blog/the-history-of-web-design/> [cit 2024-06-26]

New perspective design. 2023. The history and evolution of web design. Online. Dostupné z: <https://www.newperspectivestudio.co.za/The-History-and-Evolution-of-Web-Design/> [cit 2024-06-24]

WIUM, Lie Hakon. 2006. The CSS saga. Online. Dostupné z: <https://www.w3.org/Style/LieBos2e/history/> [cit 2024-06-24]

UX a UI RGB.wiki. 2019. The Difference Between UX and UI Design. Online. Dostupné z: <https://rgb.wiki/blog/the-difference-between-ux-and-ui-design> [cit 2024-08-01]

MDN contributors. 2024. How CSS is structured. Online. Dostupné z: <https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/How_CSS_is_structured> [cit. 2024-08-01]