

Univerzita Pardubice
Fakulta elektrotechniky a informatiky

Návrh a vývoj informačního systému pro obce a města
Richard Hynek

Bakalářská práce
2023

ZADÁNÍ

Cílem této práce je vytvoření funkční aplikace, která bude umožňovat evidenci dat, publikaci informací pro veřejnost a správu vybraných procesů řízení obce. Aplikace bude umožňovat běžné funkcionality potřebné pro obec a město. Inovací bude propojení se sociálními sítěmi Facebook a Instagram (Social Media Integration). V teoretické části bakalářské práce bude realizována rešerše systémů, které se zabývají touto problematikou, a také popis použitých technologií. Praktická část bude obsahovat návrh databáze, ER diagram a aplikační řešení. Pro vytvoření aplikace bude využit skriptovací jazyk PHP a databáze MySQL. Pro rozhraní s uživatelem bude využit jazyk JavaScript, HTML a CSS.

PROHLAŠUJI:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 30. 4. 2023

Richard Hynek

ANOTACE

Tato bakalářská práce se zabývá návrhem a vývojem informačního systému pro obce a města. Součástí práce je i vytvoření funkční webové aplikace. Jsou zde popsány podmínky, které musí webová aplikace obce a měst splňovat. Další důležitou částí práce je popis jednotlivých využitých technologií k vytvoření webové aplikace. Hlavní část práce se věnuje popisu webové aplikace a problémům, které se musely vyřešit během tvorby. Na závěr jsou uvedeny možnosti využití webové aplikace a celého informačního systému v České republice.

KLÍČOVÁ SLOVA

Webová aplikace obcí a měst, tvorba webové aplikace, HTML, CSS, PHP, MySQL, internetové stránky, WordPress, integrace sociálních médií, informační systém obcí a měst

TITLE

Design and Development of an Information System for Municipalities and Cities.

ANNOTATION

This bachelor thesis deals with the design and development of an information system for municipalities and cities. Part of the work is the creation of a functional web application. This bachelor thesis describes the conditions that the web application and the information system of the municipality and cities must meet. Another important part of the work is a description of the various technologies used to create the web application. The main part of the work is devoted to the description of the web application and the problems that had to be solved during the creation. In the end, this bachelor thesis describes the possibilities of using the web application and the entire information system in the Czech Republic are presented.

KEYWORDS

Web application of municipalities and cities, creation of web application, HTML, CSS, PHP, MySQL, internet pages, integration of social media, WordPress, information system of municipalities and cities.

PODĚKOVÁNÍ

Tímto bych chtěl vyjádřit své upřímné poděkování všem, kteří mi pomohli při psaní mé bakalářské práce. Nejprve bych rád poděkoval svému vedoucímu práce za jeho cenné rady, připomínky a vedení během celého procesu tvorby práce. Díky jeho přístupu jsem byl schopen udržet si motivaci a nalézt správný směr pro moji práci.

Rád bych také poděkoval všem svým přátelům a rodině, kteří mi poskytli nezbytnou podporu a povzbuzení, ať už to bylo prostřednictvím sdílení svých zkušeností, poskytnutím potřebných materiálů nebo jednoduše nabízením svého času a posloucháním mých myšlenek. Dále bych rád poděkoval všem lidem, kteří mi poskytli konstruktivní zpětnou vazbu na mou práci, ať už to byli kolegové nebo přátelé. Bez jejich cenných nápadů a zpětné vazby by tato práce nemohla dosáhnout svého plného potenciálu.

Nakonec bych chtěl vyjádřit své díky všem autorům knih, článků a dalších zdrojů, které jsem využil při psaní své práce. Bez jejich přínosu bych nemohl získat potřebné znalosti a informace, které mi pomohly dokončit mou práci.

OBSAH

Seznam obrázků	8
Seznam zkratk	9
Úvod	10
1 Systémy zabývající se touto problematikou	11
1.1 Firmy zabývající se touto problematikou na Českém trhu	12
2 Webová aplikace	13
2.1 Zabezpečení	13
2.1.1 Šifrování	13
2.1.2 SSL/TLS	14
2.1.3 Útoky na SSL/TLS	16
2.1.4 Digitální certifikáty	19
2.2 Front-end a back-end webové aplikace	20
2.3 Responzivní webové stránky	20
3 Použité technologie	22
3.1 HTML	22
3.1.1 Tagy (značky)	22
3.1.2 Atributy	22
3.1.3 Struktura dokumentu HTML	23
3.2 Cascading Style Sheets (CSS)	24
3.2.1 Selektory	25
3.3 Bootstrap	26
3.4 JavaScript	27
3.4.1 Výhody JavaScriptu	27
3.4.2 Omezení JavaScriptu	28
3.5 PHP (Hypertext Preprocessor)	28
3.5.1 Bezpečnost a ochrana PHP	29
3.5.2 Postupy bezpečného kódování	29
3.5.3 Postupy ověřování	29
3.5.4 Testování zabezpečení	29

3.5.5	Další bezpečnostní opatření	29
3.6	MySQL	30
3.7	phpMyAdmin.....	30
3.8	WordPress.....	31
3.9	API.....	32
3.10	Social Media Integration.....	34
4	Návrh webové aplikace.....	36
4.1	Požadavky na aplikaci	36
4.2	Databáze.....	37
4.2.1	Tabulky v databázi WordPress	37
4.2.2	ER Diagram databáze WordPressu.....	40
4.3	Vlastní struktura tabulek pro Úřední desku	41
5	Realizace webové aplikace	42
5.1	Struktura dat.....	42
5.1.1	wp-admin	42
5.1.2	wp-content	43
5.1.3	wp-includes	44
5.1.4	Ostatní soubory	45
5.2	Základní nastavení	46
5.3	Úprava vybrané šablony dle potřebných požadavků	46
5.3.1	Vytvoření vlastních pluginů.....	46
5.3.2	Vlastní kategorie a vlastní pole.....	47
5.3.3	Custom Posty Type (CPT).....	47
5.3.4	Advanced Custom Fields (ACF).....	48
5.3.5	Úprava grafiky a responzivního designu	49
5.4	Segmentace webu	50
5.5	Import a tvorba obsahu	51
5.6	Implementace některých funkcí.....	52
5.6.1	Úřední deska	52
5.6.2	Napojení webu na Munipolis.....	53

5.6.3	Rezervační systém	54
5.6.4	Propojení se sociálními sítěmi	55
6	Umístění aplikace na web	57
6.1	Doména	57
6.2	Hosting	57
6.3	Zabezpečení SSL	58
6.4	Postup nahrávání	59
6.4.1	Zkopírování obsahu webové stránky na hostingové uložení	59
6.4.2	Vytvoření MySQL databáze	59
6.4.3	Nastavení souboru wp-config.php	60
6.4.4	Nastavení DNS	60
6.4.5	Nastavení WordPress	61
7	Závěr	62
	Použitá literatura	64
	Přílohy	67

SEZNAM OBRÁZKŮ

Obrázek 1 – HTTPS vs. HTTP	15
Obrázek 2 – Man In The Middle.....	16
Obrázek 3 – Zobrazení originální stránky https://www.apple.com/	17
Obrázek 4 – Navštívení podvrhnuté stránky v prohlížeči Firefox s podporou zobrazení punycode.....	18
Obrázek 5 – Navštívení podvrhnuté stránky v prohlížeči Google Chrome bez podpory zobrazení punycode	18
Obrázek 6 – Ukázka HTML.....	22
Obrázek 7 – Ukázka HTML atributů	23
Obrázek 8 – Základní struktura dokumentu HTML 5	24
Obrázek 9 – Ukázka CSS.....	25
Obrázek 10 – Bootstrap.....	26
Obrázek 11 – Vytvoření vlastního REST API Endpointu ve WordPress	33
Obrázek 12 – ER Diagram WordPress	40
Obrázek 13 – ER Diagram Úřední desky.....	41
Obrázek 14 – Výchozí nastavení nově nainstalovaného WordPressu	42
Obrázek 15 – Vlastní CPT typu Upozornění	48
Obrázek 16 – Vytvoření vlastního pole pro nadpis Upozornění.....	49
Obrázek 17 – Kód tvorby responzivního webu.....	50
Obrázek 18 – Připojení k databázi	52
Obrázek 19 – Načtení všech kategorií	53
Obrázek 20 – Načtení všech záznamů z úřední desky	53
Obrázek 21 – Grafický vzhled rezervačního kalendáře	54
Obrázek 22 – Administrační možnosti kalendáře	55
Obrázek 23 – Přehled rezervovaných hodin	55
Obrázek 24 – Připojení WordPressu k databázi.....	60

SEZNAM ZKRATEK

WWW	World Wide Web
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
SQL	Structured Query Language
API	Application Programming Interface
SSL	Secure Sockets Layer
RSA	Rivest–Shamir–Adleman
TLS	Transport Layer Security
POP	Post Office Protocol
IMAP	Internet Message Access Protocol
SMTP	Simple Mail Transfer Protocol
MITM	Man in the Middle
ER	Entity–relationship
ID	Identifier
DNS	Domain Name System

ÚVOD

Žijeme v 21. století plné digitalizace a modernizace, mezi které patří i pojem WWW stránky (World Wide Web). V moderních krajinách Evropy, mezi které patří i Česko se tento pojem stal již samozřejmostí. V dnešní době téměř každý denně používá webové stránky a internet. Z toho důvodu je nezbytně nutné mít veškeré informace na internetu v podobě webových stránek. Ať už se jedná o nezbytné informace, propagaci firmy, zveřejňování svého portfolia nebo provoz vlastního e-shopu, vždy je zapotřebí webová aplikace a internet. Pro složitější projekty pouze webová stránka nestačí a musí zde být vytvořen i složitý informační systém. Ať už se jedná o složitý informační systém, nebo o základní statickou webovou stránku, uživatelské prostředí musí být velmi intuitivní a snadno použitelné. Z toho důvodu vzniká mnoho odvětví, každý se specializuje na něco jiného a velmi často je složité všechny části projektu složit dohromady. Mezi složitější projekty patří i webové aplikace pro obce a města, které musí splňovat všechny zákony, zajišťovat neustálou funkčnost, být vzhledově intuitivní pro občany i návštěvníky a shromažďovat důležitá data.

Autor této práce se několik let se vzdělává ve webových aplikacích, kde rovněž pracuje. Důvodem výběru tohoto tématu bylo získání nových zkušeností v pokročilých a složitějších systémech. Autor se tedy začal o toto téma více zajímat a začal přemýšlet o samotné realizaci webové aplikace. Během tvorby čelil mnoha problémům a musel se naučit nové informace z mnoha technologických aspektů a programátorských řešení. Příkladem může být zprovoznění efektivní funkčnosti archivu úřední desky.

Hlavním cílem této práce bylo vytvořit plně funkční, uživatelsky atraktivní a designově moderní webovou aplikaci, kterou bude v budoucnu možné použít jako systém pro obce a města. Systém splňuje veškeré nezbytné funkční požadavky obce, moderní webové aplikace a všechna právní nařízení. Jedinečnou funkcí je propojení webové aplikace s vlastní úřední deskou, rezervačním systémem haly, obecního domu a propojení webové stránky se sociálními médii (Social media integration). Systém je připraven na následující rozšíření, která mohou být v budoucnu implementována. Příkladem rozšiřující funkce může být integrace s obecním rozhlasem.

Na tvorbu webových stránek byla použita známá a velmi rozšířená technologie webových stránek, a to konkrétně redakční systém WordPress, skriptovací a programovací jazyk PHP, relační databázový server MySQL, kaskádové styly CSS, kódovací jazyk HTML 5 a skriptovací a programovací jazyk JavaScript. Pro správné pochopení této práce jsou potřebné alespoň minimální znalosti v oblasti informačních technologií a tvorby webu.

1 SYSTÉMY ZABÝVAJÍCÍ SE TOUTO PROBLEMATIKOU

Informační systémy pro obce a města patří mezi klíčové nástroje moderního veřejného sektoru. Tyto systémy pomáhají obcím a městům v efektivním řízení, správě jejich agend, koordinaci aktivit, zlepšení kvality života občanů, a hlavně k usnadnění komunikace mezi občany a veřejnou správou. Vývoj těchto informačních systémů je však náročný proces, který vyžaduje odbornost a zkušenosti v oblasti informačních technologií.

Jeden z hlavních cílů vývoje informačních systémů pro obce a města je optimalizace a automatizace interních procesů. Tyto systémy pomáhají snižovat náklady a zlepšovat efektivitu práce úředníků, kteří se tak mohou více věnovat komunikaci s občany a řešení konkrétních problémů. Vývoj těchto systémů je však velmi specifický a vyžaduje znalosti z oblasti veřejné správy, územního plánování, finančního řízení a mnoha dalších oblastí. Informační systémy pro obce a města musí být navrženy tak, aby byly co nejvíce přizpůsobitelné konkrétním potřebám a představám občanů. Vývoj těchto systémů musí být prováděn v úzké spolupráci s občany a správou obce s maximálním zapojením jejich názorů a požadavků. Tento přístup umožňuje vytvořit systémy, které jsou efektivní, uživatelsky přívětivé a plně odpovídající potřebám obcí a měst. Vývoj informačních systémů pro obce a města musí být v souladu s legislativními požadavky. Tyto systémy musí například zajišťovat ochranu osobních údajů občanů, splňovat právní povinnosti webové stránky obce a další zákonné požadavky. Důležitou roli hraje také technická podpora, která musí být poskytována v případě problémů s funkcionalitou systému. Dalším důležitým faktorem při vývoji informačních systémů pro obce a města je integrace s dalšími systémy a službami. Integrace s těmito systémy zvyšuje efektivitu práce úředníků a umožňuje rychlé a spolehlivé zpracování dat.

Mezi další důležité prvky, které by měly být zohledněny při vývoji informačních systémů pro obce a města, patří například uživatelská přívětivost, což znamená, že systémy musí být jednoduše ovladatelné i pro uživatele bez hlubších znalostí IT. Důležitou roli hrají také funkce pro podporu sdílení dat a komunikace mezi úředníky, občany a dalšími subjekty.

V současné době existuje mnoho firem a organizací, které se specializují na vývoj informačních systémů pro obce a města. Tyto firmy nabízejí široké spektrum služeb, včetně konzultací, návrhu a implementace systému, technické podpory, školení a dalších služeb. Důležité je vybrat si kvalitního a zkušeného dodavatele, který má předchozí zkušenosti s vývojem systémů pro veřejnou správu a který je schopen přizpůsobit se konkrétním požadavkům klientů.

Vývoj informačních systémů pro obce a města je náročný proces, který vyžaduje odbornost a zkušenosti v oblasti informačních technologií, veřejné správy, územního plánování a dalších oblastí. Tyto systémy pomáhají obcím a městům v efektivním řízení, správě jejich agend, vyšší kvalitě služeb pro občany a větší transparentnosti celého procesu. Vzhledem k důležitosti těchto systémů pro efektivní fungování měst a obcí je nutné zajistit kvalitní vývoj a údržbu. V poslední době se také v oblasti vývoje informačních systémů pro obce a města objevují nové trendy a technologie, které přináší nové možnosti a výhody. Například využití umělé inteligence a strojového učení může pomoci v automatizaci některých procesů a zvýšení efektivity práce úředníků. Dále je stále více využíváno mobilních aplikací, které umožňují občanům přístup k důležitým informacím a službám přímo z jejich mobilních zařízení.

Vývoj informačních systémů pro obce a města je důležitým tématem v oblasti veřejné správy a informačních technologií. Tyto systémy umožňují efektivní řízení a správu agend měst a obcí, větší transparentnost a kvalitu služeb pro občany. Důležité je však zajistit kvalitní vývoj a údržbu těchto systémů, a to včetně bezpečnosti dat a respektování soukromí občanů. S rozvojem nových technologií a trendů v oblasti informačních systémů se očekává další rozvoj a inovace v této oblasti, které přinesou další výhody a možnosti pro města, obce a jejich občany.

1.1 Firmy zabývající se touto problematikou na Českém trhu

- Obce na webu (<https://obcenawebu.cz/>)
- Stránky pro obce (<https://www.strankyproobce.cz/>)
- Igalileo.cz (<https://www.igalileo.cz/>)
- Pro obce (<https://www.proobce.cz/>)
- Wertu (<https://www.wertu.cz/weby-pro-mesta-a-obce/>)

2 WEBOVÁ APLIKACE

Webová aplikace je program spuštěný na vzdáleném webovém serveru, který umožňuje internetovým uživatelům vytvořenou aplikaci zobrazit a používat její obsah. Veškerá komunikace mezi klientem a webovým serverem (webovým prohlížečem) probíhá pomocí nezabezpečeného protokolu Hypertext Transfer Protocol (dále jen HTTP) nebo zabezpečeného protokolu Hypertext Transfer Protocol Secure (dále jen HTTPS). Protokol HTTP/HTTPS je na poslední 7. (aplikační) vrstvě modelu ISO/OSI. V dnešní době je více než 82.5 % webů zabezpečeno protokolem HTTPS. [1]

2.1 Zabezpečení

Původně byla data na webu přenášena v prostém textu, který si mohl v případě zachycení zprávy kdokoli přečíst. Pokud například spotřebitel navštívil webovou stránku e-shopu, kde zadal objednávku společně s údaji své kreditní karty, všechny jeho vyplněné údaje putovaly po internetu neskryté.

Protokol SSL byl vytvořen za účelem odstranění tohoto problému a zajištění ochrany soukromí uživatelů. SSL, šifrováním všech dat procházejících mezi uživatelem a webovým serverem, zajišťuje nečitelnost dat v případě zachycení útočníkem. Útočník nyní pouze uvidí zakódované znaky. Vyplněné údaje spotřebitelem jsou nyní v bezpečí. Údaje jsou viditelné pouze pro nákupní web, kde byly zadány. [2]

2.1.1 Šifrování

Principem šifrování je převod čitelných dat na nečitelná data. Zpětný převod z nečitelných dat na čitelná data se nazývá dešifrování.

2.1.1.1 Symetrické šifrování

- Stejný sdílený klíč použit pro šifrování i dešifrování,
- odesílatel a příjemce se nejprve musí domluvit na tajném klíči (sekvenci znaků),
- odesílatel zprávu zašifruje, zašle ji příjemci a ten si ji dešifruje,
- výhodou je nízká výpočetní náročnost a s tím související rychlost,
- největší slabinou je nutnost sdílení klíče s každým, kdo má zprávu šifrovat/dešifrovat,
- nepoužívanější techniku symetrického šifrování je AES.

2.1.1.2 Asymetrické šifrování

- Řeší slabinu symetrického šifrování,
- používají se dva klíče – jeden pro šifrování (veřejný) a druhý pro dešifrování (soukromý),
- nejprve odesílatel požádá příjemce o jeho šifrovací klíč,
- tímto klíčem zašifruje zprávu a odešle ji,
- příjemce si pak zprávu svým soukromým klíčem dešifruje,
- výhodou je, že neexistuje cesta, jak se dostat k šifrovanému obsahu za našich životů,
- nejpoužívanější technika asymetrického šifrování se nazývá RSA.

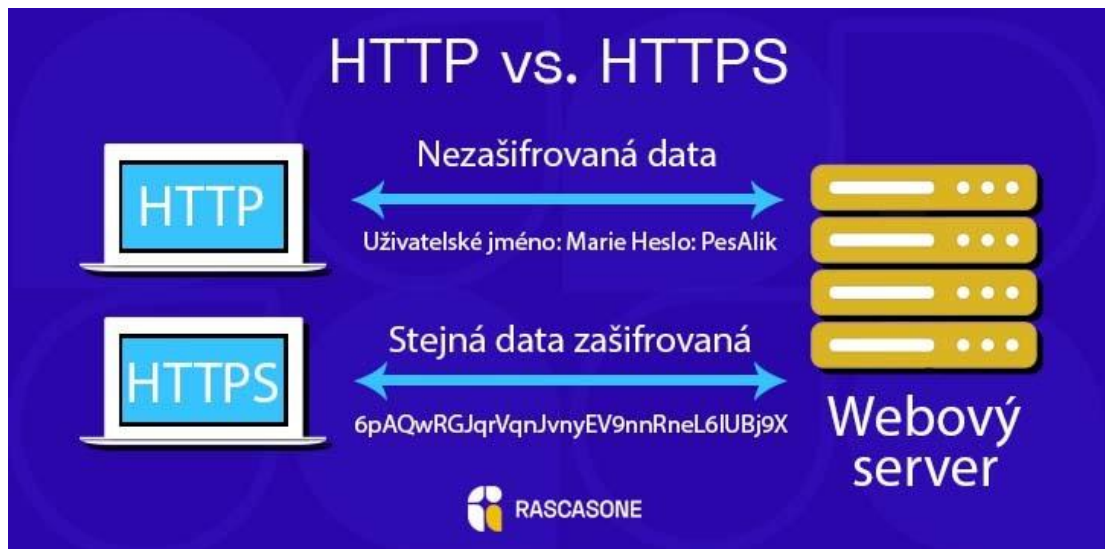
2.1.2 SSL/TLS

Secure Sockets Layer (SSL) je standardní internetový bezpečnostní protokol pro udržení bezpečného internetového připojení a ochranu citlivých dat, která jsou odesílána mezi serverem a klientem nebo serverem a serverem. SSL brání zločincům ve čtení a úpravě přenášených informací, včetně potenciálních osobních údajů.

SSL byl vyvinut společností Netscape v roce 1995 za účelem zajištění soukromí a ochrany dat v internetové komunikaci. SSL, konkrétně SSL 3.0, je předchůdcem moderního dnes používaného šifrování Transport Layer Security (TLS).

TLS bylo představeno v roce 1999. Jedná se o otevřený a všem dostupný standard, určený pro jakoukoliv společnost či projekt - tedy bez patentní nálepky Netscapu. U TLS 1.0 se již počítalo s poštovními protokoly POP3, SMTP a IMAP. Opraveny byly také některé bezpečnostní chyby, které mělo SSL 3.0. [3]

Web, který implementuje SSL/TLS, má ve své URL adrese HTTPS místo HTTP.



Obrázek 1 – HTTPS vs. HTTP

Zdroj: <https://www.rascasone.com/cs/blog/co-je-https-http-ssl-tls>

Prolomit SSL/TLS není nemožné. Je to pouze nemožné v rozumném množství času pomocí současných výpočetních zdrojů.

Vytvoření SSL spojení funguje na principu asymetrické šifry, kde každá z komunikujících stran má veřejný a soukromý klíč. Veřejný klíč je nutné zveřejnit a zajistit jeho správné předání všem, kteří jej budou chtít použít. Je zajištěno, že pokud kdokoliv pomocí tohoto veřejného klíče zašifruje zprávu, tak ji bude moci rozšifrovat pouze majitel veřejného klíče, který odpovídá soukromému klíči majitele.

Vytvoření SSL spojení (SSL handshake, tedy potřásání rukou) pak probíhá následovně:

- klient pošle serveru požadavek na SSL spojení,
- server pošle klientovi odpověď na jeho požadavek obsahující certifikát serveru,
- na základě přijatého certifikátu si klient ověří autentičnost serveru,
- certifikát také obsahuje veřejný klíč serveru,
- klient vygeneruje základ šifrovacího klíče, kterým se bude šifrovat následná komunikace,
- ten zašifruje veřejným klíčem serveru a pošle mu ho,
- server použije svůj soukromý klíč k rozšifrování základu šifrovacího klíče,
- z tohoto základu vygenerují jak server, tak klient hlavní šifrovací klíč,
- klient a server si navzájem potvrdí, že od teď bude jejich komunikace šifrovaná tímto klíčem,
- fáze handshake tímto končí,
- je ustaveno zabezpečené spojení šifrované vygenerovaným šifrovacím klíčem,

- aplikace od teď dál komunikují přes šifrované spojení.

Server může požadovat, aby byl certifikát vystaven důvěryhodnou certifikační autoritou.

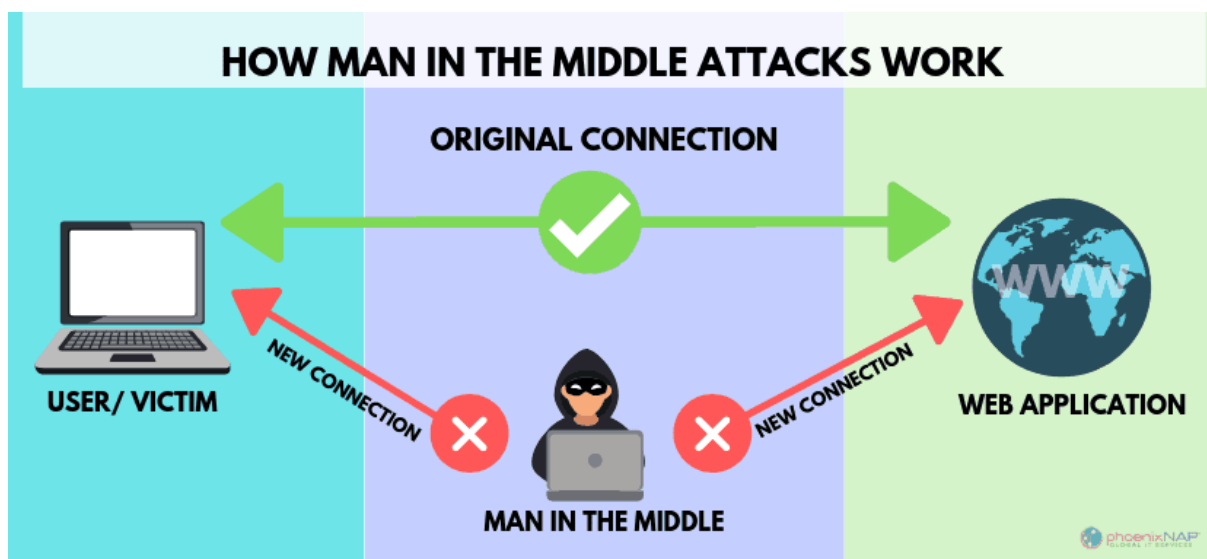
2.1.3 Útoky na SSL/TLS

Jednou z možností útoku na šifrovanou komunikaci protokolem SSL/TLS je odcizení privátního klíče přímo ze strany serveru. Pokud je ale server dostatečně zabezpečen a dodržuje bezpečnostní politiky, je odcizení privátního klíče prakticky nemožné.

2.1.3.1 Man In The Middle (MITM)

Reálnější hrozbou jsou útoky založené na MITM. Jedná se o útok, při kterém se útočník stává tajným prostředníkem mezi dvěma zařízeními v navázané komunikaci. Útočník může komunikaci odposlouchávat, měnit a získávat citlivá data bez zjištění skutečnosti komunikujících stran. Komunikující strany si myslí, že komunikují pouze mezi sebou.

Provést útok MITM je možné na začátku komunikace, kdy oběť zasílá serveru požadavek na SSL spojení, který útočník odchytí. Útočník naváže šifrované spojení se serverem a oběti odešle podvržený certifikát. Komunikace již nadále probíhá skrze útočníka.



Obrázek 2 – Man In The Middle

Zdroj: <https://phoenixnap.com/blog/man-in-the-middle-attacks-prevention>

Tato slabina lze vyřešit pomocí certifikační autority, která jim na základě žádosti ověří a vydá certifikát sloužící jako doklad o jejich pravosti. Certifikační autoritě musíme důvěřovat.

2.1.3.2 SSL spoofing

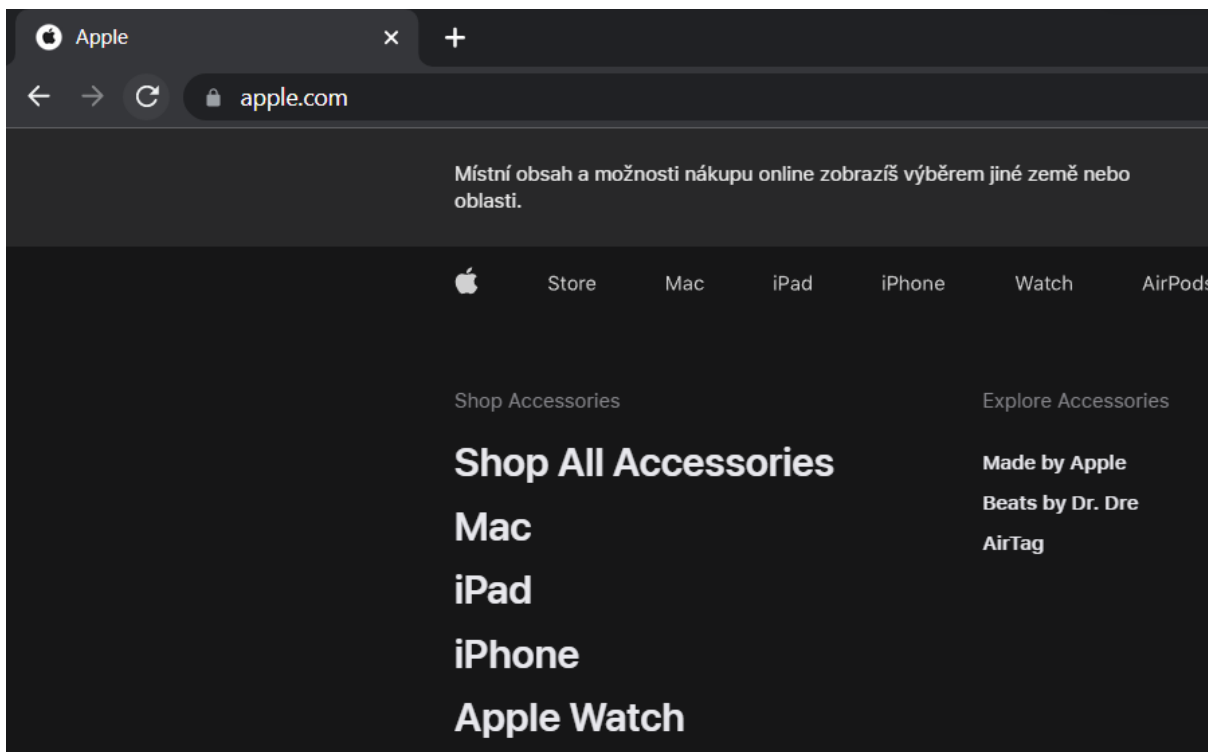
Jedná se o běžnou metodou útoku, ve kterém útočník používá doménu, která vypadá velmi podobně jako cílový důvěryhodný web. SSL spoofing neútočí na samotné SSL, ale většinou na přechod z šifrované komunikace na nešifrovanou.

2.1.3.3 HTTPS spoofing

Znaky v cílové doméně jsou nahrazeny non-ASCII znaky, které jsou vzhledově velmi podobné ASCII znakům. Útočníci nejdříve zaregistrují název domény, který je podobný cílovému důvěryhodnému webu, a také zaregistrují jeho certifikát SSL, aby falešná verze webu vypadala legitimně a bezpečně. Poté už jen pošlou odkaz oběti.

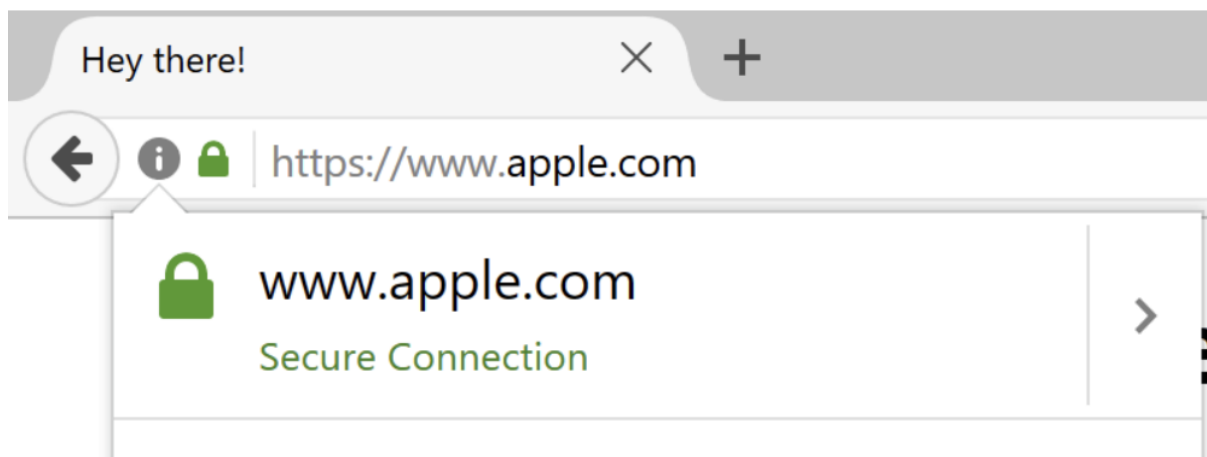
Vzhledem k tomu, že většina prohlížečů podporuje zobrazení domény v [punycode](#), klient si při navštívení adresy ani nevšimne, že se jedná o falešnou verzi webu. Prohlížeč klienta navíc ukazuje, že certifikát webu je legitimní a bezpečný, což ztěžuje detekci tohoto útoku. Klient si tedy myslí, že interaguje s legitimní šifrovanou webovou stránkou, ale ve skutečnosti se stal obětí útoku typu MITM a poskytuje své informace útočníkovi.

Bezpečnostní výzkumník Xudong Zheng ukázal důkaz tohoto útoku, ve kterém podvrhl zabezpečený HTTPS web [apple.com](#).



Obrázek 3 – Zobrazení originální stránky <https://www.apple.com/>

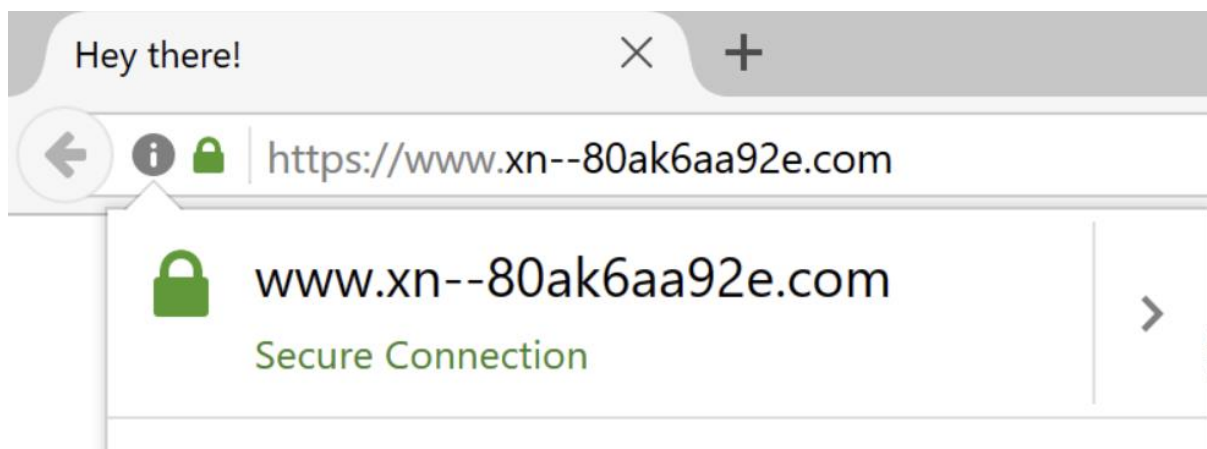
Zdroj: autor



Obrázek 4 – Navštívení podvrhnuté stránky v prohlížeči Firefox s podporou zobrazení punycode

Zdroj: autor

Jedním ze způsobů, jak zabránit falšování HTTPS, je zakázat podporu zobrazení punycode v prohlížeči. Tím zajistíte, že se v adresním řádku zobrazí skutečný název domény a upozorní vás, pokud navštívíte neautentickou webovou stránku. Když například navštívíte adresu podvrhnutou adresu, v adresním řádku uvidíte <https://www.xn--80ak6aa92e.com> místo „apple“ v azbuce psané jako apple. Převod textu do punycode lze vyzkoušet stránce punycoder.com



Obrázek 5 – Navštívení podvrhnuté stránky v prohlížeči Google Chrome bez podpory zobrazení punycode

Zdroj: autor

2.1.3.4 SSL Stripping

SSL Stripping je technika, při které je webové připojení převedeno ze zabezpečeného HTTPS protokolu na nezabezpečený protokol HTTP. Jedná se o způsob obcházení zabezpečení, které je běžně vynuceno certifikáty SSL na stránkách s protokolem HTTPS. Většinou se jedná o případ, kdy oběť vyšle požadavek z nezabezpečené HTTP stránky na

zabezpečenou HTTPS. Oběť komunikuje s útočником pomocí nezabezpečeného HTTP protokolu a útočnik přeposílá data serveru pomocí zabezpečeného protokolu.

Ohranou před SSL Stripping je kontrola adresního řádku svého prohlížeče, kde je poznat, že připojení je nešifrované. Další možnou ochranou je instalace HTTPS Everywhere. HTTPS Everywhere je rozšíření prohlížeče, které vynucuje komunikaci HTTPS.

2.1.4 Digitální certifikáty

Digitální certifikát je podepsaný veřejný šifrovací klíč sloužící pro identifikaci jednotlivých subjektů, které spolu komunikují. Podpis se použije k ověření identity odesílatele a zároveň k potvrzení, že odeslaná zpráva nebyla upravena třetí stranou.

Digitální certifikát vydává certifikační autorita (CA), která se zaručila za jeho správnost. Certifikační autority jsou společnosti, které na základě žádosti ověřují subjekty, a následně jim vydávají certifikáty sloužící jako doklad o jejich pravosti. CA potvrzuje pravdivost údajů a zajišťuje komunikaci se správným subjektem. Na základě principu přenosu důvěry můžeme důvěřovat údajům uvedeným v digitálním certifikátu za předpokladu, že důvěřujeme samotné certifikační autoritě.

Pro certifikáty se používá mezinárodní norma X.509, která jednoznačně popisuje strukturu certifikátu. Každý certifikát musí obsahovat následující údaje:

- sériové číslo, které musí být vždy unikátní,
- datum počátku a konce platnosti certifikátu,
- identifikační údaje subjektu, kterému je certifikát vydán (CA musí tyto údaje důkladně ověřit → například OP),
- typ algoritmu a veřejný klíč, nejčastěji o délce 2048 bitů,
- identifikační údaje certifikační autority.

Certifikát je běžný datový soubor, proto je potřeba zabránit, aby on sám nemohl být zfalšován. Proto certifikáty obsahují podpis soukromým klíčem od samotné certifikační autority. Platnost certifikátu lze ověřit pouze tím, že máme k dispozici také certifikát certifikační autority. Ten je zpravidla ke stažení na stránkách certifikační autority.

Certifikáty podporují také běžné operační systémy. Do seznamu kořenových certifikátů systému jsou většinou implementovány certifikáty [důvěryhodných a ověřených certifikačních autorit](#). [4]

2.2 Front-end a back-end webové aplikace

Každá webová aplikace se skládá z front-endu a back-endu. Front-end je viditelná část aplikace, která je zobrazena uživateli. Uživatel front-end vidí a může ho používat pomocí webového prohlížeče. Zobrazený výsledek, který se zobrazí uživateli, je zpracovaný kód zformulovaný do hypertextového dokumentu, který je poslán přes protokol HTTP nebo HTTPS. Zpracovaný kód je vytvořen značkovacím jazykem Hypertext Markup Language (dále jen HTML). Stylizace prvků a design webové aplikace se tvoří pomocí kaskádových stylů Cascading Style Sheets (dále jen CSS). Případné pokročilejší, rozšířené a interaktivní grafické funkce ve front-endu jsou prováděny pomocí JavaScriptu. Kód JavaScriptu se vykonává až v prohlížeči klienta. JavaScripta má ale většinou velmi negativní obsah na rychlost webu, proto je lepší JavaScript používat jen zřídka v nezbytných situacích. V následující kapitole budou tyto použité technologie popsány podrobněji.

O funkčnost webové aplikace se stará back-end, který není viditelný pro klienta. Zde se vykonává veškerá logika webových aplikací jako je například tvorba dynamického obsahu pomocí programovacích jazyků, komunikace s databází a podobné pokročilé funkce. Může se jednat o rezervační systém, e-commerce, generování obsahu na základě vyplněného formuláře, přihlášení a registrace a mnoho dalšího dynamického obsahu. Hypertextový dokument, který je odeslán klientovi, se tvoří až po provedení všech operací.

Existují tedy webové stránky statické a dynamické. U statických stránek je obsah již předem připravený a je rovnou odeslán klientovi v připraveném hypertextovém dokumentu. U dynamického obsahu se obsah vytváří až na základě vykonaných funkcí a požadovaných akcí. Například se generuje obsah na základě nejčtenějších příspěvků, nejvíce nakupovaných produktů, naposledy navštívených stránek, nebo také po vyplnění formuláře.

2.3 Responzivní webové stránky

Moderní responzivní webové stránky jsou klíčovým prvkem v současném digitálním světě, kde uživatelé používají širokou škálu různých zařízení, jako jsou mobilní telefony, tablety, notebooky a desktopy. Tato rozmanitost zařízení představuje výzvu pro vývojáře webových stránek a aplikací, kteří musí zajistit, aby byly stránky snadno použitelné a přístupné na všech těchto zařízeních, bez ohledu na různé typy rozlišení.

Pro tvorbu moderních responzivních webových stránek se používají nejnovější technologie a postupy. Jedním z hlavních prvků je použití kaskádových stylů (CSS), které umožňují

flexibilní rozvržení (flexbox). Toto rozvržení umožňuje tvůrcům webových stránek vytvářet responzivní design, který se automaticky přizpůsobí velikosti obrazovky uživatele. Dalším klíčovým prvkem je použití responsivních obrázků, které se automaticky přizpůsobují velikosti obrazovky. To umožňuje uživatelům rychle načíst stránky bez nutnosti čekání na načtení velkých obrázků, které by zpomalily načítání stránky na mobilních zařízeních. V současné době se také používají moderní JavaScriptové knihovny a rámce jako React, Angular nebo Vue.js, které umožňují tvůrcům webových stránek vytvářet dynamické a interaktivní uživatelské rozhraní.

Vývoj moderních responzivních webových stránek vyžaduje od tvůrců neustálou pozornost a sledování nových technologií a postupů. Výsledkem však je přístupná a snadno použitelná webová stránka, která poskytuje uživatelům skvělý zážitek bez ohledu na zařízení, které používají. [8][20]

3 POUŽITÉ TECHNOLOGIE

3.1 HTML

HTML je oficiální standardní značkovací protokol pro tvorbu webových stránek, který rozšiřuje text o další informace. Nejčastější použití HTML je při vývoji webů, kde ho vývojáři používají k návrhu zobrazení jednotlivých prvků webové stránky. Umožňuje vývojářům vytvářet záhlaví, zápatí, navigaci, sekce, odstavce a odkazy. HTML není považován za programovací jazyk, protože nemůže vytvářet dynamické funkce. Pro HTML nepotřebujeme žádný specializovaný program, kódovat lze i v textovém editoru.

```
<!-- HEADER : begin -->
▼<header id="header">
  ▼<div class="header__inner">
    <!-- HEADER CONTENT : begin -->
      ▼<div class="header__content">
        ▼<div class="lsvr-container">
          ▶<div class="header__content-inner">...</div> == $0
        </div>
      </div>
    </div>
    <!-- HEADER CONTENT : end -->
  </div>
</header>
<!-- HEADER : end -->
```

Obrázek 6 – Ukázka HTML

Zdroj: autor

3.1.1 Tagy (značky)

Vývojář vytváří obsah pomocí tagů (značek) a atributů. Tagy se dělí na párové a nepárové. Značky párové mají stejnou značku na začátku i na konci, pouze na konci je před názvem znak lomítka. Příklad pro označení hlavního nadpisu stránky může vypadat takto:

```
<h1>Hlavní nadpis stránky</h1>.
```

3.1.2 Atributy

Atributy v jazyce HTML slouží jako parametry, které nesou další informace o obsahu značek. Atributy se vždy zapisují do první značky elementu mezi ostré závorky. Jedná se o velmi důležitou funkci, která pomáhá vytvářet pokročilejší funkce na webu. Pomocí HTML parametrů lze předávat hodnoty například do CSS, JavaScriptu nebo PHP. Atributy vyžadují název a hodnotu. Mezi hlavní atributy může patřit například class, type, name, style a id.

```

    <div class="custom-col-100">
      <button type="submit" id="filtryByDate" name="filtryByDate">Zrušit filtr podle data</button>
    </div>
  </div>
  <h2 class="table-header-active">OBECNĚ ZÁVAZNÉ VYHLÁŠKY
</h2>

```

Obrázek 7 – Ukázka HTML atributů

Zdroj: autor

Název atributu udává druh dalších informací o obsahu prvku. Název by měl být psán malými písmeny. Hodnota je informace nebo nastavení atributu. Hodnota by měla být umístěna ve dvojitéch uvozovkách. **nazev="hodnota"**. Různé atributy mohou mít různé hodnoty. Příkladem může být **lang="cs"**. Použití atributu lang slouží k označení jazyka použitého v tomto prvku. Hodnota tohoto atributu na této stránce uvádí, že je v českém jazyce.

Většinu atributů lze použít pouze u určitých prvků (neboli tagů), i když několik atributů (například lang) se může objevit u jakéhokoli prvku. Většina hodnot atributů je buď předdefinovaná, nebo se řídí stanoveným formátem. Při zavádění každého nového atributu je zapotřebí dodržet povolené hodnoty. Hodnota atributu lang je zkrácený způsob, jak určit, který jazyk se používá uvnitř prvku, kterému rozumí všechny prohlížeče.

Zatímco několik atributů, jako je lang, lze použít pro jakoukoli značku, většinu atributů lze použít pouze na určité značky. Většina hodnot atributů je předem definovaná nebo se drží předem určené struktury. Při zavádění každého nového atributu je důležité znát povolené hodnoty. Hodnota atributu lang je zkrácený způsob, jak určit, který jazyk se používá uvnitř prvku, kterému rozumí všechny prohlížeče. [5]

3.1.3 Struktura dokumentu HTML

Každý HTML soubor má předepsanou strukturu, kterou je důležité dodržovat. V každém dokumentu by neměly chybět základní značky jako:

- **Značka <!DOCTYPE html>** – Deklarace typu dokumentu – sděluje prohlížeči, že otevřel HTML dokument.
- **Párové značky <html> a </html>** – Kořenový element – reprezentuje celý dokument.
- **Párové značky <head> a </head>** – Hlavička dokumentu – Obsahuje informace o stránce, která většinou nejsou vidět. Obsahuje metadata, která se vztahují k celému

dokumentu. Definuje např. kódování, název dokumentu, autora, popis, klíčová slova, titulek dokumentu nebo kaskádové styly.

- **Párové značky < body > a </body>** – Tělo dokumentu – Vše uvnitř tohoto prvku je zobrazeno v hlavním okně prohlížeče.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Titulek stránky</title>
  </head>
  <body>
    <h1>Nadpis stránky</h1>
    <p>Toto je <a href="http://example.com/">odkaz</a> v odstavci.</p>
    <!-- toto je komentář -->
  </body>
</html>
```

Obrázek 8 – Základní struktura dokumentu HTML 5

Zdroj: <https://cs.wikipedia.org/wiki/Wikipedie>

Struktura dokumentu je velmi podobná struktuře v novinách. U každého příběhu je titulek, text a většinou i obrázky. Pokud je článek dlouhý, může být rozdělen do podnadpisů, které rozdělují příběh do samostatných sekcí. Struktura pomáhá čtenáři porozumět článku v novinách. V online je struktura téměř totožná.

Pro lepší přehled by se měla v každém dokumentu zachovat struktura použití nadpisů a podnadpisů, která zobrazuje hierarchii informací na každé stránce. Dokument může například začínat velkým nadpisem, po kterém následuje úvod nebo nejdůležitější informace. Další informace na stránce mohou být rozšířené o podnadpisy.

Stejně jako u psaní ve Wordu se i u webových aplikací rozdělují text do sekcí a odstavců kvůli přehlednosti a struktuře dokumentu. Každé téma může mít nový odstavec a každá sekce může mít nadpis popisující její obsah. [5]

3.2 Cascading Style Sheets (CSS)

CSS slouží k vytváření grafického vzhledu webových stránek. Umožňuje měnit velikost objektů, pozice, barvy, rozložení, písma nebo přidávat vizuální efekty. CSS rovněž umožňuje přizpůsobit prezentaci různým typům zařízení tzv. responzivní web. Web lze pomocí CSS přizpůsobit na velké obrazovky (desktohy, střední obrazovky (tablety) a malé obrazovky (mobily). CSS je nezávislé na HTML a lze jej použít s jakýmkoliv značkovacím jazykem založeným na XML. CSS lze vložit do samostatného souboru s koncovkou .css nebo přímo do

hlavičky HTML dokumentu ohraničené tagem `<style></style>`. Oddělení HTML od CSS usnadňuje údržbu webů, lepší flexibilitu, sdílení šablon stylů napříč stránkami a přizpůsobení stránek různým prostředí. Pro CSS kódy je lepší používat samostatné soubory. Lze tak rozdělit obsah od prezentace (grafiky) webu. [5]

```
1 body {
2   color: #565656;
3   background-color: #f2f2f2
4 }
5 a {
6   color: #2196F3
7 }
```

Obrázek 9 – Ukázka CSS

Zdroj: autor

3.2.1 Selektory

Stylizace jednotlivých prvků se vytváří pomocí tzv. selektorů. Selektory se v CSS používají k definování prvků, které budou stylovány pomocí CSS. Existuje mnoho různých typů selektorů CSS, z nichž každý má svou vlastní jedinečnou syntaxi. Selektory sdělují prohlížeči, na které prvky má použít hodnoty vlastností CSS.

Selektory se dělí do pěti kategorií:

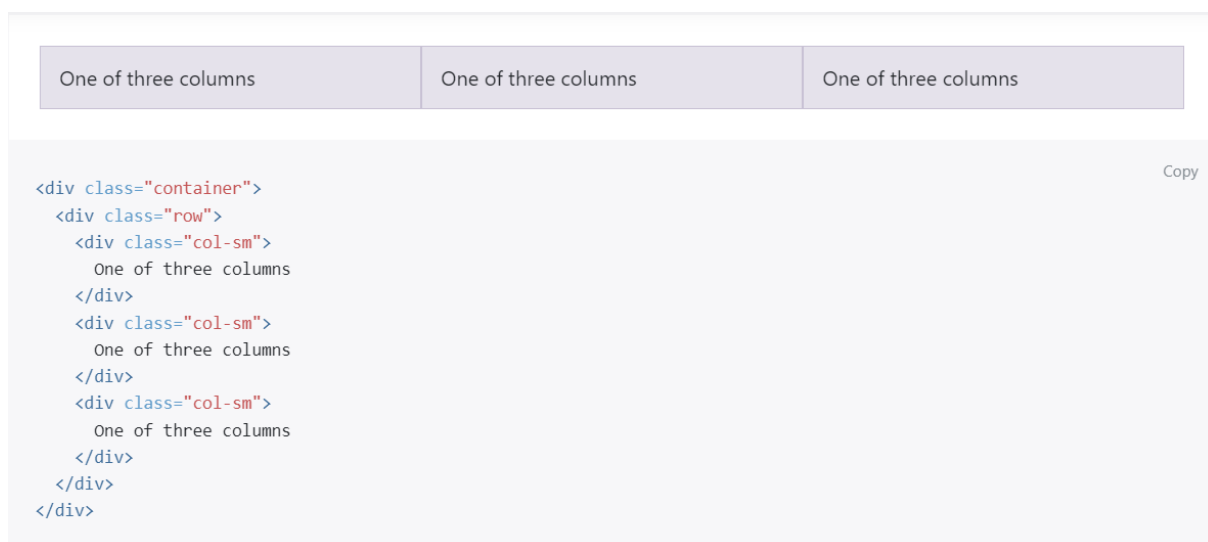
- Jednoduché selektory (výběr prvků na základě názvu, id, třídy)
 - #id, .class, div
- Selektory kombinátorů (výběr prvků na základě konkrétního vztahu mezi nimi). V CSS existují čtyři různé kombinátory:
 - selektor potomka (mezera)
 - child/dětský selektor (>)
 - sousedící selektor sourozenců (+)
 - obecný selektor sourozenců (~), vybere všechny prvky, které jsou dalšími sourozenci zadaného prvku.
- Selektor pseudo-tříd (výběr prvků na základě určitého stavu)
 - :focus, :link, :hover, :active, :visited
- Selektory pseudo-prvků (výběr a styling části prvku)
 - ::first-letter, ::first-line, ::after, ::before
- Selektor atributů (výběr prvků na základě atributu nebo hodnoty atributu, nejčastěji u inputu nebo odkazu)
 - [attribute="value"], a[target="_blank"], input[type="text"]

Všechny možnosti selektorů lze najít na stránce:
https://www.w3schools.com/cssref/css_selectors.php

3.3 Bootstrap

Bootstrap je open-source framework pro vývoj responzivních webových stránek a aplikací. Byl vyvinut týmem Twitteru a poprvé uveden v roce 2011. Bootstrap se stává stále více populárním mezi vývojáři kvůli svému snadnému použití, rozšiřitelnosti a možnosti rychle vytvářet moderní webové stránky a aplikace.

Bootstrap používá kombinaci HTML, CSS a JavaScript k vytvoření webových stránek. Jeho základním prvkem je takzvaný grid systém, který umožňuje rozdělení stránky do sloupců a řádků. Tento grid systém je flexibilní a umožňuje vytvořit stránky, které se přizpůsobí různým velikostem obrazovky. Bootstrap obsahuje také velké množství předpřipravených komponent a stylů, jako jsou například tlačítka, formuláře, modální okna a navigační panely. Tyto komponenty lze snadno přizpůsobit potřebám dané webové stránky pomocí CSS tříd a JavaScriptových funkcí. Bootstrap má také rozsáhlou dokumentaci a komunitu, takže vývojáři mohou snadno najít mnoho návodů a ukázek na internetu. Bootstrap také poskytuje podporu pro mnoho moderních prohlížečů a je optimalizován pro rychlé načítání stránek. Bootstrap rovněž podporuje rozšiřitelnost pomocí pluginů, které umožňují přidání dalších funkcí, jako jsou například kalendáře, grafy a animace. Tyto pluginy jsou většinou vyvinuty třetími stranami a jsou často zdarma ke stažení a použití.



Obrázek 10 – Bootstrap

Zdroj: <https://getbootstrap.com/docs/4.0/layout/grid/>

Vzhledem k tomu, že Bootstrap je open-source a zdarma ke stažení, stává se stále více populárním mezi vývojáři, kteří chtějí vytvářet snadno a rychle moderní responzivní webové stránky a aplikace. S jeho snadným použitím a rozšiřitelností se stává ideálním nástrojem pro vývoj moderních webových stránek a aplikací. [9]

3.4 JavaScript

JavaScript se poprvé objevil v Netscape 2.0 v roce 1995 pod názvem LiveScript. Univerzální jádro jazyka bylo zabudováno do Netscape, Internet Exploreru a dalších webových prohlížečů.

JavaScript je odlehčený a nejčastěji používaný programovací jazyk, který weboví vývojáři běžně používají k vytváření dynamičtějších interakcí při vývoji webových stránek, aplikací, serverů nebo dokonce her. Umožňuje vytvářet komplexní pokročilé funkce, jejichž implementace umožňuje skriptu na straně klienta komunikovat s uživatelem a vytvářet tak dynamické stránky. JavaScript vytváří dynamicky aktualizovaný obsah webu, jako je například interaktivní mapa, animovaná 2D/3D grafika nebo rolovací video. Vývojáři obecně používají JavaScript společně s HTML a CSS. Skriptovací jazyk dobře spolupracuje s CSS při formátování prvků HTML. JavaScript však stále udržuje interakci s uživatelem, což CSS samo o sobě nedokáže.

JavaScript je nejběžnější formou jazyka na straně klienta. Aby byl kód interpretován prohlížečem, měl by být Skript součástí dokumentu HTML nebo na něj odkazovat. To znamená, že webová stránka nemusí být statický HTML, ale může obsahovat programy, které komunikují s uživatelem, ovládají prohlížeč a dynamicky vytvářejí obsah HTML. Mechanismus JavaScript na straně klienta poskytuje mnoho výhod oproti tradičním skriptům CGI na straně serveru. JavaScript lze například použít ke kontrole, zda uživatel zadal platnou e-mailovou adresu do pole formuláře. Kód JavaScript se spustí, když uživatel odešle formulář, a pouze pokud jsou všechny položky platné, budou odeslány na webový server. JavaScript lze použít k zachycení událostí iniciovaných uživatelem, jako jsou kliknutí na tlačítka, najetí na odkazy a další akce, které uživatel explicitně nebo implicitně spouští. [7]

3.4.1 Výhody JavaScriptu

- Menší interakce se serverem: Před odesláním stránky na server lze ověřit uživatelský vstup. To šetří provoz serveru, což znamená menší zatížení serveru.

- Okamžitá zpětná vazba pro návštěvníky: Nemusí čekat na opětovné načtení stránky, aby zjistili, zda něco nezapomněli zadat.
- Zvýšená interaktivita: Lze vytvářet rozhraní, která reagují, když na ně uživatel najede myší nebo je aktivuje pomocí klávesnice.
- Bohatší rozhraní: JavaScriptu umožňuje zahrnout i položky, jako je například drag-and-drop.

3.4.2 Omezení JavaScriptu

S JavaScriptem nelze zacházet jako s plnohodnotným programovacím jazykem. Postrádá následující důležité funkce:

- JavaScript na straně klienta neumožňuje čtení nebo zápis souborů.
- JavaScript nelze použít pro síťové aplikace, protože taková podpora není k dispozici.
- JavaScript nemá žádné vícevláknové nebo víceprocesorové funkce.

JavaScript je pouze lehký interpretovaný programovací jazyk, který umožňuje zabudovat interaktivitu do statických HTML stránek, takže umožňuje tvořit dynamické stránky.

3.5 PHP (Hypertext Preprocessor)

PHP je široce používaný open source univerzální skriptovací jazyk, který je vhodný pro vývoj webových aplikací. Lze jej vložit do HTML a umožňuje webovým vývojářům vytvářet dynamický obsah, který spolupracuje s databázemi. PHP se v podstatě používá pro vývoj webových softwarových aplikací. PHP se zaměřuje hlavně na skriptování na straně serveru, takže lze dělat téměř cokoli, jako je například shromažďování dat z formulářů, generování dynamického obsahu stránky nebo odesílání a přijímání souborů cookies. Mezi další výhody používání PHP patří rychlost vývoje, snadná integrace s různými databázovými systémy a velké množství dostupných knihoven a frameworků, které usnadňují vývoj.

Skripty zpracovává PHP interpret, který následně generuje HTML, XML nebo jiný výstup, který je odeslán do webového prohlížeče. Tento výstup se pak zobrazí uživateli. Kód se spouští na straně server, kde je zapotřebí nainstalovat prostředí podobné serveru, které umožňují programy jako XAMPP, což je distribuce Apache, nebo nginx.

PHP byl vytvořen v roce 1994 Rasmusem Lerdorfem, a od té doby se stal jedním z nejpoužívanějších jazyků pro tvorbu dynamických webových stránek. Podle Web Technology

Surveys, PHP je použito na 78.1% všech webů, včetně weby s vysokou návštěvností jako je Facebook nebo Wikipedia. [10], [11], [12]

3.5.1 Bezpečnost a ochrana PHP

PHP je rychlé, bezpečné a spolehlivé. I když je PHP bezpečné, je potřeba aplikaci chránit před škodlivým kódem PHP, úniku citlivých dat a dalšími bezpečnostními hrozbami. Zabezpečení webových stránek PHP vyžaduje vícevrstvý přístup. Tento přístup musí zahrnovat postupy bezpečného kódování, správné postupy ověřování, pravidelné testování a další zabezpečení.

3.5.2 Postupy bezpečného kódování

Nejdůležitější součástí zabezpečení webových stránek PHP jsou postupy bezpečného kódování. Vývojář by měl zabránit útokům SQL injection, používat šifrování k ochraně uživatelských dat, vyhnout se přenosu citlivých dat v prostém textu a používat zabezpečená oprávnění k souborům.

3.5.3 Postupy ověřování

Ověřování lze použít k řízení přístupu uživatelů na web. Toho lze dosáhnout pomocí autentizace pomocí hesla, dvoufaktorové autentizace nebo řízení přístupu na základě rolí.

3.5.4 Testování zabezpečení

Pravidelné testování zabezpečení je nezbytné pro zajištění bezpečnosti webu. To zahrnuje ruční i automatické testování pro kontrolu zranitelností, hledání škodlivého kódu a testování správných postupů ověřování.

3.5.5 Další bezpečnostní opatření

K zabezpečení webových stránek PHP lze také použít další bezpečnostní opatření. Patří mezi ně používání brány firewall webových aplikací, používání zabezpečeného hostování a implementace silných postupů zálohování dat. [13]

3.6 MySQL

MySQL je open source relační databázový systém, který se často používá pro ukládání a zpracování dat v různých webových aplikacích. Byl vytvořen švédským vývojářem Michaelem Wideniusem v roce 1995 a dnes je součástí produktového portfolia společnosti Oracle.

MySQL podporuje mnoho funkcí, jako jsou například transakce, trigger, pohledy a mnoho dalšího. Databáze MySQL mohou být použity pro ukládání a správu různých typů dat, jako jsou například textová data, obrázky, zvuky, videa a další.

MySQL podporuje mnoho jazyků. Pro přístup k datům v databázi MySQL se používají různé programovací jazyky a technologie, jako je PHP, Java, Python, .NET a další. Existuje také mnoho nástrojů a knihoven pro práci s databází MySQL, které usnadňují vývoj a správu aplikací.

Pro správu MySQL databáze se používají různé nástroje, jako jsou například MySQL Workbench, phpMyAdmin nebo Navicat. Tyto nástroje umožňují správu databáze, vytváření tabulek, přidávání dat, zálohování a obnovování dat, kontrolu přístupových práv.

MySQL je zdarma a open source, což znamená, že je k dispozici pro každého bezplatně a zdrojový kód je k dispozici ke stažení i úpravám. MySQL je také multiplatformní, takže běží na různých operačních systémech, jako Linux, Windows nebo macOS. [10]

Mezi další výhody MySQL patří:

- Vysoká rychlost a výkon.
- Snadná škálovatelnost a možnost rozšíření.
- Podpora transakcí a zálohování dat.
- Velká komunita uživatelů a dostupnost dokumentace.

3.7 phpMyAdmin

PhpMyAdmin je open-source webová aplikace, která slouží jako správce MySQL databází. Tato aplikace umožňuje uživatelům snadno vytvářet, upravovat a mazat databázové struktury nebo data pomocí přátelského webového rozhraní. PhpMyAdmin byl poprvé vydán v roce 1998 a od té doby se stal velmi populárním nástrojem mezi vývojáři a správci MySQL databází.

PhpMyAdmin je napsán v jazyce PHP a je distribuován pod licencí GNU General Public License. Aplikace umožňuje uživatelům spravovat různé aspekty MySQL databází, jako jsou

například uživatelé a jejich oprávnění, indexy a zálohy. PhpMyAdmin také nabízí možnosti importu a exportu dat a podporuje vytváření dotazů pomocí grafického uživatelského rozhraní.

PhpMyAdmin je k dispozici zdarma ke stažení z oficiálních stránek projektu nebo přes některé distribuce Linuxu, jako je například Debian nebo Ubuntu. [14]

3.8 WordPress

WordPress je open-source webový redakční systém pro správu obsahu (CMS), který umožňuje uživatelům snadno vytvářet a spravovat webové stránky. Tento systém byl poprvé vydán v roce 2003 a od té doby se stal jedním z nejpoblárnějších CMS na světě. Je k dispozici zdarma a v současné době se odhaduje, že WordPress pohání více než 40 % webových stránek na internetu.

WordPress je napsán v jazyce PHP a využívá relační databázi MySQL pro ukládání dat. Systém je distribuován pod licencí GNU General Public License a je k dispozici zdarma ke stažení a použití. Hlavní výhodou WordPressu je jeho jednoduchost a přizpůsobivost. Uživatelé mohou snadno vytvářet nové stránky, přidávat obsah a upravovat vzhled pomocí tisíce volně dostupných šablon a pluginů. Tyto pluginy umožňují rozšířit funkčnost WordPressu a přizpůsobit ho potřebám konkrétního projektu. Každou šablonu a plugin lze měnit a upravovat pomocí kódu. Vývojářům umožňuje urychlit a zefektivnit jejich práci.

Další výhodou WordPressu je jeho aktivní komunita vývojářů a uživatelů, kteří se pravidelně podílejí na vývoji a vylepšování systému. Komunita také nabízí spoustu dokumentace a podpory pro uživatele, kteří se s WordPressem setkávají poprvé. Vývojáři WordPressu také dbají na bezpečnost systému a pravidelně vydávají aktualizace, které řeší bezpečnostní chyby a zlepšují výkon systému. Nicméně, jako u každého webového systému, je důležité zajistit zabezpečení WordPressu proti neoprávněnému přístupu a útokům hackerských skupin. Výhodou WordPressu je také jeho schopnost optimalizovat webové stránky pro vyhledávače (SEO). Systém obsahuje výchozí SEO funkce, ale existuje také mnoho pluginů, které umožňují pokročilejší SEO nastavení.

WordPress je oblíbeným nástrojem pro vytváření blogů. Jeho výchozí funkce pro správu obsahu jsou optimalizovány pro blogování a mnoho šablon je navrženo s ohledem na blogování. Blogování na WordPressu je snadné díky jeho výchozím funkcím pro formátování textu, přidávání obrázků, videí a správu komentářů. Existuje mnoho příkladů webových stránek, které jsou postaveny na WordPressu. Mezi nejznámější patří například Forbes,

Microsoft News Center nebo Sony Music. Tyto stránky potvrzují výkonnost a flexibilitu WordPressu jako kvalitního CMS.

Při používání WordPressu je důležité dodržovat základní pravidla bezpečnosti a správy webové stránky. Mezi doporučené postupy patří pravidelné zálohování stránky, používání silných hesel, aktualizace systému a pluginů a instalace bezpečnostních pluginů.

Vzhledem k jeho výkonnosti a flexibilitě, se WordPress stal jedním z nejoblíbenějších CMS na světě a jeho popularita nadále roste. Jeho snadná použitelnost a množství dostupných pluginů i šablon umožňují uživatelům vytvářet a spravovat webové stránky bez nutnosti hlubších znalostí programování. [15], [16], [17]

3.9 API

API, neboli Application Programming Interface, je dnes v informatice velmi důležitý koncept. API umožňuje různým aplikacím a systémům vzájemnou komunikaci i výměnu dat, což zvyšuje efektivitu a flexibilitu celého systému.

API lze definovat jako rozhraní, které umožňuje komunikaci mezi různými aplikacemi nebo systémy. Jde o sadu funkcí, které jsou dostupné pro využití ostatními aplikacemi, aniž by musely znát detaily implementace. API tak umožňuje jednoduchou a efektivní integraci různých aplikací a systémů.

Existují různé typy API, například webové API, které umožňuje vzdálenou komunikaci pomocí HTTP protokolu, nebo desktopové API, které se používají při vývoji desktopových aplikací.

Webové API jsou dostupné pomocí URL adresy, která vrací data ve formátu JSON, XML nebo jiných formátech. Tyto data mohou být poté zpracována a zobrazena v různých aplikacích, jako jsou například mobilní aplikace, webové stránky nebo desktopové aplikace.

Mezi hlavní výhody použití API patří zlepšení efektivity a snížení nákladů na vývoj. Díky použití API je možné využít již existující funkce a služby, což umožňuje rychlejší vývoj nových aplikací. Další výhodou je snížení rizika chyb, protože API zajišťuje správnou formu a validaci dat.

API se také využívá k propojení různých systémů, což umožňuje snadnou integraci a zlepšuje komunikaci mezi systémy. Například webové API mohou být použity pro integraci různých služeb a aplikací, jako jsou například platební brány, dopravní systémy nebo sociální sítě.

V současné době existuje mnoho populárních webových API, jako jsou například Google Maps API, Twitter API nebo Facebook API. Tyto API poskytují různé funkce, jako jsou například zobrazení map nebo odesílání tweetů. Další výhodou API je možnost propojování různých aplikací a služeb, což zajišťuje integraci systémů. Díky tomu mohou aplikace a webové stránky snadno komunikovat s různými datovými zdroji, například s databázemi, CRM systémy nebo sociálními sítěmi. To umožňuje snadné sdílení dat a zlepšení funkcionality aplikací. API také umožňuje vývojářům vytvářet aplikace pro různé platformy a zařízení, což zvyšuje flexibilitu a škálovatelnost aplikací.

Příkladem může být ukázka vytvoření vlastní komunikace pomocí REST API, které získává data z realitního systému RealMan. Data jsou následně zpracována a ukládána na webové stránce www.qbrealty.cz pro realitní firmu. Níže je ukázka vytvoření vlastního endpointu, na který jsou nahrávána data ze systému RealMan a poté následně zpracována a uložena na web realitní firmy.

```
add_action('rest_api_init', function () {  
  
    register_rest_route('import/v1', '/estate/(?P<id>\d+)', array(  
        'methods' => 'POST, DELETE',  
        'callback' => 'import_estate_data',  
        'permission_callback' => 'check_apikey'  
    ));  
  
});
```

Obrázek 11 – Vytvoření vlastního REST API Endpointu ve WordPress

Zdroj: autor

API se také často využívá v oblasti digitálního marketingu. Například poskytovatelé reklamních sítí využívají API ke sdílení dat s reklamními partnery a umožňují tak cílení reklamních kampaní na konkrétní skupiny uživatelů. API také umožňuje automatizaci marketingových procesů a zlepšení efektivity marketingových kampaní.

API představuje významný nástroj pro vývojáře, kteří chtějí vytvářet aplikace a služby, které jsou snadno přístupné a integrovatelné. API zvyšuje efektivitu vývoje a umožňuje rychlou integraci různých systémů. Je důležité, aby vývojáři měli dostatečné znalosti o API a jejich správném využívání. [18]

3.10 Social Media Integration

Integrace sociálních médií s webovými stránkami znamená proces začlenění platform sociálních médií, jako jsou Facebook, Twitter, Instagram, LinkedIn a další, do webových stránek. Tato integrace umožňuje návštěvníkům webových stránek sdílet obsah webových stránek nebo s nimi komunikovat prostřednictvím svých účtů na sociálních sítích. Integrace sociálních médií se pro firmy stává základním nástrojem pro rozšíření jejich dosahu a zapojení cílové skupiny.

Integrace sociálních médií s webovými stránkami nabízí několik výhod, včetně většího zviditelnění značky, lepší optimalizace pro vyhledávače a lepšího zapojení zákazníků. Začleněním tlačítek pro sdílení sociálních médií na webové stránky mohou uživatelé snadno sdílet obsah webových stránek se svými přáteli, což může vést ke zvýšenému zviditelnění značky a návštěvnosti webových stránek.

Integrace sociálních médií také zlepšuje optimalizaci pro vyhledávače (SEO), protože platformy sociálních médií jsou považovány za rozhodující faktor při určování pořadí webových stránek. Signály ze sociálních sítí, jako jsou lajky, sdílení a komentáře, mohou zlepšit viditelnost webových stránek ve vyhledávačích, což v konečném důsledku vede ke zvýšení návštěvnosti a lepšímu umístění ve vyhledávačích.

Integrace sociálních médií navíc umožňuje podnikům navázat kontakt se zákazníky přímo prostřednictvím platform sociálních médií. Vložení kanálů sociálních médií na své webové stránky mohou podniky prezentovat svou přítomnost v sociálních médiích a zapojit své publikum prostřednictvím lajků, komentářů a sdílení. Toto zapojení může vést ke zvýšení loajality zákazníků a v konečném důsledku ke zvýšení prodeje.

Existuje několik způsobů, jak integrovat sociální média do webových stránek, včetně použití tlačítek pro sdílení sociálních médií, kanálů sociálních médií, přihlašování do sociálních sítí a komentování na sociálních sítích. Tlačítka pro sdílení na sociálních sítích umožňují uživatelům sdílet obsah přímo na jejich platformách sociálních médií jediným kliknutím. Kanály sociálních médií umožňují podnikům prezentovat svou přítomnost v sociálních médiích na svých webových stránkách, zatímco sociální přihlášení umožňuje uživatelům přihlásit se na webové stránky pomocí přihlašovacích údajů ze sociálních médií, což zjednodušuje proces registrace. Sociální komentáře umožňují uživatelům komentovat obsah webových stránek pomocí jejich profilů v sociálních médiích, což může vést k většímu zapojení a konverzaci na webových stránkách.

Integrace sociálních médií s webovými stránkami je zásadní pro rozšíření dosahu a zapojení cílové skupiny podniku. Nabízí několik výhod, včetně většího zviditelnění značky, lepší optimalizace pro vyhledávače a lepšího zapojení zákazníků. Začleněním sociálních médií do webových stránek mohou podniky zlepšit svou online přítomnost a v případě e-shopu zvýšit i svůj prodej. [19]

4 NÁVRH WEBOVÉ APLIKACE

Webová aplikace pro obce a města je navržena tak, aby zlepšila komunikaci mezi obcí a jejich obyvateli a k tomu zvýšila zapojení občanů do místních záležitostí. Webová aplikace umožňuje obyvatelům snadný a rychlý přístup k informacím o dění v obci, kde mají možnost zapojit se do diskuzí a přispět svými názory.

Hlavní funkcionalitou aplikace je poskytování informací o plánovaných a probíhajících akcích, zasedáních zastupitelstva, přístup k online úřední desce, rezervační systémy (například na rezervaci Sálu nebo Sportovní haly a poskytování dalších důležitých informací a událostí v obci. Aplikace umožňuje sdílení fotografií i videí z místních akcí nebo událostí a umožňuje obyvatelům komentovat a hodnotit obsah. Další funkcí je možnost zasílání oznámení a upozornění, jako například na plánované výluky dopravy, havárie či jiné mimořádné události. Občané mohou podávat své návrhy a stížnosti přímo prostřednictvím aplikace pomocí kontaktních formulářů nebo emailů.

Aplikace je navržena s ohledem na jednoduché ovládání pro uživatele s různými úrovněmi digitální gramotnosti. Aby se aplikace stala pro obyvatele atraktivní, je přizpůsobena potřebám a preferencím místní komunity s moderním a přehledným designem.

Webová aplikace pro obce a města přináší řadu výhod, jako je zvýšení zapojení občanů do místních záležitostí, zlepšení transparentnosti rozhodovacích procesů, poskytování aktuálních informací a umožnění efektivní komunikace mezi obcí a jejich obyvateli.

4.1 Požadavky na aplikaci

Nařízení České republiky stanovuje požadavky na webové aplikace pro obce a města s cílem zlepšit transparentnost, efektivitu a kvalitu poskytovaných služeb pro obyvatele. Některé z požadavků jsou:

- Zabezpečení ochrany osobních údajů obyvatel a soulad s legislativou EU GDPR.
- Zpřístupnění informací o veřejných zakázkách, smlouvách a jejich plnění, aby byly přístupné pro širokou veřejnost.
- Zpřístupnění a vedení online úřední desky.
- Poskytování informací o rozpočtu obce a města a jeho plnění, včetně detailního rozpisu příjmů a výdajů.
- Zpřístupnění informací o zasedáních zastupitelstva a výborů, včetně programu, usnesení a zápisů.

- Zajištění možnosti podávat online žádosti o vydání dokladů, jako například občanských průkazů nebo povolení k provozování podnikání.
- Zajištění možnosti podání elektronických podnětů, návrhů a stížností a jejich následné řešení.
- Zabezpečení přístupnosti webové aplikace pro osoby se zdravotním postižením a zohlednění principů univerzálního návrhu.
- Rezervační systém Sportovní haly a Obecního domu.
- Propojení s Munipolis.
- Propojení s Facebook a Instagramem pomocí Social Media Integration.

4.2 Databáze

WordPress pro ukládání obsahu, načítání obsahu, nastavení a konfiguraci webových stránek používá relační databázi. Ve výchozím nastavení používá databázi MySQL. Podporuje ale i jiné databázové systémy, například MariaDB a PostgreSQL.

Jádrem databáze WordPress je několik klíčových tabulek, včetně `wp_posts`, `wp_comments`, `wp_users`, `wp_terms` a `wp_options`, které slouží k ukládání různých typů obsahu a konfiguračních dat. WordPress používá k manipulaci a načítání dat z těchto tabulek dotazy SQL. Obsahuje také systém ORM (Object-Relational Mapping) s názvem `WP_Query`, který vývojářům umožňuje snadno načítat z databáze konkrétní sady dat na základě různých kritérií. Databáze WordPress je klíčovou součástí tohoto CMS systému. Přestože práce s databází WordPress může být někdy náročná, jedná se o kritickou součást systému CMS, která je nezbytná pro efektivní správu a škálování webů WordPress. Pochopení struktury a fungování databáze WordPressu je proto důležitou dovedností pro každého vývojáře nebo majitele webu, který s touto platformou pracuje. [15], [16], [17]

4.2.1 Tabulky v databázi WordPress

Jak již bylo zmíněno, databáze WordPress obsahuje několik klíčových tabulek, které slouží k ukládání různých typů obsahu a konfiguračních dat. Nejznámější a zároveň nejdůležitější tabulky jsou.

- **`wp_posts`:** Tato tabulka obsahuje všechny příspěvky, stránky a vlastní typy příspěvků vytvořené na webu. Každý řádek tabulky představuje jeden příspěvek a obsahuje údaje, jako je název příspěvku, obsah, autor, datum vytvoření a různá pole metadat. Tato

tabulka je ústředním prvkem systému WordPress CMS a je hojně využívána tématy a moduly plug-in k zobrazování a manipulaci s obsahem.

- **wp_comments:** v této tabulce jsou uloženy všechny komentáře na webu. Každý řádek tabulky představuje jeden komentář a obsahuje údaje, jako je autor komentáře, e-mail, webová stránka, datum vytvoření a samotný text komentáře. Tuto tabulku používá jádro systému WordPress ke správě komentářů a lze ji také rozšířit pomocí zásuvných modulů a přidat do ní další funkce.
- **wp_users:** Tato tabulka obsahuje informace o všech registrovaných uživateli na webu. Každý řádek tabulky představuje jednoho uživatele a obsahuje údaje, jako je uživatelské jméno, e-mail, heslo (uložené v zaheslované podobě) a různá pole metadat, například zobrazované jméno uživatele a jeho role na webu. Tato tabulka je hojně využívána jádrem systému WordPress a zásuvnými moduly ke správě uživatelských účtů a oprávnění.
- **wp_terms:** v této tabulce jsou uloženy termíny taxonomie používané ke kategorizaci příspěvků, stránek a dalšího obsahu. Každý řádek tabulky představuje jeden termín a obsahuje údaje, jako je název termínu, slug a různá pole metadat. Tato tabulka je hojně využívána jádrem systému WordPress a zásuvnými moduly pro správu taxonomií a kategorií.
- **wp_options:** Tato tabulka obsahuje konfigurační nastavení webu a další možnosti. Každý řádek tabulky představuje jednu možnost a obsahuje údaje, jako je název možnosti, hodnota a různá pole metadat. Tuto tabulku používá jádro systému WordPress a zásuvné moduly k ukládání různých konfiguračních nastavení, například názvu webu, struktury odkazů a různých dalších možností.

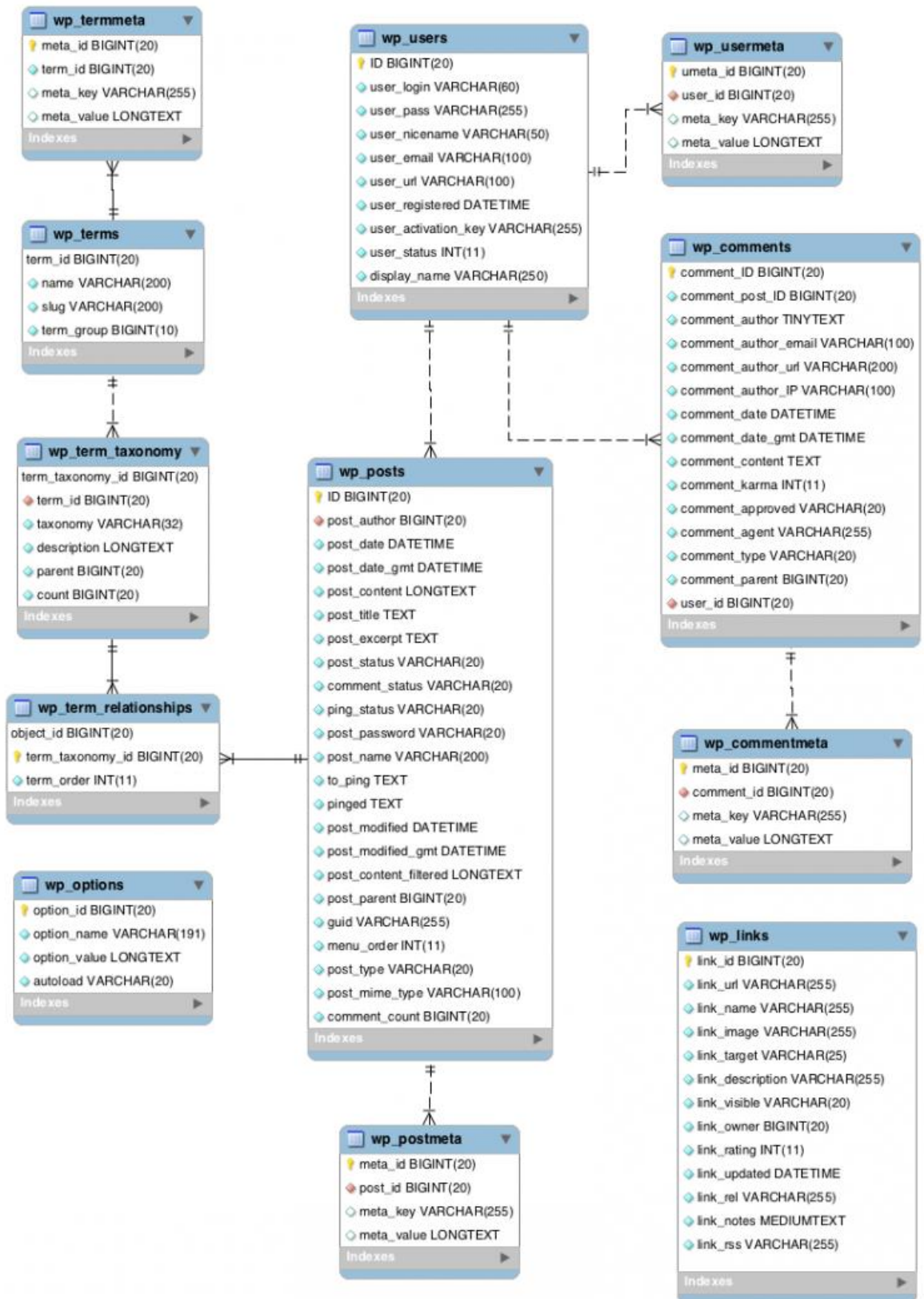
Pochopení struktury a vztahů mezi těmito tabulkami je pro efektivní práci s databází WordPress klíčové. Zásuvné moduly a témata se při ukládání a načítání dat často spoléhají na tyto tabulky a optimalizace databázových dotazů a indexování může mít významný vliv na výkon a škálovatelnost webu WordPress. Další tabulky v databázi WordPress například jsou:

- **wp_postmeta:** v této tabulce jsou uložena další metadata pro každý příspěvek v tabulce wp_posts. Každý řádek tabulky představuje jednu položku metadat a obsahuje údaje, jako je ID příspěvku, klíč metadat a hodnota metadat. Tuto tabulku hojně využívají zásuvné moduly a témata k ukládání vlastních dat příspěvků, například vlastních polí, a k rozšíření funkčnosti tabulky wp_posts.

- **wp_termmeta:** Tato tabulka je podobná tabulce wp_postmeta, ale ukládá metadata pro termíny taxonomie v tabulce wp_terms. Každý řádek tabulky představuje jednu položku metadat a obsahuje údaje, jako je ID termínu, klíč metadat a hodnota metadat. Tuto tabulku používají zásuvné moduly a motivy k ukládání vlastních dat taxonomie.
- **wp_term_taxonomy:** Tato tabulka uchovává informace o taxonomiích používaných na webových stránkách, jako jsou kategorie a značky. Každý řádek tabulky představuje jednu taxonomii a obsahuje údaje, jako je ID taxonomie, název taxonomie a přípona taxonomie. Tuto tabulku používá jádro systému WordPress a zásuvné moduly ke správě taxonomií a jejich vztahů s příspěvky a dalším obsahem.
- **wp_links:** Tato tabulka uchovává informace o odkazech (známých také jako záložky) přidáných na webovou stránku. Každý řádek tabulky představuje jeden odkaz a obsahuje údaje, jako je adresa URL odkazu, název odkazu a různá pole metadat. Tuto tabulku používá jádro systému WordPress a zásuvné moduly pro správu odkazů a jejich zobrazení na webu.
- **wp_usermeta:** Tato tabulka je podobná tabulce wp_postmeta, ale ukládá metadata pro uživatelské účty v tabulce wp_users. Každý řádek tabulky představuje jednu položku metadat a obsahuje údaje, jako je ID uživatele, klíč metadat a hodnota metadat. Tuto tabulku používají zásuvné moduly a motivy k ukládání vlastních údajů o uživateli.
- **wp_commentsmeta:** Tato tabulka je podobná tabulce wp_postmeta, ale ukládá metadata pro komentáře v tabulce wp_comments. Každý řádek tabulky představuje jednu položku metadat a obsahuje údaje, jako je ID komentáře, klíč metadat a hodnota metadat. Tuto tabulku používají zásuvné moduly a témata k ukládání vlastních dat komentářů.

Toto je jen několik klíčových tabulek v databázi WordPress. Existuje mnoho dalších v závislosti na konkrétní konfiguraci a zásuvných modulech používaných na daném webu. Pochopení vztahů mezi těmito tabulkami a optimalizace dotazů a indexování může mít významný vliv na výkon a škálovatelnost webu WordPress. [15], [16], [17]

4.2.2 ER Diagram databáze WordPressu



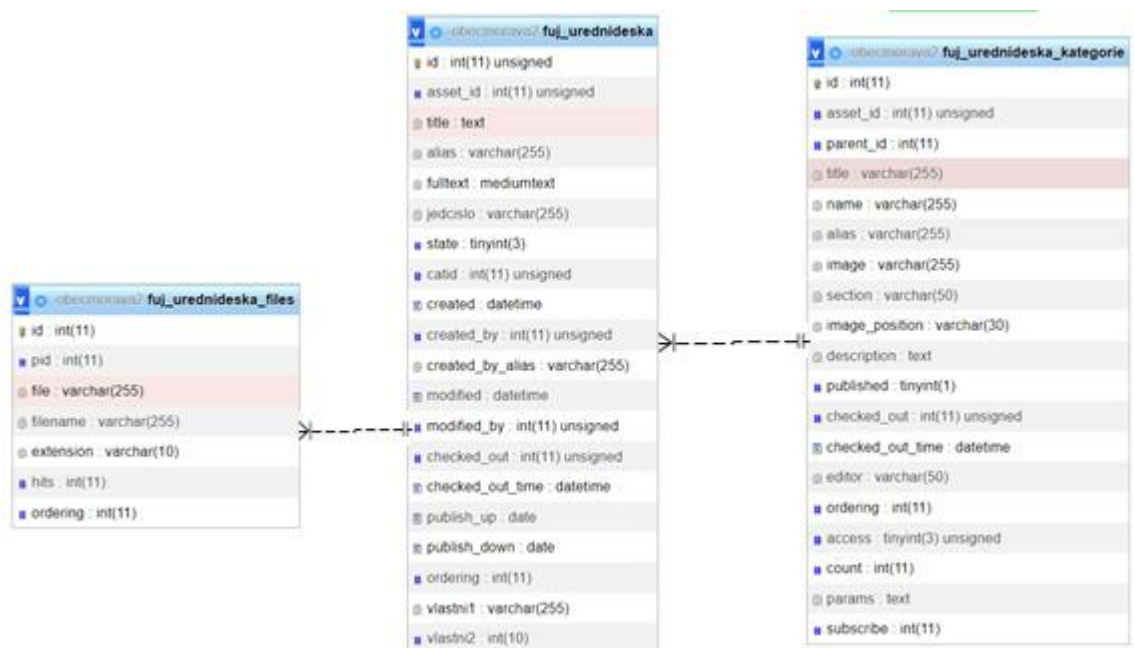
Obrázek 12 – ER Diagram WordPress

Zdroj: https://codex.WordPress.org/Database_Description

4.3 Vlastní struktura tabulek pro Úřední desku

Databáze pro úřední desku obsahuje 3 tabulky, které slouží ke správné funkčnosti úřední desky.

- **fuj_urednideska:** Tato tabulka obsahuje všechny záznamy úřední desky. Každý řádek tabulky představuje jeden specifický záznam a obsahuje údaje o daném záznamu, jako je id, název, kategorie, datum vyvěšení, datum sejmutí nebo počet a další pole pro případný rozvoj. Tato tabulka je ústředním prvkem funkčnosti úřední desky.
- **fuj_urednideska_files:** Tato tabulka obsahuje všechny záznamy o přidání souborů úřední desky. Každý řádek tabulky představuje jeden soubor a obsahuje údaje o daném souboru, jako je jeho id, id záznamu, ke kterému patří, název souboru, typ koncovky, pořadí a další pole pro případný rozvoj.
- **fuj_urednideska_kategorie:** Tato tabulka obsahuje všechny kategorie úřední desky. Každý řádek tabulky představuje jednu kategorii a obsahuje údaje o dané kategorii, jako je její id a název a další pole pro případný rozvoj.



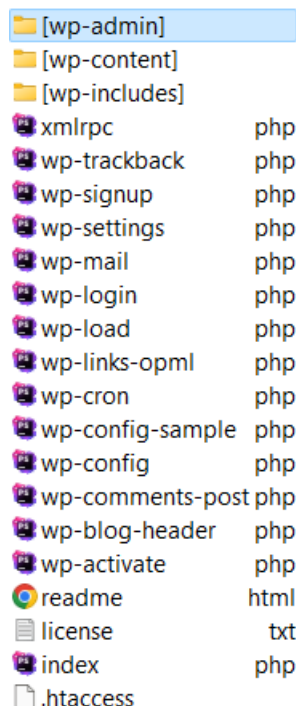
Obrázek 13 – ER Diagram Úřední desky

Zdroj: autor

5 REALIZACE WEBOVÉ APLIKACE

5.1 Struktura dat

Datová struktura WordPressu obsahuje několik jedinečných adresářů a souborů. Při připojení na FTP webové stránky se zobrazí tři základní adresáře, do kterých WordPress ukládá různá data v závislosti na jejich použití.



Obrázek 14 – Výchozí nastavení nově nainstalované nainstalovaného WordPressu

Zdroj: autor

5.1.1 wp-admin

Složka wp-admin je jednou z nejdůležitějších složek v systému WordPress. Obsahuje soubory, které slouží k řízení a správě celého webu. Tato složka je určena pouze pro administrátory a uživatele s příslušnými oprávněními. Ve složce wp-admin se nachází několik podadresářů, v nichž jsou uloženy soubory pro jednotlivé funkce administrace. Například složka „includes“ obsahuje soubory, které řídí základní funkce WordPressu, jako je například správa uživatelů a přístupových práv. Další důležitou složkou je „css“, která obsahuje stylovací soubory pro vzhled administrace. Složka wp-admin obsahuje také soubor „index.php“, který je vstupním bodem pro administraci celého webu. Tento soubor obsahuje základní funkce a načítá další soubory, které jsou potřebné pro správu a řízení webu.

Bez správně fungující složky wp-admin by nebylo možné správně spravovat a řídit web v systému WordPress. Proto je velmi důležité, aby byla tato složka správně zabezpečena před neoprávněným přístupem a aby byla vždy aktuální verze WordPressu. Při správném použití a správě této složky může být správa a řízení webu v systému WordPress snadná a efektivní.[15] [16]

5.1.2 wp-content

Složka wp-content je jednou z dalších nejdůležitějších složek v systému WordPress. Obsahuje soubory, které určují vzhled a funkce webu. Tato složka obsahuje především všechny soubory spojené s obsahem webu, jako jsou například obrázky, videa, zvuky, ale také šablony, pluginy, databázové zálohy a další soubory. Složka wp-content obsahuje také soubory pro vytváření databázových záloh, které jsou důležité pro zabezpečení a zálohování dat webu. Správná správa složky wp-content je klíčová pro správné fungování a vzhled webu. Je důležité mít zabezpečenou tuto složku proti neoprávněnému přístupu, aby nedošlo k nežádoucímu poškození dat. Dále je důležité pravidelně zálohovat data a udržovat pluginy a šablony v aktuální verzi.[15] [16]

5.1.2.1 Šablony

Šablony jsou výchozím vzhledem celého webu a jsou umístěny v podadresáři „themes“. Každá šablona obsahuje soubory pro základní design webu, jako jsou styly, obrázky, JavaScripty, funkce pro zobrazování a zpracování obsahu a další. Šablony jsou složeny z řady souborů, které definují různé části stránky, jako jsou hlavičky, zápatí, menu, obsah, widgety a mnoho dalšího. Tyto soubory jsou organizovány do adresářů, které jsou součástí šablony. WordPress obsahuje několik výchozích šablon, ale uživatelé mohou také vytvářet své vlastní šablony. Šablony WordPressu jsou psány v jazyce PHP a používají speciální funkce a značky pro zobrazování dat. V šablonách jsou také používány šablony značek, které umožňují zobrazovat data z databáze WordPressu, jako jsou například titulky, kategorie, popisky a obsahy stránek. Šablony lze také přizpůsobit pomocí CSS kaskádových stylů, což umožňuje uživatelům upravovat vzhled webových stránek a přizpůsobit je svým potřebám. Uživatelé mohou také použít pluginy pro úpravy šablon a zvýšení flexibility a funkcionalit.

Výběr správné šablony je klíčovým faktorem tvorby kvalitní webové stránky. Šablony by měly být vybírány s ohledem na potřeby webu a jeho cílového publika. Šablony by měly být také optimalizovány pro rychlost a výkon, aby bylo zajištěno rychlé načítání stránek a snadná

navigace pro uživatele. Lze také vytvořit vlastní šablony WordPressu, což umožňuje vytvořit unikátní design a funkčnost webu přesně podle požadavků uživatele. Vytvoření vlastní šablony však vyžaduje pokročilé znalosti programování a vývojových nástrojů.[15] [16]

5.1.2.2 Pluginy

Pluginy jsou rozšíření funkčnosti webu a jsou uloženy v podadresáři „plugins“. Pluginy jsou samostatné softwarové moduly, které jsou navrženy tak, aby mohly být snadno instalovány, aktivovány a spravovány z administrátorského rozhraní WordPressu.

Pluginy umožňují přidávat nové funkce do systému WordPress, jako je například vkládání formulářů, zobrazování fotogalerií, integrace sociálních sítí, vylepšení SEO, zabezpečení webu, správa záloh a mnoho dalšího. Pluginy lze instalovat z oficiálního repozitáře pluginů WordPressu, nebo je lze zakoupit na specializovaných webových stránkách a následně nainstalovat manuálně. Pluginy mají obvykle svou vlastní konfigurační stránku, na které lze nastavit jejich parametry a přizpůsobit je potřebám webu. Některé pluginy také obsahují rozhraní pro tvorbu vlastních funkcí a formulářů, což umožňuje uživatelům vytvářet vlastní funkce a přizpůsobit tak pluginy svým potřebám.

Používání pluginů může mít vliv na rychlost a výkon webu. Příliš mnoho nainstalovaných pluginů může zpomalit načítání stránek a zvýšit nároky na server. Proto je důležité vybírat pouze ty pluginy, které jsou skutečně potřebné a optimalizovat je pro rychlost a výkon webu. Správná správa pluginů je důležitá pro zabezpečení webu. Pluginy by měly být vždy aktualizovány na nejnovější verzi, aby byly zajištěny jejich bezpečnostní aktualizace a minimalizováno riziko zranitelností.[15] [16]

5.1.3 wp-includes

Složka wp-includes je třetí klíčovou složkou v systému WordPress, která obsahuje základní funkce a knihovny, které jsou nezbytné pro běh celého webu. Tato složka obsahuje soubory, které jsou součástí jádra WordPressu a řídí jeho základní funkce. Mezi nejdůležitější soubory v této složce patří soubory, které řídí funkce pro správu uživatelů a přístupových práv, správu obsahu, databázi, formuláře, obrázky, šifrování a mnoho dalšího. Tyto soubory jsou důležité pro správné fungování a zabezpečení webu. Složka wp-includes obsahuje také mnoho knihoven a souborů, které jsou nutné pro správné fungování pluginů a šablon. Tyto soubory obsahují funkce pro vkládání a zpracování kódu, validaci vstupních dat, správu souborů a mnoho dalšího. Je důležité, aby byla složka wp-includes vždy aktuální verze WordPressu a aby byla

správně zabezpečena proti neoprávněnému přístupu. Správná správa této složky a pravidelná aktualizace zajišťují správné fungování webu a minimalizují riziko zranitelností a útoků na web.[15] [16]

5.1.4 Ostatní soubory

Výchozí instalace WordPressu obsahuje několik klíčových souborů, z nichž každý má svou specifickou roli. Jednotlivé soubory mohou být upraveny pro vlastní potřeby. Zde je seznam těchto klíčových souborů a jejich účelu:

- **index.php**: Hlavní soubor, který je spuštěn při načtení webu. Tento soubor se stará o zobrazení obsahu stránek, pokud nebyl specifikován konkrétní soubor.
- **license.txt**: Soubor obsahující licenční podmínky WordPressu.
- **readme.html**: Soubor obsahující informace o aktuální verzi WordPressu.
- **wp-activate.php**: Soubor používaný pro aktivaci uživatelských účtů, pokud byla povolena funkce registrace uživatelů na webu.
- **wp-blog-header.php**: Soubor obsahující základní kód pro načtení obsahu blogu a vytvoření hlavičky stránky.
- **wp-comments-post.php**: Soubor obsahující kód pro zpracování přidaných komentářů a jejich uložení do databáze.
- **wp-config-sample.php**: Soubor obsahující vzorový konfigurační soubor pro WordPress. Uživatelé musí tento soubor přejmenovat na wp-config.php a upravit v něm připojení k databázi.
- **wp-cron.php**: Soubor, který se stará o spuštění automatických úloh WordPressu, jako je například plánování příspěvků.
- **wp-links-opml.php**: Soubor obsahující seznam odkazů na webu ve formátu OPML.
- **wp-load.php**: Soubor, který zavádí WordPress a všechny jeho součásti.
- **wp-login.php**: Soubor obsahující kód pro přihlašování uživatelů na web.
- **wp-mail.php**: Soubor používaný pro zpracování e-mailů, které jsou zasílány přes WordPress.
- **wp-settings.php**: Soubor obsahující nastavení WordPressu, jako jsou například připojení k databázi nebo nastavení adres URL.
- **wp-signup.php**: Soubor používaný pro registraci nových uživatelů na webu.
- **wp-trackback.php**: Soubor používaný pro sledování zpětných odkazů a zpracování požadavků na zpětné odkazy.

- **xmlrpc.php**: Soubor používaný pro komunikaci s WordPressem pomocí XML-RPC protokolu.

5.2 Základní nastavení

Základní nastavení webové stránky na WordPress je hlavním krokem k tomu, aby stránka byla funkční a připravena pro veřejné zveřejnění.

Po instalaci WordPressu je nutné vyplnit základní informace o webové stránce, jako je název, popis, časové pásmo, jazyk a kontaktní informace. Tyto informace se objeví na stránce a budou poskytovat základní informace pro návštěvníky. Následujícím krokem je výběr a nastavení šablony pro webovou stránku. WordPress nabízí mnoho různých šablon, které se liší designem a funkcemi. Po výběru šablony lze nastavit další detaily jako je logo, barvy a typografii, aby šablona odpovídala daným požadavkům.

Menu na stránce je důležitou součástí navigace pro návštěvníky. Ve WordPressu lze vytvořit menu a umístit ho na určená místa na stránce. V menu lze zahrnout odkazy na stránky, kategorie, příspěvky a zkratka na cokoliv. Stránka Domů (Home) je obvykle hlavní stránkou webu a je zobrazena jako výchozí stránka po načtení webu. V této části lze nastavit vzhled a obsah této stránky.

Widgety jsou malé prvky, které mohou být umístěny v různých oblastech na stránce a poskytují další informace pro návštěvníky. Ve WordPressu lze nastavit různé widgety jako například vyhledávací pole, poslední příspěvky nebo oblíbené kategorie.

Optimalizace pro vyhledávače (SEO) je další nezbytnou součástí tvorby webových stránek. Ve WordPressu lze použít různé pluginy pro SEO, jako například Yoast SEO.

5.3 Úprava vybrané šablony dle potřebných požadavků

Výběr vhodné šablony je klíčový pro vytvoření profesionální a funkční webové stránky. Každopádně pokud se vytváří složitější webové aplikace, ve většině případů nebude šablona dostačující a bude ji potřeba upravit dle vlastních požadavků.

5.3.1 Vytvoření vlastních pluginů

Prvním krokem k vytvoření vlastního pluginu je připojení k FTP serveru webové aplikace. Po úspěšném připojení je nutné nejprve vytvořit vlastní složku s názvem vlastního pluginu

v adresáři **/wp-content/plugins/**. V této složce je zapotřebí vytvořit soubor s názvem vlastního pluginu s příponou `.php`. Tento soubor složí jako hlavní soubor vlastního pluginu. Do souboru je nutné vložit hlavičku, které bude popisovat základní informace o pluginu, jako je název, popis a verze. Pod hlavičkou se naprogramují jednotlivé funkce, které budou provádět požadovanou funkcionalitu pluginu. Při tvorbě funkcí lze použít i základní funkce WordPressu jako je například `add_action` nebo `add_filter` pro připojení funkcí k určitým událostem ve WordPressu. Při tvorbě funkcí je možné rovněž vytvořit instalační a deinstalační funkce pro plugin. Tyto funkce se spustí při aktivaci nebo deaktivaci pluginu a mohou být použity k vytvoření nebo odstranění databázových tabulek nebo jiných položek, které jsou potřebné pro plugin. Po zhotovení všech potřebných funkcí se soubory pluginu nahrají do složky v adresáři `/wp-content/plugins/` a plugin se aktivuje v administračním prostředí WordPressu.

Při tvorbě vlastních pluginů pro WordPress je důležité dodržovat zásady bezpečnosti a správně dokumentovat kód pro usnadnění údržby a rozšíření pluginu v budoucnu. Dále je vhodné dodržovat standardy WordPressu pro vývoj pluginů.

5.3.2 Vlastní kategorie a vlastní pole

Pro vytvoření vlastních kategorií příspěvků a vlastních polí v příspěvku je možné využít mnoho externích pluginů. Mezi nejznámější patří CPT UI nebo ACF. Vlastní kategorie příspěvků a vlastní pole lze rovněž vytvořit pomocí PHP.

5.3.3 Custom Posty Type (CPT)

K vytvoření CPT stačí zapisovat jednotlivé funkce do souboru `functions.php`. Pro lepší přehled lze vytvořit vlastní plugin, do kterého se budou zapisovat tyto jednotlivé funkce. V souboru `functions.php` nebo ve vlastním pluginu lze vytvořit nový CPT pomocí WordPress API funkcí. Nejprve se musí zaregistrovat nový typ obsahu pomocí funkce `register_post_type`. Tato funkce obsahuje několik argumentů, jako je název CPT, popis, ikona a další. Následující kód ukazuje základní příklad vytvoření CPT pro Upozornění:

```

function create_alert_post_type() {
    register_post_type( 'upozorneni',
        array(
            'labels' => array(
                'name' => __( 'Upozornění' ),
                'singular_name' => __( 'Upozornění' )
            ),
            'public' => true,
            'has_archive' => true,
            'rewrite' => array('slug' => 'upozorneni'),
        )
    );
}
add_action( 'init', 'create_alert_post_type' );

```

Obrázek 15 – Vlastní CPT typu Upozornění

Zdroj: autor

5.3.4 Advanced Custom Fields (ACF)

ACF jsou pluginy pro WordPress, které umožňují vývojářům snadno přidávat vlastní pole a vlastnosti na webové stránky. Vlastní ACF pole lze vytvořit i pomocí PHP. K vytvoření vlastních ACF polí lze využít API funkce WordPressu nebo ACF API funkce.

Jednotlivé funkce se stvoří buď v souboru functions.php nebo ve vlastním pluginu. V daném souboru lze vytvořit nové pole pomocí funkce `acf_add_local_field_group`. Tato funkce umožňuje přidávat vlastní skupiny polí pro vlastnosti.

Následující kód ukazuje příklad vytvoření vlastního pole pro nadpis Upozornění:


```

function my_acf_add_local_field_groups() {
    acf_add_local_field_group(array(
        'key' => 'skupina_1',
        'title' => 'Upozornění',
        'fields' => array(
            array(
                'key' => 'pole_1',
                'label' => 'Nadpis Upozornění',
                'name' => 'upozorneni_title', 'type' => 'text',
                'instructions' => 'Zadejte nadpis Upozornění',
                'required' => true,
                'wrapper' => array('width' => '', 'class' => '', 'id' => ''),
                'default_value' => '', 'placeholder' => '', 'prepend' => '', 'append' => '',
                'maxlength' => '', 'readonly' => false, 'disabled' => false,
            ),
        ),
        'location' => array(
            array(
                array('param' => 'post_type', 'operator' => '==', 'value' => 'upozorneni'),
            ),
        ),
        'menu_order' => 0, 'position' => 'normal', 'style' => 'default',
        'label_placement' => 'top', 'instruction_placement' => 'label',
        'hide_on_screen' => '', 'active' => true,
        'description' => '',
    ));
}
add_action('acf/init', 'my_acf_add_local_field_groups');

```

Obrázek 16 – Vytvoření vlastního pole pro nadpis Upozornění

Zdroj: autor

5.3.5 Úprava grafiky a responzivního designu

Vlastní úprava grafiky byla provedena pomocí CSS kódování. Použití vlastního CSS kódu umožnilo přizpůsobit vzhled a styl stránky k potřebám webové stránky pro obec Moravany.

K vytvoření responzivního designu je důležité využít vlastností CSS, jako je například max-width, která umožňuje určit maximální šířku prvku v závislosti na velikosti obrazovky zařízení. To znamená, že prvky na stránce se automaticky přizpůsobí velikosti obrazovky, na které jsou zobrazeny. Tímto způsobem lze vytvořit responzivní layout, který zobrazuje obsah na stránce v závislosti na velikosti obrazovky. Úprava grafiky pomocí CSS kódování a vytváření

responzivního designu vyžaduje značnou zkušenost s CSS a designem webových stránek.

```
/* -----  
   Velká zařízení s šířkou displeje od 992px do 1199px  
----- */  
@media ( max-width: 1199px ) {  
   /* Vložení CSS kódu pro velké displeje */  
}  
/* -----  
   Střední zařízení s šířkou displeje od 768px do 991px  
----- */  
@media ( max-width: 991px ) {  
   /* Vložení CSS kódu pro střední displeje */  
}  
/* -----  
   Malé zařízení s šířkou displeje od 481px do 767px  
----- */  
@media ( max-width: 767px ) {  
   /* Vložení CSS kódu pro malé displeje */  
}  
/* -----  
   Extra malé zařízení s šířkou displeje do 480px  
----- */  
@media ( max-width: 480px ) {  
   /* Vložení CSS kódu pro extra malé displeje */  
}
```

Obrázek 17 – Kód tvorby responzivního webu

Zdroj: autor

5.4 Segmentace webu

Segmentace webu je klíčovým prvkem vytváření efektivní webové stránky. Ve webové aplikaci bylo využito pět základních segmentů – aktuality, dokumenty, galerie, upozornění a události. Každý z těchto segmentů má svůj specifický účel a své vlastní podkategorie. Zajišťuje tak přehlednost ve velkém množství dat, se kterým pracuje každé město nebo obec.

- **Aktuality** jsou pravděpodobně nejčastějším segmentem webu. Zahrnují nejnovější informace o tom, co se v obci nebo její oblasti děje. Aktuality mohou zahrnovat zprávy, novinky, aktuální události, pozvánky na akce, nebo jiné informace, které by mohly být pro návštěvníky nebo obyvatele zajímavé.
- **Dokumenty** jsou další částí segmentace webu. Zahrnují dokumenty, jako jsou Formuláře a tiskopisy, Zpravodaje, Služby, Dokumenty z Úřadu a další informační materiály, které mohou být užitečné pro návštěvníky webu a obyvatele obce.

- **Upozornění** se zaměřují na varování nebo důležitá upozornění pro návštěvníky. Mohou zahrnovat důležité informace z obce, plánované výpadky, varování nebo jiné důležité informace, které by mohly být užitečné pro obyvatele obce.
- **Události** se zaměřují na plánované události nebo akce. Mohou to být například plesy, sportovní akce, konference, představení, semináře, festivaly nebo jiné akce, které se konají v blízké budoucnosti.
- **Galerie** je posledním segmentem webu, který se zaměřuje pouze na vizuální informace. Galerie mohou zahrnovat fotografie, videa nebo jiné multimediální materiály.

5.5 Import a tvorba obsahu

Vytváření obsahu pro webové stránky ve WordPressu je relativně jednoduchý proces, který umožňuje rychle a snadno publikovat různé druhy obsahu na webových stránkách. Mezi nejpoužívanější patří:

- **Příspěvky (Posts)** - Příspěvky jsou základní jednotkou obsahu ve WordPressu. K publikování nových příspěvků stačí kliknout na tlačítko „Přidat nový“ v sekci Příspěvky v hlavním menu WordPressu. Zde se tvoří nový příspěvek, kterému lze napsat nadpis, přidat text, obrázky a další média, a poté ho publikovat.
- **Stránky (Pages)** - Stránky jsou velmi podobné jako příspěvky, ale slouží spíše k tvorbě statického obsahu, jako jsou kontaktní stránky, O nás, Podmínky, atd. Stejně jako při tvorbě příspěvku, nová stránka se vytváří pomocí tlačítka „Přidat novou“ v sekci Stránky v hlavním menu WordPressu. Poté lze přidat obsah stejně jako u příspěvku.
- **Vlastní typy příspěvků (Custom Post Types)** – K publikaci specifických typů obsahu, který není zahrnut v základním nastavení WordPressu, se využívá vlastní typ příspěvku pomocí pluginů jako je například „Custom Post Type UI“. Tento plugin umožňuje vytvářet vlastní typy příspěvků a přizpůsobit je svým potřebám. Vše lze opět udělat pomocí vlastního PHP kódu.
- **Taxonomie (Taxonomies)** - Taxonomie jsou kategorie nebo štítky, které mohou být použity k organizaci obsahu na webových stránkách. V základním nastavení WordPressu jsou k dispozici dvě taxonomie - kategorie a štítky. Vytvoření vlastní taxonomie se provádí pomocí pomocných pluginů, jako je například Custom Taxonomy UI, nebo pomocí vlastního PHP kódu.
- **Media** – Soubory, které lze nahrát do WordPressu a použít je v různých částech webové stránky. Lze nahrávat obrázky, videa, audia a další soubory. K nahrání nového média

stačí kliknout na „Média“ v hlavním menu WordPressu a poté kliknout na „Nahrát soubor“.

5.6 Implementace některých funkcí

5.6.1 Úřední deska

Ve WordPressu neexistuje žádný plugin, který by zajišťoval požadovanou funkčnost úřední desky. Z toho důvodu bylo nutné tuto funkčnost na web zcela implementovat.

Hlavním požadavkem bylo zajištění kompatibility navržené databáze se staršími soubory, které byly již uloženy v archivu. Mnoho dalších požadavků vyplývá z právních povinností úřední desky. Nezbytným požadavkem bylo zajistit zabezpečení, aby mohl do úřední desky přidávat soubory pouze administrátor.

Do úřední desky je možné vkládat soubory různého typu. Lze vytvořit zcela novou kategorii a přidat jednotlivé dokumenty. Jeden záznam v úřední desce může obsahovat i více položek. Hlavním inputem je „Název“, kde jeho délka může být maximálně 255 znaků a musí být vyplněna. Další povinným inputem je „Datum vyvěšení“ a „Datum sejmutí“. V případě, že je dokument po datu sejmutí, automaticky se přesune z aktuálně vyvěšených dokumentů do archivu. Jelikož některé záznamy nemusí mít datum sejmutí, jako jsou například Obecně závazné vyhlášky, je při zadání data povolena funkce zadat hodnotu 0000-00-00, která automaticky zajistí, že datum sejmutí bude nastaven na hodnotu „nikdy“. Další povinný input je typu select, u kterého se vybírá kategorie záznamu. A jako poslední jsou inputy pro možnost vložení souboru. Alespoň jeden soubor musí být přidán, jinak záznam nebude přidán.

Na stránce úřední desky lze vyhledávat pomocí pole „Hledat výraz“, filtrovat záznamy podle Kategorie, nebo filtrovat záznamy v rozmezí data vyvěšení a data sejmutí.

Některé PHP funkce, zajišťující funkčnost úřední desky.

```
function db_connection() {
    $database = new mysqli( hostname: DB_SERVER, username: DB_USERNAME, password: DB_PASSWORD, database: DB_NAMEE);
    if ($database->connect_errno > 0) {
        die("Nepodarilo se připojit k databázi" . $database->connect_error);
    }
    $database->set_charset( charset: "utf8");
    return $database;
}
```

Obrázek 18 – Připojení k databázi

Zdroj: autor

```
function loadAllCategories() {
    $db = db_connection();
    $sql = "SELECT * FROM kategorie ORDER BY ID ASC";
    $result = mysqli_query($db, $sql);
    $datas = array();
    if (mysqli_num_rows($result) > 0) {
        while ($row = mysqli_fetch_assoc($result)) {
            $datas[] = $row;
        }
    }
    return $datas;
}
```

Obrázek 19 – Načtení všech kategorií

Zdroj: autor

```
function loadAll() {
    $db = db_connection();
    $sql = "SELECT * FROM uredni_deska JOIN kategorie ON (uredni_deska.kategorie_id = kategorie.ID)";
    $result = mysqli_query($db, $sql);
    $datas = array();
    if (mysqli_num_rows($result) > 0) {
        while ($row = mysqli_fetch_assoc($result)) {
            $datas[] = $row;
        }
    }
    return $datas;
}
```

Obrázek 20 – Načtení všech záznamů z úřední desky

Zdroj: autor

5.6.2 Napojení webu na Muniopolis

Propojení webu a Muniopolis bylo provedeno pomocí RSS feedu, který umožňuje jednoduché a efektivní sdílení obsahu mezi webovými stránkami. RSS feed je formát pro distribuci obsahu, který umožňuje odběratelům sledovat aktualizace webového obsahu bez nutnosti pravidelně navštěvovat webové stránky. Pro propojení dvou webů pomocí RSS feedu se nejprve vytvoří RSS feed na zdrojovém webu, který bude sdílen. To lze udělat pomocí pluginů nebo funkcí, které jsou většinou dostupné v každé moderní platformě pro správu obsahu (CMS), jako je například ve WordPress. Po vytvoření RSS feedu se musí na cílovém webu importovat tento feed. To lze provést také pomocí pluginů nebo kódů, které jsou k dispozici v mnoha CMS. Nebo

se musí vše nastavit ručně. Importovaný obsah lze dále formátovat a zobrazit na cílovém webu libovolným způsobem.

Při propojení webů pomocí RSS feedu je důležité mít na paměti několik faktorů, aby bylo zajištěno správné fungování. Především je důležité věnovat pozornost formátování feedu pro zajištění, že obsah bude zobrazován v požadovaném formátu a bude dobře čitelný. Dále je důležité zvolit vhodný časový interval pro aktualizaci RSS feedu, aby se nový obsah zobrazoval včas a zároveň nezatěžoval příliš server. Při správném použití může být RSS feed velmi užitečným nástrojem pro sdílení obsahu mezi webovými stránkami.

5.6.3 Rezervační systém

Na webu je rovněž implementován rezervační systém Obecního domu a sportovní haly. Aktuálně ale není spuštěn z důvodu čekání na potvrzení přechodu na tuto online verzi ze strany obce. Plánované spuštění je v létě roku 2023. K vytvoření základních funkcí byl použit plugin pro tvorbu rezervací, který byl následně upraven podle požadavků obce.

duben 2023							květen 2023						
PO	ÚT	ST	ČT	PÁ	SO	NE	PO	ÚT	ST	ČT	PÁ	SO	NE
					1	2	1	2	3	4	5	6	7
3	4	5	6	7	8	9	8	9	10	11	12	13	14
10	11	12	13	14	15	16	15	16	17	18	19	20	21
17	18	19	20	21	22	23	22	23	24	25	26	27	28
24	25	26	27	28	29	30	29	30	31				

17/04/2023
22:00-23:00

SPORT* _____ JMÉNO* _____
PŘÍJMENÍ* _____ E-MAIL* _____
TELEFON* _____ POZNÁMKA _____

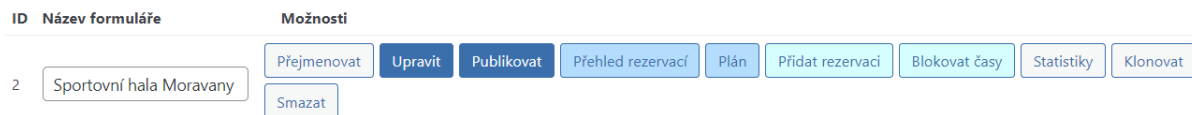
SOUHLASÍM SE ZPRACOVÁNÍM OSOBNÍCH ÚDAJŮ.*
Bezpečnostní kód:

Prosím, zadejte bezpečnostní kód: _____

Odeslat

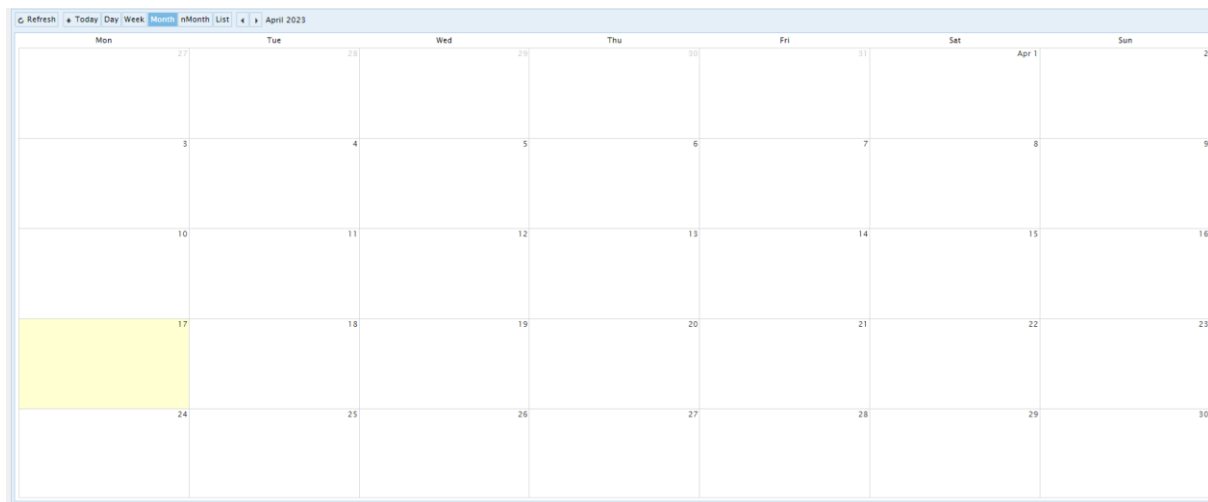
Obrázek 21 – Grafický vzhled rezervačního kalendáře

Zdroj: autor



Obrázek 22 – Administrační možnosti kalendáře

Zdroj: autor



Obrázek 23 – Přehled rezervovaných hodin

Zdroj: autor

5.6.4 Propojení se sociálními sítěmi

Obec Moravany používá pouze Facebook stránku. Nicméně navrženou webovou aplikaci pro obce a města je možné rozšířit i o jiné sociální síť. Propojení s webovou aplikací je možné například i pro Instagram, Twitter nebo WhatsApp. Vše je tedy připraveno pro případ, že obec bude v budoucnu potřebovat propojení s těmito aplikacemi.

K propojení Facebookové stránky Obce Moravany s webovou aplikací bylo zapotřebí vytvořit Business Manager, který umožňuje pro Facebookovou stránku používat pokročilé funkce jako je propojení s externími aplikacemi, které je nezbytné pro funkčnost sdílení obsahu z webové stránky. Na straně webové aplikace byl použit externí plugin, který umožňuje propojení Facebookového účtu s Wordpressem. Po propojení účtu s Wordpressem byly na straně webové aplikace nastaveny pravidla pro export příspěvků na Facebookovou stránku a také ošetřeno oprávnění, které umožňuje na Facebook sdílet příspěvky vytvořené uživatelem, který má nastavenou roli administrátora webu.

Tímto propojením se tak dosáhlo větší efektivity, jelikož nové příspěvky stačí přidat pouze na webovou stránku a automatizace se postará o přidání příspěvku na ostatní webové kanály za administrátora.

5.6.4.1 Business Manager (BM)

Business Manager je nástroj vyvinutý společností Facebook. Slouží pro správu firemních stránek, reklam, aplikací a dalších aktivit na Facebooku a Instagramu. BM pomáhá organizovat a spravovat online přítomnost firmám na platformách společnosti Meta, Inc., pod které patří aplikace jako Facebook, Instagram a WhatsApp. Meta je nový název společnosti Facebook, který byl oznámen v roce 2021. Předtím se společnost jmenovala Facebook, Inc. BM umožňuje vytvářet a spravovat firemní stránky, reklamní účty, přidělovat oprávnění nebo role pro jednotlivé členy týmu, sledovat výkonnost reklamních kampaní, spravovat katalogy produktů a mnoho dalších funkcí. V rámci Business Manageru lze vytvářet reklamní kampaně, sledovat analýzy, porovnávat výsledky reklamních kampaní, spravovat obsah na stránkách a komunikovat se zákazníky prostřednictvím messengeru. [26]

6 UMÍSTĚNÍ APLIKACE NA WEB

6.1 Doména

Doména je základním stavebním kamenem pro vytvoření webové stránky. Jedná se o unikátní adresu, která umožňuje uživatelům najít webovou stránku na internetu. Doména se skládá z názvu a přípony, například google.com nebo facebook.cz. Každá doména musí být registrována a může být využita pouze jedním majitelem. Registrace domény se provádí u akreditovaných registrátorů. V České republice existuje několik akreditovaných registrátorů, například CZ.NIC, Wedos, REGzone nebo ACTIVE 24. Tyto společnosti nabízejí registraci domény s různými příponami, jako jsou .com, .cz, .eu nebo .org. Cena za registraci se může lišit v závislosti na vybrané příponě a délce registrace.

Doména je důležitá pro každou webovou stránku, protože umožňuje uživatelům snadno najít stránku na internetu. Kromě toho má doména nemalý vliv na SEO. Vyhledávače preferují stránky s výstižným názvem domény, což může mít vliv na umístění stránky v hledání. Je důležité zvolit si vhodný název domény, který bude snadno zapamatovatelný a odlišný od ostatních stránek na internetu. Pokud je název domény již obsazen, může být obtížné najít alternativu, která bude stejně účinná. V případě osazení domény lze doménu překoupit, nicméně tato možnost může vyjít až na desítky tisíc nebo dokonce milióny.

Zabezpečení domény je také důležité. Registrátor musí zajistit, aby žádná jiná osoba nebo společnost nevyužívala stejnou doménu, a musí zajistit, aby byla doména chráněna před neoprávněným přístupem nebo útoky. Při koupi domény je možné si přibjednat další funkce, jako je například pravidelný monitoring domény.

Výběr a registrace domény je tedy velmi důležitá součást vytvoření webové stránky. Správná doména může mít vliv na SEO, zapamatovatelnost a rozpoznatelnost stránky na internetu. Proto je důležité pečlivě zvážit výběr domény a zvolit si akreditovaného registrátora, který zajistí bezpečnost a ochranu vaší domény.

6.2 Hosting

Hosting je služba poskytovaná společnostmi, které umožňují uživatelům nahrávat a spravovat webové stránky na internetu. Jedná se o úložiště, kde jsou umístěny soubory a data související s webovou stránkou, kde je daná webová stránka přístupná prostřednictvím internetu. Pro získání webhostingu je nutné kontaktovat hostingovou společnost a zvolit si plán,

který bude vyhovovat potřebám webové stránky. Lze vytvořit i vlastní hosting, nicméně tato možnost se v dnešní době nevyplatí a nejedná se o často používanou volbu. V případě vytvoření vlastního hostingu je potřeba zajistit několik funkcí, mezi které rovněž patří nepřetržitý provoz 24 hodin denně. Ve většině případů je lepší zvolit hostingové společnosti, které všechny potřebné funkce zajišťují s téměř nulovou chybovostí.

Hostingové společnosti nabízejí různé druhy hostingových služeb, jako jsou sdílené servery, VPS (Virtual Private Server) nebo dedikované servery. Sdílený server je vhodný pro menší webové stránky s nízkou návštěvností, kde se server sdílí s dalšími uživateli. VPS a dedikované servery jsou vhodné pro větší webové stránky, které vyžadují větší množství zdrojů a flexibilitu v nastavení serveru. Důležitost webhostingu spočívá v tom, že bez něj nelze webovou stránku provozovat. Kvalitní hostingová služba zajistí stabilitu a dostupnost vaší webové stránky, což je klíčové pro udržení návštěvnosti a dobrého výkonu. Pokud by webová stránka byla nedostupná, mohlo by to vést ke ztrátě potenciálních zákazníků a negativně ovlivnit dané podnikání. Je důležité zvolit hostingovou společnost s dobrým zázemím a bezpečnostními opatřeními, aby byly data a soubory chráněny před neoprávněným přístupem a útoky hackerů. Hostingová společnost by měla také poskytovat dostatečnou technickou podporu a možnost vzdáleného přístupu pro správu serveru a webové stránky.

Koupit hostingovou službu je velmi snadné a levné. Vše lze zakoupit přímo od hostingových společností. Pro výběr vhodného hostingu je třeba zvážit potřeby vaší webové stránky a vybrat plán, který bude splňovat tyto potřeby. Většina hostingových společností nabízí různé plány s různým množstvím prostoru a přenosovou kapacitou, což umožní zvolit přesně to, co je potřeba pro dané podnikání.

6.3 Zabezpečení SSL

SSL certifikát (Secure Sockets Layer) je důležitým nástrojem pro zabezpečení webových stránek. Jeho hlavním úkolem je zabezpečit přenos dat mezi webovým serverem a prohlížečem uživatele tak, aby byly data šifrované a chráněné proti případnému odposlechu či útokům ze strany třetích stran. Pro získání SSL certifikátu je třeba se obrátit na poskytovatele webhostingu nebo na specializovanou certifikační autoritu. Existuje několik typů certifikátů, které se liší svou úrovní zabezpečení a obvykle také cenou.

Po získání SSL certifikátu je třeba certifikát nainstalovat na webový server a aktivovat na webových stránkách. Nejprve je nutné ověřit, zda webhosting podporuje SSL certifikáty a zda

je certifikát správně nainstalován. Většina webhostingových služeb již dnes podporuje SSL certifikáty, takže většinou není žádný problém s jeho instalací.

Po instalaci je nutné upravit nastavení webového serveru tak, aby byl SSL certifikát aktivován. To se obvykle provádí v souboru .htaccess, kde je nutné vložit příslušný kód. Existují však i jiné způsoby, jak SSL certifikát aktivovat, záleží na konkrétní konfiguraci webového serveru. Po aktivaci SSL certifikátu je nutné provést změny na samotných webových stránkách. To zahrnuje úpravu odkazů tak, aby používaly protokol HTTPS místo HTTP, což zajistí, že všechny přenosy dat jsou šifrované a chráněné. Také je třeba ověřit, zda jsou všechny zdroje na webových stránkách (obrázky, skripty, styly) odkazovány pomocí HTTPS.

Zabezpečení webových stránek pomocí SSL certifikátu je důležitým krokem pro ochranu citlivých dat uživatelů a zajištění důvěryhodnosti webových stránek. Při nákupu domény a hostingu je proto důležité zvolit takové poskytovatele, kteří nabízejí podporu SSL certifikátů a umožňují snadnou instalaci a aktivaci tohoto zabezpečení.

6.4 Postup nahrávání

Po zakoupení domény a hostingového prostoru je nutné nahrát webové stránky na hostingové uložení, aby byly stránky dostupné na internetu.

6.4.1 Zkopírování obsahu webové stránky na hostingové uložení

Prvním krokem je nahrát soubory webové stránky WordPress na hostingové uložení. Nejdříve je nutné se připojit k FTP serveru. K připojení k FTP servu lze použít FTP klient jako je například FileZilla nebo TotalCommander. Po připojení do FTP účtu je nutné nahrát obsah a soubory webové stránky do požadované složky.

6.4.2 Vytvoření MySQL databáze

Dalším krokem je vytvoření MySQL databáze pro webovou stránku WordPress. Tento krok se liší podle poskytovatele hostingu, ale obvykle lze vytvořit databázi pomocí nástrojů pro správu webu, které jsou poskytovány v rámci kontrolního panelu hostingu.

6.4.3 Nastavení souboru wp-config.php

Po vytvoření MySQL databáze se musíte připojit k databázi pomocí souboru wp-config.php. Soubor wp-config.php se nachází ve složce WordPress souborů, které jste nahráli na váš hostingový účet. Musíte upravit tento soubor tak, aby se připojil k databázi.

```
// ** MySQL settings - You can get this info from your web host ** //  
/** The name of the database for WordPress */  
define( 'DB_NAME', 'název databáze' );  
  
/** MySQL database username */  
define( 'DB_USER', 'název uživatele' );  
  
/** MySQL database password */  
define( 'DB_PASSWORD', 'heslo' );  
  
/** MySQL hostname */  
define( 'DB_HOST', 'server' );  
  
/* Database Charset to use in creating database tables. */  
define( 'DB_CHARSET', 'nastavení znakové sady (utf8)' );  
  
/** The Database Collate type. Don't change this if in doubt. */  
define( 'DB_COLLATE', 'utf8_general_ci' );
```

Obrázek 24 – Připojení WordPressu k databázi

Zdroj: autor

6.4.4 Nastavení DNS

Po nahrání webové stránky na uložení hostingu a vytvoření databáze je potřeba nastavit DNS záznamy pro danou doménu. Pokud je doména zakoupena u poskytovatele hostingu, tento krok se provede automaticky. Pokud tomu tak není, musí se DNS záznamy nastavit ručně. V tomto případě se musí vytvořit nové DNS záznamy pro danou webovou stránku, včetně záznamu typu A, který odkazuje na IP adresu serveru, a záznamu typu CNAME, který odkazuje na název webové stránky. Po potvrzení nastavení se musí počkat, až se změny projeví. Většinou se změny projeví do 1 hodiny, ale v některých případech může změna nastavení trvat až 48

hodin. Po úspěšném dokončení lze funkčnosti ověřit pomocí nástrojů jako DNS Lookup nebo Ping.

6.4.5 Nastavení WordPress

Nakonec je potřeba nastavit WordPress tak, aby se připojil k databázi a zobrazoval webovou stránku. Potřebné nastavení se provede v administraci WordPress v záložce Nastavení.

To je celý proces nahrání webové stránky WordPress na hosting a doménu. Při dodržení všech kroků bude webová stránka k dispozici na nové doméně.

7 ZÁVĚR

Cílem bakalářské práce bylo vytvořit plně funkční, uživatelsky atraktivní a moderní webovou aplikaci, která slouží jako informační systém pro obce a města. Hlavním stanoveným cílem bylo splnění všech vládních nařízení pro tvorbu webových stránek pro obce a města. Dalším důležitým vytyčeným cílem bylo vytvoření zcela funkční úřední desky s obsahem kompletního archivu z předchozích let, do které je možné vkládat nové vyvěšené dokumenty s případnými přílohami a zajistit automatické přenesení starých dokumentů do archivu. Mezi další určené cíle patří vytvoření přehledné organizace a struktury webu, možnost přidávání událostí, dokumentů, aktualit, fotografií, zajištění přesměrování URL adres ze starého webu nebo vytvoření rezervačního systému, který bude v budoucnu použit pro rezervaci obecního domu obce a sportovní haly. Posledním vytyčeným cílem bylo propojení webu se sociálními sítěmi a se speciálním systémem Munipolos, který sjednocuje všechny obce a města.

Autor všechny tyto stanovené cíle splnil a vytvořil tak inovativní webovou aplikaci pro obec Moravany. Webová aplikace je již v testování několik měsíců. Občané i správci jsou velmi spokojeni s její funkčností. Autor zatím obdržel pouze pozitivní zpětnou vazbu, a to dokonce i od starosty obce. Autor je i nadále správcem obecního webu. Použití známých technologií, jako je redakční systém WordPress, PHP, MySQL, CSS, HTML 5 a JavaScript, zajišťuje spolehlivost a efektivitu webové aplikace. Unikátní funkcionalita propojení s úřední deskou a inovativní propojení se sociálními médii přináší další možnosti pro komunikaci a interakci s občany. Propojení zaměstnancům obce usnadňuje jejich práci a zefektivňuje celkovou správu digitálních kanálů. Nová oznámení přidaná na web jsou automaticky distribuována mezi další webové kanály jako je Munipolis nebo Facebook.

V teoretické části práce byly zmíněny systémy a firmy, které se touto problematikou zabývají. Byl zde popsán přehled o současném stavu moderních webových aplikací a požadavků na webové stránky pro obce. V teoretické části byly rovněž popsány použité technologie, které jsou potřebné k realizaci funkčního webové aplikace pro obce a města. Společně s popisem zde bylo zmíněno i několik praktických ukázek zmíněných technologií.

V praktické části byl představen návrh webové aplikace společně s požadavky. Všechny požadavky byly úspěšně naimplementovány, takže web splňuje všechny nařízení a požadavky. Byla zde rovněž detailně popsána databáze a vlastní struktura tabulek úřední desky. Poté zde byla podrobně popsána samotná realizace, kde byla detailně popsána struktura dat, segmentace webu a implementace některých funkcí. A jako poslední v této části bylo popsáno samostatné

umístění aplikace na web, postup registrace webhostingu s doménou, zabezpečení SSL a postup nahrávání aplikace na zakoupený webhosting.

Tato práce je dalším krokem směrem k modernizaci a digitalizaci webových stránek obcí a měst. S rozvojem technologií a digitálního prostředí je tato aplikace připravena na budoucnost a může sloužit jako vzor pro další podobné projekty. Aplikace může být snadno rozšířena o další funkce. Do budoucna je plánováno rozšíření webové aplikace obce o rezervační systém sportovní haly a obecního domu. Další integrací, na které se aktuálně pracuje, je záznam hlášení z rozhlasů v obci, které bude mít svůj vlastní archiv kvůli možnosti zpětného poslechu. Záznamy z rozhlasů budou automaticky ukládány na uložení FTP, ze kterého se následně budou číst data, která se budou dynamicky zobrazovat na webu obce. K implementaci budou použity výše uvedené technologie.

V digitálním světě 21. století je nezbytně nutné držet krok s modernizací a využívat potenciál webových aplikací. Držet krok je nezbytně nutné nejen pro lepší správu, ale i pro kvalitnější komunikaci obce s občany. Autor věří, že tato práce přispěje k efektivnějšímu fungování webových stránek a lepší správě obcí a měst, což bude mít pozitivní dopad na jejich technologický rozvoj.

POUŽITÁ LITERATURA

- [1] W3Techs. (n.d.). Používání protokolu HTTPS jako výchozího. Získáno 1. dubna 2023, z <https://w3techs.com/technologies/details/ce-httpsdefault>
- [2] ŘÍHA, Pavel. Šifrování - úvod do problematiky. Root [online]. 17. 11. 1999, [cit. 2023-03-10]. ISSN 1212-8309. Dostupné z: <http://www.root.cz/clanky/sifrovani-uvod-do-problematiky/>.
- [3] KOŘOUSKOVÁ, Barbora. HTTPS v KOSTCE: CO TO JE, JAK FUNGUJE a JAK NA NĚJ PŘEJÍT Rascasone [online]. 10. 3. 2023. [cit. 2021-11-09]. Dostupné z: <https://www.rascasone.com/cs/blog/co-je-https-http-ssl-tls>.
- [4] ZECHMEISTER, Jindřich. Analýza SSL certifikátů a HTTPS v českém internetu. SSL Market [online]. 13. 6. 2014. [cit. 2023-03-10]. Dostupné z: <https://www.sslmarket.cz/blog/analyza-ssl-certifikatu-a-https-v-ceskem-internetu>.
- [5] PIROUZ, Raymond. HTML: kouzla na Webu: [praktické postupy pro vytváření atraktivních webových stránek: více než 40 tipů a triků pro vaše webové stránky]. Brno: UNIS Publishing. ISBN 80-86097-17-X.
- [6] FREEMAN, Jamie. Webové stránky: pět set rad, tipů a technik: jednoduchý a praktický průvodce tajemstvím tvorby webových stránek. V Praze: Slovart. ISBN 978-80-7391-091-4.
- [7] TutorialsPoint. JavaScript Tutorial in PDF. [online] [cit. 2023-04-16]. Dostupné z: <https://matfuvit.github.io/UVIT/predavanja/literatura/TutorialsPoint%20JavaScript.pdf>
- [8] IHNAT, Štefan. Moderní trendy webdesignu: responzivní webové stránky. Pixel. 19(1), 42-43. ISSN 1211-5401.
- [9] Bootstrap: Get started. In: Bootstrap [online]. ©2023 [cit. 2023-03-17]. Dostupné z: <https://getbootstrap.com/docs/5.1/getting-started/introduction/>
- [10] PHP a MySQL: vytváříme dynamické webové stránky. Praha: Softpress. ISBN 80-86497-40-2.

- [11] W3Techs. Technologies: PHP Usage Statistics [online]. [cit. 2023-04-17]. Dostupné z: <https://w3techs.com/technologies/details/pl-php>
- [12] Lerner, David. PHP Cookbook: Solutions & Examples for PHP Programmers [online]. 3rd ed. ©2014 [cit. 2023-04-17]. Dostupné z: https://assets.ctfassets.net/nkydfjx48olf/5qFMF3mvitLMahX67i7iOb/028229996c13cbc27a0538f055a41b46/php_cookbook.pdf
- [13] Zhang, Z., Liu, X., Chen, X., Xu, L. and Lu, Y., 2018. A Parallel Model for Large-Scale Online Social Networks. IEEE Access, 6, pp.64899-64907. doi: 10.1109/ACCESS.2018.2883461.
- [14] phpMyAdmin. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [citováno 7. 3. 2023]. Dostupné z: <https://en.wikipedia.org/wiki/PhpMyAdmin>.
- [15] Brazell, Aaron. WordPress Bible 2nd edition. Indianapolis: Wiley Publishing, Inc. ISBN 978-0-470-93781-5. Dostupné z: https://books.google.cz/books?hl=cs&lr=&id=216an-wP9FkC&oi=fnd&pg=PA1&dq=WordPress&ots=PSreEcmlFV&sig=8bXKorKEXZ5s5iGrd5mgBxfhkkM&redir_esc=y#v=onepage&q=WordPress&f=false
- [16] SABIN-WILSON, Lisa. WordPress for dummies. John Wiley & Sons, 2011. Dostupné z: https://books.google.cz/books?hl=cs&lr=&id=cU8D6v3DqIoC&oi=fnd&pg=PR15&dq=WordPress&ots=Ox7_W8nvWk&sig=GMMR483Yo6OHBIUbPqwXO7wfkvw&redir_esc=y#v=onepage&q=WordPress&f=false
- [17] WILLIAMS, Brad; DAMSTRA, David; STERN, Hal. Professional WordPress: design and development. John Wiley & Sons, 2015. Dostupné z: https://books.google.cz/books?hl=cs&lr=&id=8AUaBgAAQBAJ&oi=fnd&pg=PR31&dq=WordPress&ots=7LdXz9h7y-&sig=R7Yr_FUUNO0oUTp7otGBXcFKZQY&redir_esc=y#v=onepage&q=WordPress&f=false
- [18] DE, Brajesh. API management. Apress, 2017.

- [19] KILLIAN, Ginger; MCMANUS, Kristy a marketing communications approach for the digital era: Managerial guidelines for social media integration. *Business horizons*, 2015, 58.5: 539-549.
- [20] IHNAT, Štefan. Jak realizovat responzivní web. *Pixel*. 19(2), 44-45. ISSN 1211-5401.
- [21] W3Schools Free Online Web Tutorials. (n.d.). W3Schools Online Web Tutorials. <https://www.w3schools.com/>
- [22] PHP: Hypertext Preprocessor. (2023, April 13). PHP: Hypertext Preprocessor. <https://www.php.net>
- [23] Blog Tool, Publishing Platform, and CMS - WordPress.org. (2023, April 3). Blog Tool, Publishing Platform, and CMS – WordPress.org. <https://wordpress.org/>
- [24] D'AGOSTINO, Maria J., et al. A study of e-government and e-governance: An empirical examination of municipal websites. *Public Administration Quarterly*, 2011, 3-25.
- [25] YOUNGBLOOD, Norman E.; MACKIEWICZ, Jo. A usability analysis of municipal government website home pages in Alabama. *Government Information Quarterly*, 2012, 29.4: 582-588.
- [26] Facebook. (n.d.). Facebook. <https://www.facebook.com/business/help>
- [27] WELLING, Luke; THOMSON, Laura. *PHP and MySQL Web development*. Sams publishing, 2003.
- [28] GREENSPAN, Jay; BULGER, Brad. *MySQL/PHP database applications*. John Wiley & Sons, Inc., 2001.

PŘÍLOHY

Příloha A – Odkaz na webovou stránku obce Moravany.....	68
---------------------------------------------------------	----

PŘÍLOHA A – ODKAZ NA WEBOVOU STRÁNKU OBCE MORAVANY

<https://obec-moravany.cz/>