DELTA - Střední škola informatiky a ekonomie, s.r.o.
Ke Kamenci 151, Pardubice

Chatky24

Ondřej Bělohradský
4.A
Informační technologie (18-20-M/01)

2024/2025

Poděkování

Chtěl bych poděkovat Ing. Monice Borkovcové, Ph.D. za vedení tohoto maturitního projektu a konzultaci problémů. Dále bych chtěl poděkovat Danielu Vaclavíkovi za pomoc se designovou stránkou mého projektu, Patriku Stohanzlovi za pomoc při problémech se syntaxí.

Anotace

Cílem projektu je vytvořit webovou aplikaci, která umožní snadnou a pohodlnou rezervaci chat a chalup. Uživatelé budou mít možnost vyhledávat dostupné objekty podle různých kritérií, detailně si je prohlédnout, provádět rezervace a následně spravovat své pobyty. Aplikace nabídne moderní a intuitivní uživatelské rozhraní, které zajistí přehlednou navigaci a snadné ovládání. Součástí bude také efektivní administrační prostředí, které umožní správcům jednoduše řídit nabídku objektů, kontrolovat rezervace a spravovat uživatelské účty.

Klíčová slova

Next.js, Typescript, MongoDB, Stripe, Webová aplikace, Rezervace

Annotation

The aim of the project is to create a web application that will enable easy and convenient booking of chalets and cottages. Users will be able to search for available properties according to various criteria, view them in detail, make reservations and then manage their stays. The application will offer a modern and intuitive user interface that will ensure clear navigation and easy operation. It will also include an effective administration environment that will allow administrators to easily manage the offer of properties, check reservations and manage user accounts.

Keywords

Next.js, Typescript, MongoDB, Stripe, Web Application, Reservations

Zadání maturitního projektu z informatických předmětů

Jméno a příjmení:	*Ondřej Bělohradský*
Školní rok:	*2024/2025*
Třída:	*4.A*
Obor:	*Informační technologie 18-20-M/01*
Téma práce:	*Webová aplikace pro rezervaci chat a chalup*
Vedoucí práce:	*Ing. Monika Borkovcová, Ph.D.*

Způsob zpracování, cíle práce, pokyny k obsahu a rozsahu práce

Cílem projektu je vytvoření webové aplikace pro rezervaci chat a chalup, která nabídne uživatelům intuitivní prostředí pro vyhledávání a správu rezervací. Aplikace bude obsahovat uživatelskou a administrátorskou část.

Popis funkčnosti aplikace:

- **Registrace a správa účtu** – Uživatelé mohou spravovat svůj účet a rezervace.
- **Filtrace chat** – Vyhledávání dle lokality, ceny a dalších parametrů.
- **Rezervace chaty/chalupy** – Výběr, ověření dostupnosti a rezervace.
- **Platba a notifikace** – Online platba s potvrzovacím emailem.
- **Správa uživatelů a rezervací** – Administrátor spravuje data.
- **Kontrola obsazenosti a kalkulace ceny** – Přehled dostupnosti a nákladů.

Stručný časový harmonogram

Březen - Srpen: Analýza obdobných řešení, návrh architektury, výběr technologií
Září - Prosinec: Implementace funkcí, vývoj rezervačního systému, integrace plateb
Leden - Únor: Testování aplikace, pokročilé filtrování, mapová vizualizace
Březen: Dokončení administrace, podpora jazyků, finální dokumentace

Prohlašuji, že jsem maturitní projekt vypracoval samostatně, výhradně s použitím uvedené literatury.

V Pardubicích, 31.3.2025

Ondřej Bělohradský

Obsah

Úvod **1**

1 Porovnání s ostatními aplikacemi na trhu **2**
1.1 Booking.com . . . 2
1.2 Airbnb . . . 2
1.3 Czech-cottages.cz . . . 3
1.4 e-Chalupy.cz . . . 3

2 Technologie **4**
2.1 Licence . . . 4
2.2 Next.js . . . 4
2.2.1 Hlavní funkce Next.js . . . 4
2.2.1.1 Server-side Rendering (SSR) . . . 4
2.2.1.2 Static Site Generation (SSG) . . . 4
2.2.1.3 Hybridní přístup . . . 5
2.2.1.4 API Routes . . . 5
2.2.1.5 Optimalizace výkonu . . . 5
2.2.2 Next.js a React . . . 5
2.2.3 Next.js pro produkci . . . 5
2.3 Node.js . . . 6
2.3.1 Hlavní vlastnosti . . . 6
2.3.1.1 Asynchronní I/O . . . 6
2.3.1.2 NPM . . . 6
2.3.1.3 Event-driven . . . 6
2.4 MongoDB . . . 6
2.4.1 Hlavní vlastnosti . . . 6
2.4.1.1 Dokumentově orientovaná databáze . . . 6
2.4.1.2 Škálovatelnost . . . 6
2.4.1.3 Vysoká dostupnost . . . 7
2.5 TypeScript . . . 7
2.5.1 Hlavní vlastnosti . . . 7
2.5.1.1 Statické typování . . . 7
2.5.1.2 Kompilace do JavaScriptu . . . 7
2.5.1.3 Podpora moderních funkcí . . . 7

2.6 Tailwind CSS . . . 7
2.6.1 Hlavní vlastnosti . . . 8
2.6.1.1 Utility-first přístup . . . 8
2.6.1.2 Responzivita . . . 8
2.6.1.3 Přizpůsobení . . . 8
2.6.2 Příklad použití . . . 8
2.7 Stripe . . . 8
2.7.1 Hlavní vlastnosti . . . 9
2.7.1.1 Jednoduchá integrace . . . 9
2.7.1.2 Bezpečnost . . . 9
2.7.1.3 Podpora více metod plateb . . . 9
2.7.2 Příklad použití . . . 9
2.8 Brevo API . . . 9
2.8.1 Hlavní vlastnosti . . . 10
2.8.1.1 E-mailový marketing . . . 10
2.8.1.2 SMS marketing . . . 10
2.8.1.3 Automatizace . . . 10
2.8.1.4 CRM integrace . . . 10
2.8.2 Výhody používání Brevo API . . . 10
2.9 Cloudinary API . . . 10
2.9.1 Hlavní vlastnosti . . . 10
2.9.1.1 Nahrávání médií . . . 10
2.9.1.2 Transformace a úpravy . . . 11
2.9.1.3 Optimalizace médií . . . 11
2.9.1.4 Doručování přes CDN . . . 11
2.9.2 Výhody . . . 11
2.10 Google Maps API v Reactu s TypeScriptem . . . 11
2.10.1 Kroky pro použití . . . 11
2.10.2 Příklad použití v Reactu s TypeScriptem . . . 11
2.10.3 Popis příkladu . . . 12
2.11 Dodatečné technologie . . . 12
2.11.1 package.json . . . 12
2.11.2 Příklad package.json . . . 13

3 Chatky24 **14**
3.1 Popis architektury . . . 14
3.1.1 Backend . . . 14
3.1.2 Frontend . . . 14

3.2 Backend . . . 15
3.2.1 MongoDB databáze . . . 15
3.2.1.1 Proč právě MongoDB . . . 15
3.2.1.2 Návrh databáze . . . 16
3.2.2 Server . . . 17
3.2.2.1 Zabezpečení API pomocí JWT . . . 17
3.2.2.2 Struktura API endpointů . . . 18
3.2.3 Stripe pro platby . . . 19
3.2.4 Odesílání emailů skrze Brevo (dříve Sendinblue) . . . 19
3.3 Frontend . . . 20
3.3.1 Proč jsem zvolil Next.js . . . 20
3.3.2 Výběr TypeScriptu jako hlavního jazyk . . . 20
3.3.3 Proč Tailwind . . . 20
3.3.4 Proč Cloudinary . . . 21
3.3.5 Proč Google Maps API . . . 21
3.4 Výsledná aplikace . . . 21
3.4.1 Rezervační systém . . . 22
3.4.2 Hledání ideální chaty . . . 22
3.4.2.1 Filtrování . . . 22
3.4.3 Navbary . . . 24
3.4.3.1 Přihlášen . . . 24
3.4.3.2 Nepřihlášen . . . 24
3.4.4 Přihlášení/Registrace . . . 24
3.4.4.1 Stránka pro přihlášení . . . 24
3.4.4.2 Stránka pro registraci . . . 24
3.4.5 Chaty . . . 26
3.4.5.1 Nabídka chat . . . 26
3.4.5.2 Detaily Chaty . . . 27
3.4.5.3 Cena za chatu . . . 28
3.4.5.4 Obrázky . . . 28
3.4.5.5 Mapa . . . 29
3.4.5.6 Komentáře . . . 29
3.4.6 Rezervace . . . 30
3.4.6.1 Moje rezervace . . . 30
3.4.6.2 Platba . . . 31
3.4.6.3 Přidání komentáře . . . 32
3.4.7 Profil . . . 33

3.4.7.1 Základní údaje 33
3.4.8 Email komunikace 34
3.4.8.1 Obnova zapomenutého hesla 34
3.4.8.2 Potvrzovací e-mail o rezervaci 35
3.4.9 Administrátor 36

4 Budoucnost **37**
4.1 Budoucnost aplikace Chatky24 37
4.1.1 Co v aplikaci chybí 37
4.1.1.1 Zabezpečení 37
4.1.1.2 Chybějící funkce a vylepšení 37
4.1.1.3 Optimalizace 38
4.1.2 Plány do budoucna 38

Závěr **39**

Literatura **40**

Seznam ukázek kódu **41**

Seznam obrázků **42**

Úvod

V současné době je na trhu k dispozici velké množství aplikací pro správu a rezervace chat a chalup. Tyto aplikace slouží k vyhledávání, rezervaci a správě ubytování v rekreačních objektech. Jedná se o běžně používané softwarové nástroje, které usnadňují proces hledání a rezervace vhodného ubytování pro dovolenou či víkendový pobyt. Aplikace pro správu chat a chalup jsou převážně dostupné ve formě webových nebo mobilních aplikací.

Základní funkce, které takové aplikace obvykle obsahují, zahrnují vyhledávání a filtrování ubytování, správu rezervací, stanovení termínů pobytu, možnosti hodnocení a recenzí, fotogalerie objektů a například i integraci s mapovými službami. Často bývá součástí těchto aplikací i funkcionalita přímé komunikace mezi nájemcem a pronajímatelem, která usnadňuje domluvu detailů pobytu.

Aplikace pro správu chat a chalup cílí buď na jednotlivce hledající ubytování, nebo na majitele nemovitostí, kteří chtějí své objekty pronajímat. Problém nastává, jakmile má uživatel zájem o komplexní řešení, které by umožňovalo jak snadné vyhledávání a rezervaci, tak i správu vlastních nemovitostí. V tomto případě je často nucen používat dvě oddělené aplikace. Aplikace na trhu také bývají buď velmi zjednodušené, nebo naopak příliš komplexní, což může paradoxně ztížit proces rezervace a správy ubytování. Komplexnější aplikace bývají také placené nebo poskytují jen velmi omezenou dostupnost funkcionalit v základní neplacené verzi.

Cílem maturitní práce je vytvořit intuitivní webovou aplikaci pro správu a rezervace chat a chalup, která umožňuje rychlé vyhledávání, snadnou rezervaci a efektivní správu ubytovacích objektů. Uživatelé aplikace budou moci snadno filtrovat nabídky podle různých kritérií, jako je lokalita, cena, kapacita či vybavení. Rezervace budou mít jasně stanovené termíny a uživatelé je uvidí v integrovaném kalendáři. Aplikace také poskytne majitelům nemovitostí nástroje pro správu jejich objektů a rezervací.

Tato práce je rozdělena do dvou základních částí. První část se soustředí na vysvětlení základních pojmů a analýzu podobných aplikací na trhu, které popisuje kapitola 2. Následující kapitola pak seznamuje se "základním technologickým stackem". V práci bude použit Next.js framework postavený na React knihovně, TypeScript pro typovou bezpečnost, MongoDB jako databáze, Tailwind CSS pro styling, Google Maps API pro mapovou vizualizaci, Stripe API pro zpracování plateb a další moderní technologie pro vývoj webových aplikací.

Druhá část se již prakticky věnuje samotnému vývoji webové aplikace. Popis tvorby zahrnující jednotlivé kroky vývoje webové aplikace od analýzy až po implementaci je popsán v kapitole 4 Aplikace.

1. Porovnání s ostatními aplikacemi na trhu

V rámci teoretické části práce jsem se před vývojem vlastní aplikace pro správu a rezervace chat a chalup rozhodl porovnat existující řešení na trhu. Průzkum probíhal v týdnu 15.09.2023 - 22.09.2023. Na trhu se již vyskytuje několik významných platforem pro rezervaci ubytování. Níže jsou popsány nejčastěji používané aplikace pro rezervaci chat a chalup na trhu v ČR.

1.1 Booking.com

Booking.com je jednou z nejznámějších platforem pro rezervaci ubytování. Nabízí širokou škálu ubytování, včetně hotelů, apartmánů, a také chat a chalup. Hlavní výhody:

- Rozsáhlá databáze ubytování po celém světě
- Uživatelsky přívětivé rozhraní s možností filtrace
- Systém hodnocení a recenzí od uživatelů

Nevýhody:

- Není primárně zaměřen na chaty a chalupy, což omezuje výběr a možnosti filtrace pro tento typ ubytování
- Vysoké provize pro majitele ubytování

[1]

1.2 Airbnb

Airbnb je platforma, která umožňuje pronájem různých typů nemovitostí, včetně chat a chalup. Hlavní výhody:

- Široká nabídka unikátních ubytování
- Přímá komunikace mezi hostitelem a hostem
- Flexibilní cenová politika

Nevýhody:

- Pro pronajímatele je složitější a vyžaduje vysoké poplatky
- Méně specializovaná na chaty a chalupy v ČR

[2]

1.3 Czech-cottages.cz

Czech-cottages.cz je platforma specializující se na chaty a chalupy v České republice. Hlavní výhody:

- Zaměření přímo na chaty a chalupy v ČR
- Detailní informace o ubytování a okolí

Nevýhody:

- Zastaralé uživatelské rozhraní
- Omezené rezervační funkce

[3]

1.4 e-Chalupy.cz

e-Chalupy.cz je lokální platforma s širokou nabídkou chat a chalup v České republice. Hlavní výhody:

- Rozsáhlá databáze chat a chalup v ČR
- Detailní filtry pro vyhledávání

Nevýhody:

- Absence možnosti okamžité online rezervace
- Méně uživatelsky přívětivé rozhraní

Na základě tohoto průzkumu jsem identifikoval několik klíčových oblastí, na které se zaměřím při vývoji vlastní aplikace:

1. Jednoduchost ovládání a přehledné uživatelské rozhraní
2. Specializace na chaty a chalupy s možností detailní filtrace
3. Implementace systému okamžité online rezervace
4. Přímá komunikace mezi nájemcem a pronajímatelem
5. Flexibilní cenová politika s nižšími poplatky pro pronajímatele
6. Integrovaný systém hodnocení a recenzí

[4]

2. Technologie

2.1 Licence

Jednou z prvních věcí, kterou je vhodné zvážit ještě než na projektu začneme pracovat, je licence pod kterou projekt bude zpracován. Pro projekt byla zvolena MIT Licence, také známá jako Expat nebo X11[**GNU-Mit**]. MIT Licence vznikla na Massachusettském technologickém institutu roku 1988 a umožňuje ostatním uživatelům software jakkoliv upravovat a vydávat nové verze programu, i pod jinou licencí, pokud uvedou autora a ponechají původní kód pod MIT Licencí[**Github-Mit**].[5]

2.2 Next.js

Next.js[1] je vysoce populární open-source framework pro vývoj webových aplikací postavený na knihovně `React`. Umožňuje vývojářům snadno vytvářet aplikace s vysoce optimalizovaným výkonem, a to díky svým pokročilým funkcím jako je server-side rendering (SSR), statická generace (SSG) a hybridní přístup, který kombinuje obě metody. Next.js se zaměřuje na zjednodušení vývoje aplikací a poskytuje silné nástroje pro práci s routingem, API a server-side logikou.

2.2.1 Hlavní funkce Next.js

`Next.js` nabízí širokou škálu funkcí, které usnadňují tvorbu výkonných webových aplikací. Některé z hlavních funkcí zahrnují:

2.2.1.1 Server-side Rendering (SSR)

Server-side rendering[2] je klíčová vlastnost Next.js. Tento přístup zajišťuje, že uživatelé dostanou již renderovaný obsah na stránce, což vede k rychlejšímu načítání stránky a lepšímu SEO. SSR je ideální pro aplikace, kde je důležitá SEO optimalizace a kde se obsah často mění.

2.2.1.2 Static Site Generation (SSG)

Static site generation[3] je dalším důležitým konceptem v Next.js. Stránky jsou generovány na serveru při nasazení aplikace, což znamená, že uživatelé obdrží statické soubory, které se rychle

[1] Zkráceně `Next.js`

[2] SSR umožňuje vykreslování obsahu na serveru před jeho odesláním do prohlížeče.

[3] SSG umožňuje generovat statické stránky během build procesu.

načítají. SSG je ideální pro aplikace, které mají obsah, který se nemění často (např. blogy nebo landing pages).

2.2.1.3 Hybridní přístup

Next.js také podporuje hybridní přístup, kde je možné kombinovat SSR a SSG na jednom projektu. To znamená, že různé stránky mohou používat různé metody renderování podle potřeby. Tato flexibilita umožňuje vývojářům optimalizovat výkon na základě specifických požadavků projektu.

2.2.1.4 API Routes

Next.js poskytuje podporu pro API routes[4] v rámci projektu. To znamená, že vývojáři mohou jednoduše vytvářet serverové funkce přímo v rámci aplikace, aniž by bylo potřeba nastavovat oddělený backend server. API routes jsou ideální pro jednoduché backendové operace, jako je práce s databázemi, autentizace nebo interakce s externími API.

2.2.1.5 Optimalizace výkonu

Next.js obsahuje vestavěné nástroje pro optimalizaci výkonu aplikací. To zahrnuje automatické dělení kódu[5], minimalizaci velikosti souborů a optimalizaci obrázků. Kromě toho poskytuje nástroje pro optimalizaci načítání stránek, což vedlo k rychlému rozšíření tohoto frameworku pro komerční aplikace.

2.2.2 Next.js a React

`Next.js` je postavený na `React`, což znamená, že všechny funkce a komponenty Reactu jsou plně podporovány. Kromě toho Next.js nabízí několik vylepšení, které umožňují snadnou integraci Reactu s různými renderovacími technikami, jako je SSR a SSG. Když vývojáři využívají Next.js, mohou využívat všechny výhody Reactu, včetně komponent, hooks, context API a dalších funkcí, které umožňují vytvářet dynamické a interaktivní webové aplikace.

2.2.3 Next.js pro produkci

Next.js je ideální pro aplikace, které jsou nasazeny v produkčním prostředí. Framework podporuje různé možnosti nasazení, včetně nasazení na cloudové platformy jako Vercel, AWS a další. Je také optimalizován pro práci s CDN (Content Delivery Network), což zajišťuje rychlý přístup k aplikaci pro uživatele po celém světě. Next.js se také snadno integruje s externími službami, jako jsou databáze, cloudové úložiště a další nástroje pro správu aplikací. [6]

[4] API routes umožňují definovat backendovou logiku přímo v rámci Next.js aplikace.
[5] Code splitting rozděluje kód na menší části, které se načítají pouze tehdy, když jsou potřeba.

2.3 Node.js

Node.js[6] je open-source platforma pro vývoj serverových aplikací, postavená na V8 JavaScript engine od Google.

2.3.1 Hlavní vlastnosti

2.3.1.1 Asynchronní I/O

Node.js používá asynchronní a neblokující I/O operace, což umožňuje efektivní zpracování více požadavků současně.

2.3.1.2 NPM

NPM (Node Package Manager) je nástroj pro správu balíčků, který usnadňuje přidávání knihoven a nástrojů do projektů.

2.3.1.3 Event-driven

Node.js využívá event-driven model, kde aplikace reaguje na události, což zajišťuje vysoký výkon při zpracování požadavků.[7]

2.4 MongoDB

MongoDB[7] je populární NoSQL databáze, která používá model dokumentů místo tradičních tabulek. Umožňuje flexibilní strukturu dat a snadno se přizpůsobuje změnám v aplikacích.

2.4.1 Hlavní vlastnosti

2.4.1.1 Dokumentově orientovaná databáze

Data jsou ukládána jako dokumenty ve formátu BSON[8], což zajišťuje vysokou flexibilitu a rychlost při práci s různými datovými strukturami.

2.4.1.2 Škálovatelnost

MongoDB je navržen pro horizontální škálování, což znamená, že lze snadno přidávat nové servery pro zvýšení kapacity bez výrazného zhoršení výkonu.

[6] Node.js je runtime prostředí pro JavaScript umožňující běh kódu mimo prohlížeč.
[7] MongoDB je NoSQL databázový systém, který ukládá data ve formátu dokumentů ve formátu JSON.
[8] BSON je binární formát JSON, který umožňuje ukládání dat s různými datovými typy.

2.4.1.3 Vysoká dostupnost

MongoDB podporuje replikaci dat mezi více servery, což zajišťuje vysokou dostupnost a odolnost proti výpadkům.[8]

2.5 TypeScript

TypeScript[9] je jazyk vyvinutý společností Microsoft, který rozšiřuje JavaScript o podporu statického typování a objektově orientovaného programování. TypeScript se kompiluje do JavaScriptu, což znamená, že může běžet v jakémkoliv prostředí, které podporuje JavaScript.

2.5.1 Hlavní vlastnosti

2.5.1.1 Statické typování

TypeScript umožňuje definovat datové typy pro proměnné, funkce a objekty. To přispívá k lepší čitelnosti kódu, prevenci chyb a efektivnějšímu ladění.

2.5.1.2 Kompilace do JavaScriptu

TypeScript kód je kompilován do čistého JavaScriptu, což znamená, že jej lze používat v libovolném prostředí podporujícím JavaScript, včetně webových prohlížečů a serverů.

2.5.1.3 Podpora moderních funkcí

TypeScript podporuje moderní funkce JavaScriptu (např. asynchronní funkce, třídy, rozhraní), ale umožňuje také využívat funkce, které budou dostupné až v budoucích verzích JavaScriptu.[9]

2.6 Tailwind CSS

Tailwind CSS[10] je moderní CSS framework, který se zaměřuje na princip utility-first, což znamená, že místo definování vlastních tříd pro každý komponent využívá existující utility třídy, které určují vzhled elementů. Tento přístup umožňuje rychlé prototypování a výrazně zjednodušuje práci s CSS.

[9] TypeScript je nadmnožina jazyka JavaScript, která přidává statické typování.

[10] Tailwind CSS je utility-first CSS framework, který poskytuje množství nízkoúrovňových utilit pro rychlé vytváření designů.

2.6.1 Hlavní vlastnosti

2.6.1.1 Utility-first přístup

Tailwind nabízí množství utility tříd pro práci s rozestupy, barvami, fonty, pozicováním a dalšími vlastnostmi. Tento přístup zjednodušuje kódování a zajišťuje větší flexibilitu při tvorbě designu bez potřeby psát vlastní CSS.

2.6.1.2 Responzivita

Tailwind nabízí snadný způsob, jak implementovat responzivní design pomocí předdefinovaných tříd, které umožňují specifikovat různé styly pro různé šířky obrazovky, což usnadňuje tvorbu adaptivních aplikací.

2.6.1.3 Přizpůsobení

Tailwind je vysoce přizpůsobitelný díky konfiguračnímu souboru, kde je možné měnit základní hodnoty jako barvy, písma, breakpoints a další vlastnosti podle potřeb projektu.

2.6.2 Příklad použití

Zde je jednoduchý příklad, jak pomocí Tailwind CSS vytvořit responzivní kartu, která se přizpůsobí velikosti obrazovky:

```
<div class="max-w-sm rounded overflow-hidden shadow-lg bg-white p-6">
  <h2 class="text-xl font-semibold text-gray-800">Název karty</h2>
  <p class="text-gray-600 mt-2">Toto je příklad karty s textem a obrá
    ↪ zkem.</p>
  <button class="mt-4 px-4 py-2 bg-blue-500 text-white rounded hover:bg-
    ↪ blue-700">Klikni zde</button>
</div>
```

Ukázka kódu 2.1 Ukázka integrace Tailwindu v aplikaci React s TypeScriptem

Tento kód vytváří jednoduchou kartu s textem a tlačítkem. Využívá utility třídy pro ovládání šířky, zaoblení rohů, stínování, barvy pozadí, textu a interakce s tlačítkem. Karta se automaticky přizpůsobí velikosti obrazovky, díky použitému designu.[10]

2.7 Stripe

Stripe[11] je široce používaná platforma pro zpracování plateb online. Umožňuje podnikům přijímat platby pomocí karet, bankovních převodů a dalších metod. Stripe nabízí snadné API pro

[11] Stripe je populární platební platforma, která umožňuje online platby a zpracování transakcí. Stripe poskytuje rozhraní API pro snadnou integraci platebních funkcí do webových aplikací.

integraci do webových aplikací a poskytuje nástroje pro správu transakcí, vrácení plateb a výstupy pro účetní potřeby.

2.7.1 Hlavní vlastnosti

2.7.1.1 Jednoduchá integrace

Stripe umožňuje jednoduchou integraci platebních metod na webové stránky a aplikace pomocí API, což usnadňuje implementaci platebních bran.

2.7.1.2 Bezpečnost

Stripe se soustředí na bezpečnost platebních údajů a zajišťuje, že všechny transakce jsou chráněny pomocí šifrování a splňují standardy PCI DSS.

2.7.1.3 Podpora více metod plateb

Stripe podporuje širokou škálu platebních metod, včetně kreditních a debetních karet, Apple Pay, Google Pay a dalších regionálních metod.

2.7.2 Příklad použití

Zde je jednoduchý příklad, jak integrovat Stripe do backendu aplikace v Node.js pro vytvoření platby:

```
1 [caption={Vytvoření platebního záměru pomocí Stripe API}] const stripe =
    ↪ require('stripe')('sk_test_yourSecretKey');  app.post('/create-
    ↪ payment-intent', async (req, res) => {   const paymentIntent =
    ↪ await stripe.paymentIntents.create({     amount: 2000, // částka v
    ↪  centích     currency: 'usd',   });   res.send({     clientSecret:
    ↪  paymentIntent.client_secret,   }); });
```

Ukázka kódu 2.2 Ukázka integrace Stripe v Node.js

[11]

2.8 Brevo API

Brevo[12] je služba, která poskytuje API pro správu e-mailů, SMS zpráv a dalších komunikačních kanálů. Brevo je oblíbené pro svou jednoduchost, bohatou funkcionalitu a přehlednost, a je široce používáno pro rozesílání e-mailů, správu kontaktů a automatizaci marketingových kampaní.

[12]Brevo, dříve známý jako Sendinblue, je platforma pro automatizaci marketingu a e-mailovou komunikaci. Nabízí nástroje pro e-mailový marketing, SMS kampaně, automatizaci, CRM a další.

2.8.1 Hlavní vlastnosti

2.8.1.1 E-mailový marketing

Brevo umožňuje vytváření a odesílání e-mailových kampaní. Uživatelé mohou využívat pokročilé funkce pro personalizaci zpráv, správu seznamů a analýzu výsledků kampaní.

2.8.1.2 SMS marketing

Kromě e-mailů Brevo podporuje i odesílání SMS zpráv, což je užitečné pro přímou komunikaci se zákazníky a zvýšení interakce.

2.8.1.3 Automatizace

S Brevo lze nastavit různé marketingové automatizace, jako jsou e-maily odesílané po určité akci uživatele nebo specifickém časovém intervalu.

2.8.1.4 CRM integrace

Brevo poskytuje integrované CRM nástroje pro správu kontaktů a sledování interakcí se zákazníky, což usnadňuje správu vztahů s klienty a cílení kampaní.

2.8.2 Výhody používání Brevo API

Brevo API umožňuje snadnou integraci všech těchto funkcí přímo do webových aplikací. S využitím API mohou vývojáři snadno automatizovat odesílání e-mailů, správu uživatelů a sledování kampaní bez potřeby složitých konfigurací. Brevo je skvélé pro firmy, které hledají flexibilní a efektivní nástroj pro digitální marketing.[12]

2.9 Cloudinary API

Cloudinary[13] je služba pro nahrávání, transformaci, optimalizaci a doručování multimediálního obsahu. Umožňuje efektivní správu mediálních souborů bez nutnosti jejich hostování na vlastních serverech.

2.9.1 Hlavní vlastnosti

2.9.1.1 Nahrávání médií

Cloudinary umožňuje nahrávat obrázky a videa přímo do cloudového úložiště přes API.

[13]Cloudinary je cloudová platforma pro správu a zpracování obrázků a videí.

2.9.1.2 Transformace a úpravy

S Cloudinary lze dynamicky upravovat obrázky a videa, například měnit velikost, oříznout nebo přidávat efekty.

2.9.1.3 Optimalizace médií

Platforma automaticky optimalizuje obsah pro různé zařízení a připojení, zajišťující rychlé načítání.

2.9.1.4 Doručování přes CDN

Cloudinary využívá globální CDN pro rychlé doručení obsahu uživatelům.

2.9.2 Výhody

Cloudinary umožňuje efektivní práci s multimediálním obsahem, automatizuje optimalizaci a urychluje doručování médií bez nutnosti složité infrastruktury.[13]

2.10 Google Maps API v Reactu s TypeScriptem

Integrace Google Maps API do aplikace postavené na Reactu a TypeScriptu je jednoduchá, pokud použijeme knihovnu `@react-google-maps/api`. Tato knihovna poskytuje komponenty, které umožňují snadnou práci s Google Mapami v aplikaci React.

2.10.1 Kroky pro použití

1. Nainstalujte knihovnu `@react-google-maps/api` pomocí `npm` nebo `yarn`. 2. Vytvořte komponentu, která bude obsahovat Google Mapu. 3. Použijte `GoogleMap` komponentu a přidejte mapu s markerem.

2.10.2 Příklad použití v Reactu s TypeScriptem

```
1 import React from 'react';
2 import { GoogleMap, LoadScript, Marker } from '@react-google-maps/api';
3
4 const containerStyle = {
5   width: '100%',
6   height: '500px'
7 };
8
9 const center = {
10   lat: -34.397,
```

```
  lng: 150.644
};

const MyMapComponent: React.FC = () => {
  return (
    <LoadScript googleMapsApiKey="YOUR_API_KEY">
      <GoogleMap
        mapContainerStyle={containerStyle}
        center={center}
        zoom={8}
      >
        <Marker position={center} />
      </GoogleMap>
    </LoadScript>
  );
};

export default MyMapComponent;
```

Ukázka kódu 2.3 Ukázka integrace Google Mapy v aplikaci React s TypeScriptem

2.10.3 Popis příkladu

V tomto příkladu:

- `LoadScript` komponenta načítá Google Maps API pomocí vašeho API klíče.
- `GoogleMap` komponenta vykresluje samotnou mapu.
- `Marker` komponenta přidává marker na specifikovanou pozici.

Tato aplikace zobrazuje mapu s markerem na pozici definovanou v proměnné `center`.[14]

2.11 Dodatečné technologie

Projekt je spouštěn pomocí `Node.js` v kombinaci s `NPM`[14]. Celý kód je verzován verzovacím systémem `Git` a psán v textovém editoru `Visual Studio Code (VS Code)`.

2.11.1 package.json

Soubor `package.json` je klíčový konfigurační soubor pro projekty postavené na `Node.js`. Tento soubor spravuje závislosti, což jsou knihovny a balíčky, které projekt využívá. Kromě toho definuje skripty pro automatizaci úkolů, jako je spuštění serveru, testování nebo nasazení.

[14]Node Package Manager - správce balíčků pro JavaScript, standardně používán v ekosystému `Node.js`

Po naklonování projektu stačí spustit `npm i`, který stáhne všechny závislosti uvedené v souboru a nainstaluje je do složky `node_modules`[15]

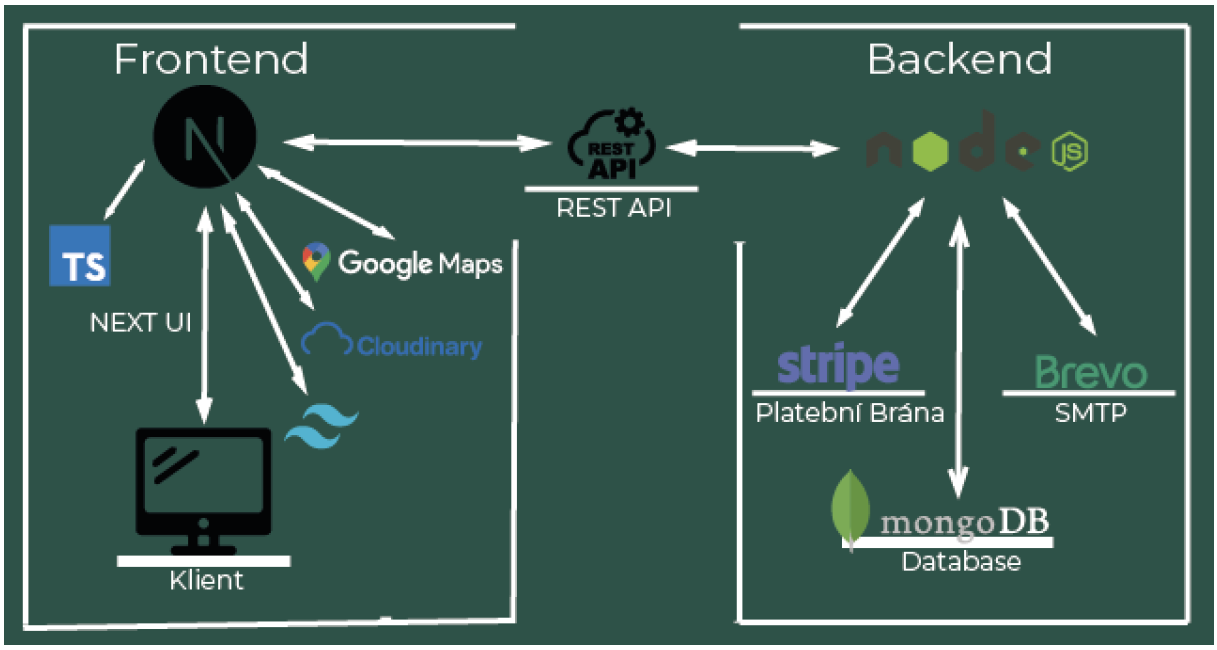
2.11.2 Příklad package.json

```
{
  "name": "chatky24",
  "version": "1.0.0",
  "description": "A sample Node.js project",
  "main": "index.js",
  "scripts": {
    "start": "node index.js",
    "dev": "nodemon index.js",
    "test": "jest"
  },
  "dependencies": {
    "express": "^4.17.1",
    "mongoose": "^5.10.9",
    "dotenv": "^8.2.0"
  },
  "devDependencies": {
    "nodemon": "^2.0.4",
    "jest": "^26.6.3"
  },
  "author": "Your Name",
  "license": "MIT"
}
```

Ukázka kódu 2.4 Příklad souboru `package.json`

3. Chatky24

3.1 Popis architektury



Obrázek 3.1: Architektura systému Chatky24

Tento obrázek zobrazuje architekturu systému Chatky24, která je rozdělena na backendovou a frontendovou část.

3.1.1 Backend

- **Node.js (JS)**: Backend je postavený na prostředí Node.js.
- **REST API**: Frontend komunikuje s backendem pomocí REST API.
- **Stripe**: Pro zpracování plateb je integrována platební brána Stripe.
- **Brevo**: Pro odesílání e-mailů je použita služba Brevo (SMTP).
- **MongoDB**: Data jsou uložena v databázi MongoDB.

3.1.2 Frontend

- **Next.js (N)**: Frontend je postavený na frameworku Next.js, který je založený na Reactu.

- **TypeScript (TS)**: Pro vývoj frontendu je použit TypeScript, který zajišťuje statické typování a lepší bezpečnost kódu.
- **Next UI**: Pro tvorbu uživatelského rozhraní je použita knihovna Next UI.
- **Google Maps**: Pro integraci mapových funkcí je použito Google Maps API.
- **Cloudinary**: Pro ukládání obrázků je použita služba Cloudinary.
- **Klient**: V levém dolním rohu je zobrazeno zařízení klienta (např. počítač, mobilní telefon), které komunikuje s frontendem.

3.2 Backend

V aplikaci Chatky24 roli backendu plní Node.js a Express.js. Frontend komunikuje s backendem pomocí REST API. Express.js poskytuje robustní a flexibilní backendové řešení, které umožňuje rychlou odezvu API, efektivní správu dat v MongoDB a bezpečné zpracování uživatelských požadavků.

3.2.1 MongoDB databáze

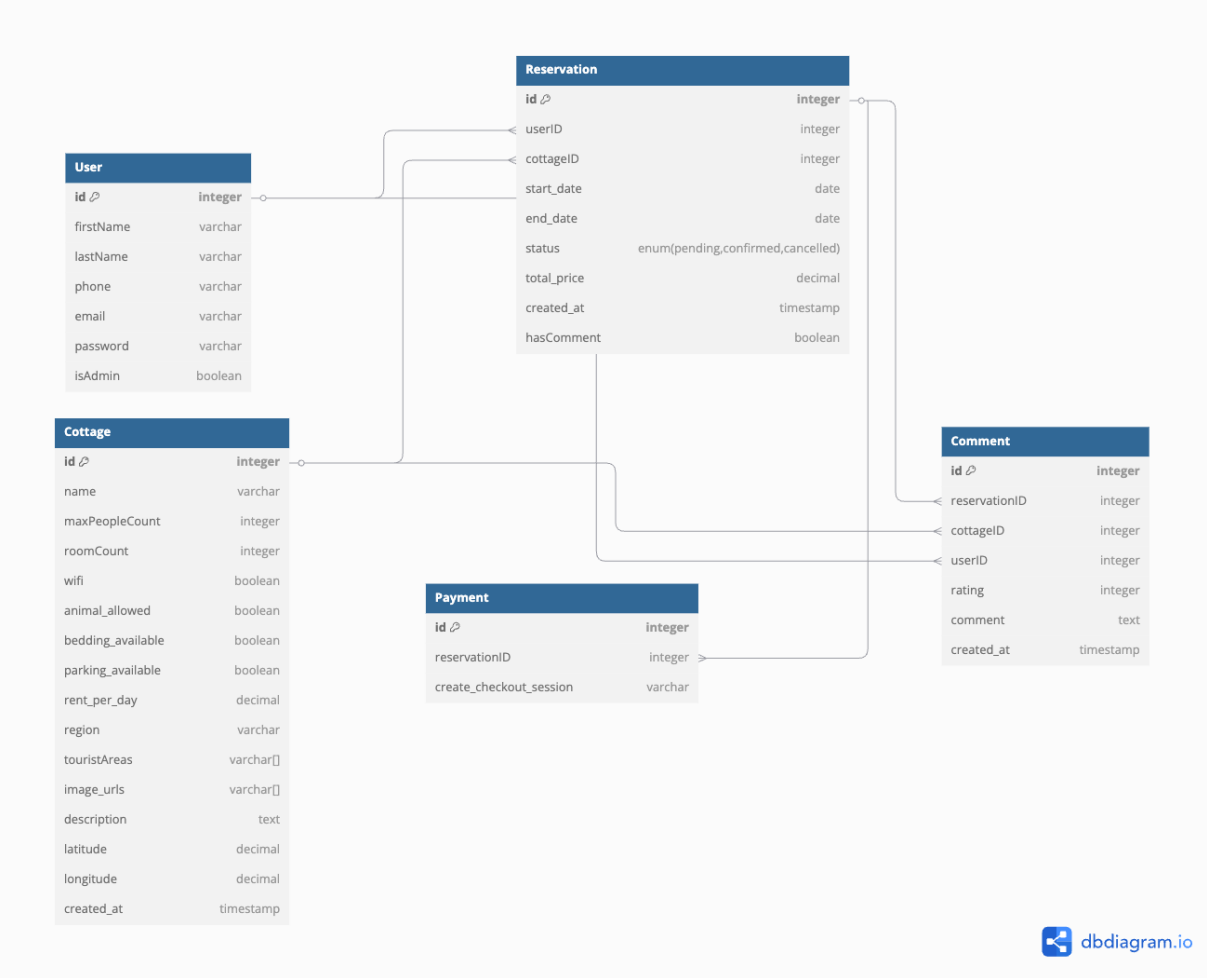
MongoDB je flexibilní dokumentová databáze, která se vyznačuje snadným použitím, vysokou škálovatelností a efektivní prací s nestrukturovanými daty. Pro projekt Chatky24, zaměřený na rezervaci chat a chalup, se ukázala jako ideální volba z několika důvodů. Díky své schopnosti dynamicky spravovat data umožňuje snadnou úpravu struktury databáze bez nutnosti složitých migrací. Podporuje rychlé vyhledávání a filtrování nabídek podle různých parametrů, což je klíčové pro uživatele hledající ideální ubytování. Navíc její distribuce a horizontální škálování zajišťují, že aplikace zvládne i vysokou zátěž v obdobích zvýšené poptávky.

3.2.1.1 Proč právě MongoDB

- **MongoDB**: Pro databázi je zvolena MongoDB, která umožňuje flexibilní a dynamickou strukturu pro ukládání dat v dokumentech ve formátu JSON.
- **Škálovatelnost a výkon**: MongoDB poskytuje vysoký výkon při vyhledávání a filtrování dat, což je ideální pro systém rezervací a vyhledávání chat.
- **Cloudová správa**: MongoDB umožňuje snadné nasazení v cloudu s automatickou replikací a zálohováním, což zajišťuje spolehlivost systému.
- **Flexibilita struktury**: MongoDB umožňuje snadné přizpůsobení struktury databáze bez nutnosti složitých migrací, což je výhodné pro dynamické požadavky aplikace.

- **Osobní zkušenosti**: I když jsem s MongoDB dříve nepracoval, její vlastnosti mi přišly jako nejlepší volba pro systém správy chat a rezervací.

3.2.1.2 Návrh databáze



Obrázek 3.2: Schéma Databáze

- **Uživatelé a chaty**: Databáze propojuje uživatele s nabídkou chat a chalup. Uživatelé si mohou prohlížet chaty s detailními informacemi o vybavení a lokaci.
- **Rezervační systém**: Klíčovou funkcí je systém pro rezervaci chat, který zahrnuje sledování dostupnosti, stavu rezervace a celkové ceny.
- **Platební integrace**: Databáze obsahuje tabulku "Payment"pro záznam plateb, což naznačuje integraci s platební bránou pro online platby.
- **Hodnocení a komentáře**: Systém umožňuje uživatelům hodnotit a komentovat chaty, což poskytuje důležitou zpětnou vazbu pro ostatní uživatele.

- **Geolokace**: Databáze obsahuje GPS souřadnice pro každou chatu.
- **Filtrování chat**: Databáze umožňuje filtrovat chaty podle různých kritérií, jako je vybavení, kapacita nebo lokace.
- **Historie rezervací a plateb**: Systém uchovává historii rezervací a plateb, což umožňuje sledovat trendy a analyzovat data.

Databázový návrh aplikace Chatky24 je pečlivě koncipován tak, aby efektivně podporoval správu uživatelských účtů, rezervací a interakcí s nabídkami chat a chalup. Struktura byla navržena s důrazem na výkon, škálovatelnost a snadné vyhledávání ubytování podle různých parametrů. Díky flexibilitě MongoDB lze data dynamicky upravovat a optimalizovat tak, aby aplikace zvládla rostoucí počet uživatelů i rezervací bez ztráty rychlosti nebo efektivity.

3.2.2 Server

Backend aplikace je postaven na platformě Node.js, která poskytuje spolehlivou a bezpečnou infrastrukturu pro zpracování požadavků od klienta. Architektura je navržena s cílem maximální efektivity, jednoduchosti a bezpečnosti, přičemž se vyhýbám použití zbytečných knihoven třetích stran. Node.js je ideální pro asynchronní operace a umožňuje rychlé zpracování požadavků, což je důležité pro aplikace s vysokými nároky na výkon.

3.2.2.1 Zabezpečení API pomocí JWT

- **Autentizace a autorizace pomocí JWT**: Pro zabezpečení přístupu k databázi a API je implementována autentizace a autorizace pomocí JWT (JSON Web Tokens).
- **Přihlašování uživatelů**:
  - Uživatelé se ověřují pomocí e-mailu a hesla.
  - Hesla jsou uložena v databázi hashovaná (s použitím saltu).
  - Po úspěšném ověření je vygenerován JWT token obsahující:
    * ID uživatele
    * E-mail uživatele
    * Roli uživatele (isAdmin)
  - Token je podepsán a odeslán klientovi.
  - Server ověřuje platnost tokenu při každém požadavku.

- **Výhody**:
  - Zvýšená bezpečnost díky digitálnímu podpisu a hashování hesel.
  - Bezstavová autentizace, která usnadňuje škálování aplikace.
  - Možnost přesného řízení přístupu k datům na základě rolí uživatelů.
  - Nezávislost na externích autentizačních službách.

3.2.2.2 Struktura API endpointů

- **REST API Endpointy**: Server nabízí REST API s přehledně navrženými endpointy, které umožňují efektivní komunikaci s klientskou aplikací.
- **Autorizace a autentifikace uživatelů**:
  - `/api/auth/register`: Endpoint pro registraci uživatelů.
  - `/api/auth/login`: Endpoint pro přihlášení uživatelů.
  - `/api/auth/logout`: Endpoint pro odhlášení uživatele.
  - `/api/auth/reset-password`: Endpoint obnovení hesla.
  - `/api/auth/send-reset-email`: Endpoint pro zasílání emailu pro obnovení.
- **Komentáře**:
  - `/comments`: Načte všechny komentáře.
  - `/comments`: Přidá nový komentář.
  - `/comments/:cottageID`: Získá všechny komentáře k dané chatě.
  - `/comments/:id`: Smaže komentář (pouze pro administrátory).
- **Chaty**:
  - `/cottages`: Načte seznam všech chat.
  - `/cottages`: Vytvoří novou chatu.
  - `/cottages/:id`: Získá detailní informace o konkrétní chatě.
  - `/cottages/:id`: Smaže chatu (pouze pro administrátory).
  - `/cottages/filter`: Aplikuje filtry pro vyhledávání chat.
  - `/cottages/:id`: Upraví informace o chatě.
  - `/send-email`: Odeslání e-mailu (např. dotazy na chatu).

- **Rezervace**:
  - `/`: Vytvoří novou rezervaci.
  - `/`: Načte všechny rezervace uživatele.
  - `/admin`: Načte všechny rezervace (pouze pro administrátory).
  - `/admin/:id/status`: Aktualizuje stav rezervace (pouze pro administrátory).
  - `/admin/:id`: Smaže rezervaci.
  - `/user/:id`: Upraví rezervaci uživatele.
  - `/cottage/:id`: Získá rezervace pro konkrétní chatu.
  - `/confirmed`: Načte potvrzené rezervace uživatele.

- **Stripe**:
  - `/create-checkout-session`: Vytvoří platební relaci pro rezervaci.

3.2.3 Stripe pro platby

Pro zpracování online plateb v aplikaci *Chatky24* jsem se rozhodl využít službu **Stripe**, protože nabízí jednoduchou, ale robustní integraci pro bezpečné a efektivní platby. Stripe je ideální volbou, protože poskytuje širokou škálu funkcí, včetně správy rezervací a platby kartami, což je pro můj projekt nezbytné. Díky integraci s **Node.js** dokáže backend aplikace zpracovávat platební požadavky asynchronně, což zajišťuje rychlý a plynulý chod systému i při větším zatížení. Stripe navíc poskytuje silné nástroje pro zajištění bezpečnosti transakcí a prevenci podvodů, což pro mě znamená důvěryhodnost a ochranu uživatelů bez nutnosti vyvíjet složité platební procesy od začátku.

3.2.4 Odesílání emailů skrze Brevo (dříve Sendinblue)

Pro odesílání e-mailů v aplikaci *Chatky24* jsem zvolil službu **Brevo**. Rozhodl jsem se pro ni, protože nabízí spolehlivé a jednoduché řešení pro odesílání e-mailových notifikací, jako jsou potvrzení rezervací a další důležité zprávy pro uživatele. Služba je snadno integrovatelná do aplikace a její použití s **Node.js** umožňuje efektivní správu odesílání e-mailů bez nutnosti spravovat vlastní e-mailový server. Díky Brevo se mohu soustředit na samotnou funkčnost aplikace, aniž bych se musel starat o technické detaily správy e-mailového systému. Navíc Brevo poskytuje flexibilitu při tvorbě šablon a správě kampaní, což usnadňuje personalizaci a komunikaci s uživateli.

3.3 Frontend

3.3.1 Proč jsem zvolil Next.js

Framework **Next.js** postavený na Reactu jsem si pro svůj projekt vybral z několika klíčových důvodů:

- **Jednoduchost a intuitivnost** – Next.js poskytuje intuitivní prostředí pro vývoj i nasazení aplikace, což výrazně usnadňuje práci s kódem.
- **Jednodušší správa cookies** – Oproti čistému Reactu nabízí Next.js jednodušší práci s cookies, což usnadňuje správu uživatelských dat a autentizaci.
- **Efektivní routing** – Systém založený na souborové struktuře (`pages/`) umožňuje přehlednou organizaci a snadnou navigaci v kódu.
- **Optimalizace výkonu** – Next.js podporuje jak server-side rendering (SSR), tak client-side rendering (CSR), což umožňuje rychlé načítání statického obsahu i efektivní zpracování dynamických prvků podle potřeby.

3.3.2 Výběr TypeScriptu jako hlavního jazyk

Při vývoji frontendu jsem zvažoval, zda použít **JavaScript** nebo **TypeScript**. Po pečlivém zvážení jsem se přiklonil k TypeScriptu, protože poskytuje lepší strukturu kódu a umožňuje statické typování, což výrazně snižuje pravděpodobnost chyb. Díky tomu je kód čitelnější, lépe udržovatelný a předchází nečekaným problémům během vývoje. Dalším důvodem byla pozitivní zpětná vazba od zkušenějších vývojářů, kteří TypeScript doporučovali pro větší projekty. S touto volbou jsem naprosto spokojen a opět bych ji učinil.

3.3.3 Proč Tailwind

Tailwind CSS jsem vybral kvůli jeho **utility-first** přístupu, který umožňuje rychlé a efektivní stylování bez nutnosti psát rozsáhlé CSS soubory. Oproti Bootstrapu mi poskytuje **větší flexibilitu** a plnou kontrolu nad designem bez omezení předdefinovaných stylů. Další výhodou je **menší velikost výsledného CSS**, protože Tailwind zahrne pouze použité třídy, což zlepšuje výkon aplikace. Díky těmto vlastnostem se ukázal jako ideální volba pro tento projekt.

3.3.4 Proč Cloudinary

Pro aplikaci jsem zvolil **Cloudinary**, protože nabízí **automatické škálování a optimalizaci obrázků**, což je ideální pro responzivní design. Díky tomu se obrázky rychle načítají v různých formátech pro různá zařízení bez zatížení výkonu. Cloudinary také umožňuje **jednoduchou integraci**, což šetří čas a zjednodušuje vývoj.

3.3.5 Proč Google Maps API

Mapová integrace vyžadovala **spolehlivé a přesné řešení**, které umožní uživatelům snadno najít a zobrazit jednotlivé chaty. Google Maps API nabízí **detailní mapové podklady**, **globální pokrytí** a **možnost přizpůsobení**, což umožňuje přehledné zobrazení lokalit přímo v aplikaci. Díky **dobré dokumentaci** a jednoduché implementaci bylo možné rychle nasadit mapové funkce bez zbytečných komplikací. Google Maps API tak poskytlo ideální kombinaci **kvality, dostupnosti a jednoduchosti integrace**.

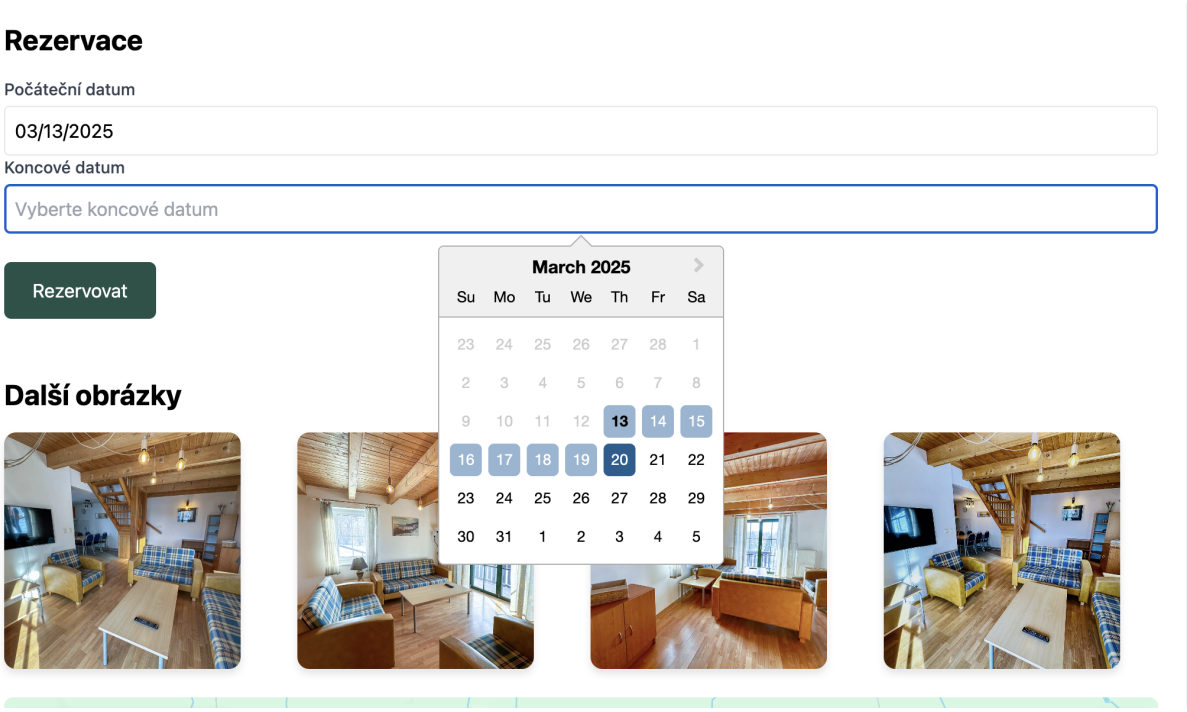
3.4 Výsledná aplikace

Aplikace **Chatky24** splňuje všechny stanovené cíle a poskytuje moderní, intuitivní a uživatelsky přívětivý systém pro rezervaci chat a chalup. Kromě základní funkcionality rezervací obsahuje i řadu dalších **užitečných funkcí**, které vylepšují uživatelský komfort a celkový chod platformy. Mezi hlavní vlastnosti aplikace patří:

- **Přehledný a precizní design**, který zajišťuje snadné ovládání a příjemnou uživatelskou zkušenost.
- **Přihlášení a registrace uživatelů**, včetně možnosti obnovení hesla.
- **Možnost rezervace chat**, včetně správy termínů a dostupnosti.
- **Detailní zobrazení všech chat** s informacemi o lokalitě, vybavení a fotografiemi.
- **Implementace platební brány**, umožňující bezpečné a rychlé platby za rezervace.
- **Přidávání komentářů** a hodnocení po skončení pobytu pro zpětnou vazbu.
- **Správa rezervací, chat a komentářů** jak pro běžné uživatele, tak pro administrátory.
- **Automatizovaný emailing** pro potvrzení rezervací, notifikace a další důležité zprávy.
- **Responzivní design**, optimalizovaný pro různá zařízení, včetně mobilních telefonů a tabletů.

Aplikace **Chatky24** tak nabízí komplexní a efektivní řešení pro uživatele, kteří hledají jednoduchý a spolehlivý systém správy rekreačních pobytů.

3.4.1 Rezervační systém



Obrázek 3.3: Obrázek Rezervačního systému

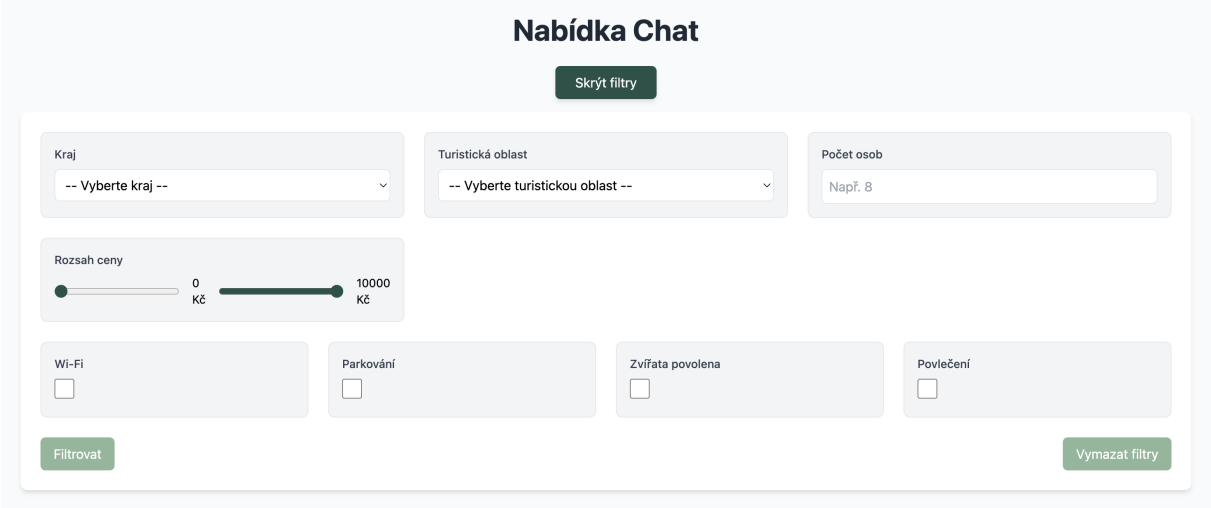
Rezervační systém je navržen pro snadné a intuitivní plánování pobytů. Uživatel si jednoduše vybere počáteční a koncové datum pomocí interaktivního kalendáře, který zobrazuje dostupné termíny. Po výběru stačí kliknout na tlačítko „Rezervovat“ a pokračovat k dokončení rezervace a platbě. Systém klade důraz na přehlednost, uživatelskou přívětivost a rychlý průběh rezervace, aby uživatelé mohli pohodlně a bez komplikací zajistit svůj pobyt.

3.4.2 Hledání ideální chaty

Systém vyhledávání chat umožňuje uživatelům snadno najít ideální ubytování díky kombinaci filtrování a řazení.

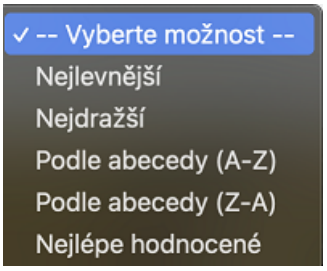
3.4.2.1 Filtrování

Uživatelé mohou vyhledávat chaty podle klíčových kritérií:



Obrázek 3.4: Obrázek Filtrovacího systému

- **Lokalita** – kraj a turistická oblast
- **Kapacita** – počet osob
- **Cena** – cenové rozpětí
- **Vybavení** – Wi-Fi, parkování, domácí mazlíčci, povlečení



Obrázek 3.5: Obrázek Řadícího systému

Po aplikaci filtrů lze výsledky seřadit podle:

- **Ceny** – od nejlevnějších po nejdražší a naopak
- **Abecedy** – podle názvu chaty vzestupně či sestupně
- **Hodnocení** – od nejlépe po nejhůře hodnocené

Filtrování i řazení jsou intuitivní a snadno dostupné, což zjednodušuje celý proces vyhledávání.



Obrázek 3.6: Obrázek Přihlášeného navbaru

3.4.3 Navbary

3.4.3.1 Přihlášen

Na obrázku je zobrazen přihlášený navbar aplikace Chatky24, kde je vidět uživatelský účet. To naznačuje, že uživatel je přihlášen a může přistupovat k funkcím aplikace, jako je správa profilu nebo vyhledávání chatů.

3.4.3.2 Nepřihlášen



Obrázek 3.7: Obrázek Nepřihlášeného navbaru

Na obrázku je zobrazen nepřihlášený navbar aplikace Chatky24, kde je hlavní akcí „Přihlášení“. To ukazuje, že uživatel není přihlášen a musí se nejprve přihlásit, aby mohl využívat funkce aplikace, jako je vyhledávání chatů podle různých kritérií, která jsou popsána v třetím obrázku.

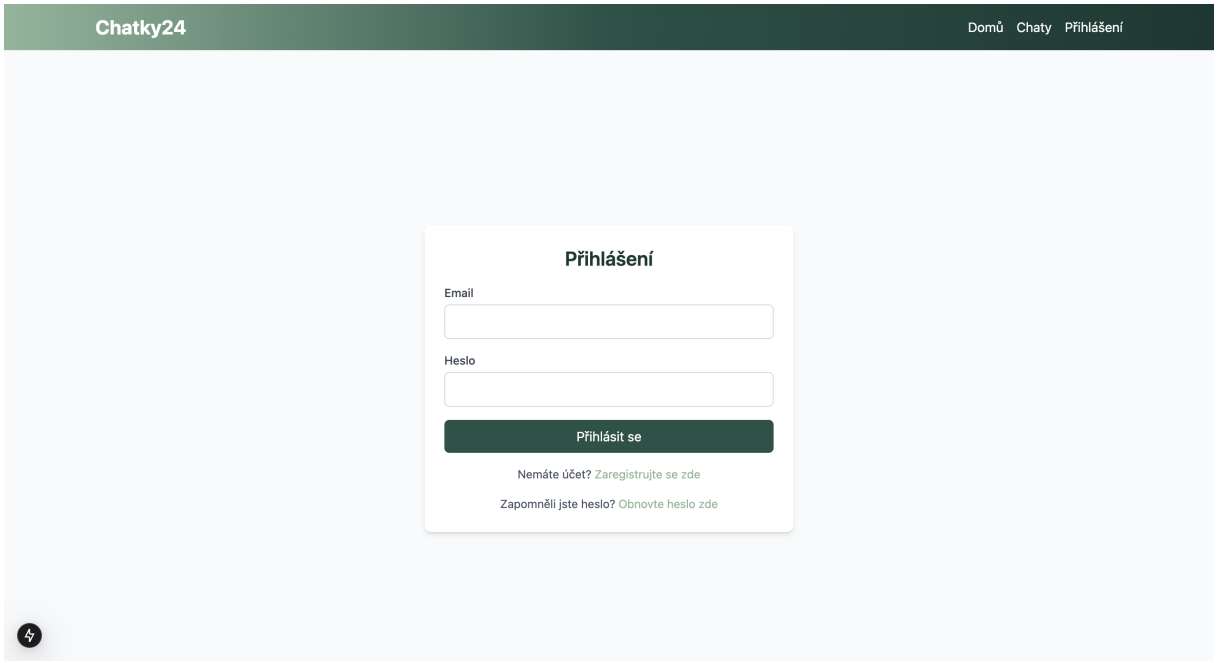
3.4.4 Přihlášení/Registrace

3.4.4.1 Stránka pro přihlášení

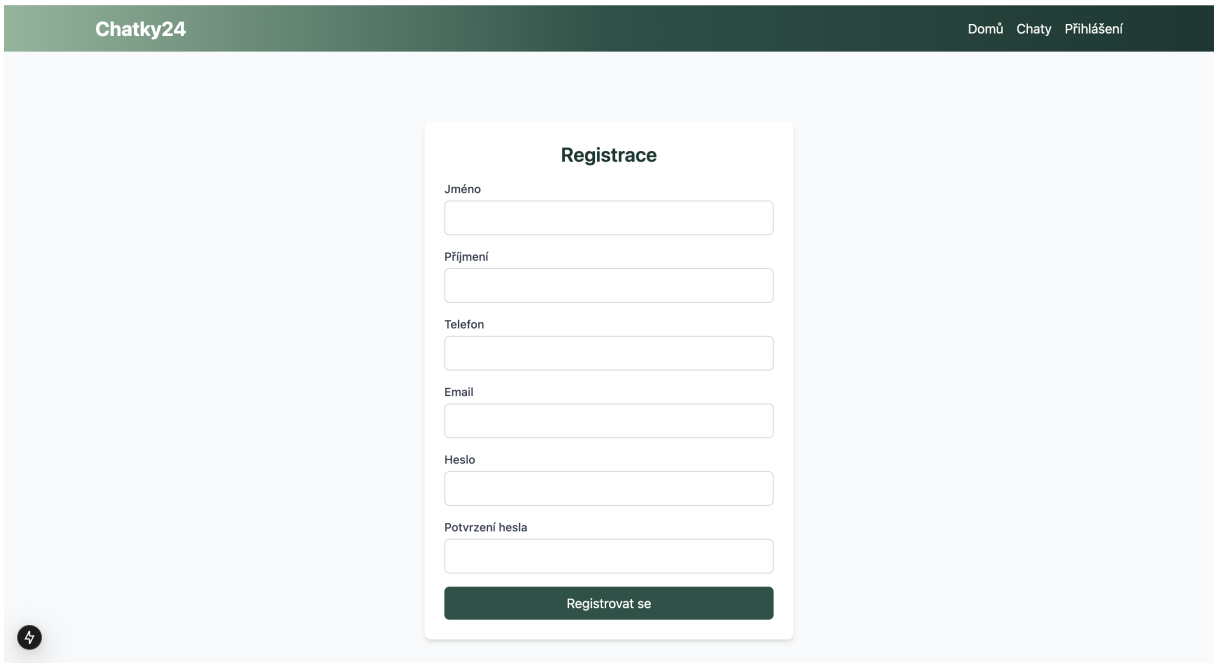
První obrázek ukazuje přihlašovací formulář (Login). Uživatelé zde mohou zadat svůj email a heslo, které je vždy hashováno pro zajištění bezpečnosti. Po vyplnění těchto údajů se mohou přihlásit pomocí tlačítka „Přihlásit se“. Pro uživatele, kteří ještě nemají účet, je k dispozici odkaz „Zaregistrujte se zde“, který je přesměruje na registrační formulář. V případě, že uživatel zapomněl heslo, může využít odkaz „Obnovte heslo zde“ pro jeho obnovení.

3.4.4.2 Stránka pro registraci

Druhý obrázek zobrazuje registrační formulář (Register). Zde si uživatelé mohou vytvořit nový účet vyplněním svého jména, příjmení, telefonního čísla, emailu a hesla. Heslo je nutné potvrdit pro ověření správnosti zadání. Po vyplnění všech požadovaných údajů se



Obrázek 3.8: Obrázek Login stránky

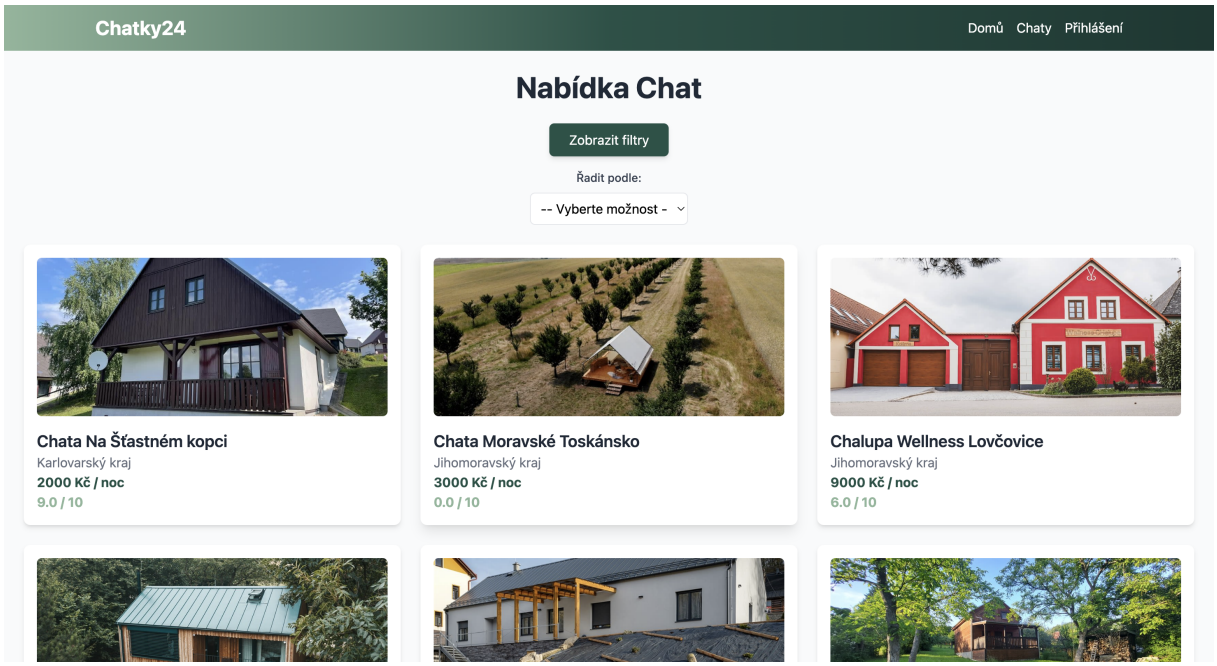


Obrázek 3.9: Obrázek Register stránky

uživatelé mohou zaregistrovat pomocí tlačítka „Registrovat se“. I zde jsou hesla hashována, aby byla zajištěna bezpečnost uživatelských dat.

3.4.5 Chaty

3.4.5.1 Nabídka chat



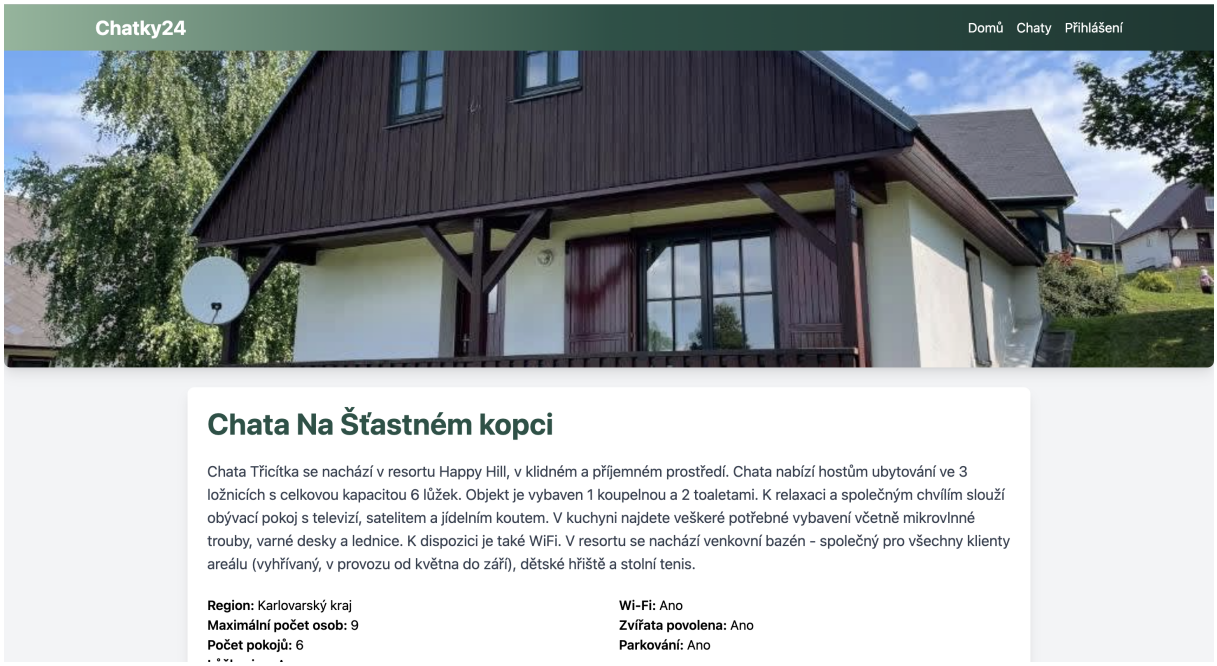
Obrázek 3.10: Obrázek Nabídky chat

Na obrázku je vidět "seznam chatů"v aplikaci Chatky24. Každá chata je zobrazena s několika základními informacemi, které uživatelům pomáhají při výběru ubytování.

Jako příklad je zde "Chata Na Šťastném kopci", která se nachází v Karlovarském kraji. Cena za noc je 2000 Kč a chata má hodnocení 9.0 z 10. Tento stručný popis obsahuje lokaci, cenu a hodnocení, což uživatelům poskytuje rychlý přehled o nabízeném ubytování.

Podobně jsou zobrazeny i další chaty, jako "Chata Moravské Toskánsko"a "Chalupa Wellness Lovčovice", každá s informacemi o lokaci, ceně a hodnocení. Tento seznam umožňuje uživatelům snadno porovnat různé možnosti a vybrat si chatu, která nejlépe vyhovuje jejich potřebám.

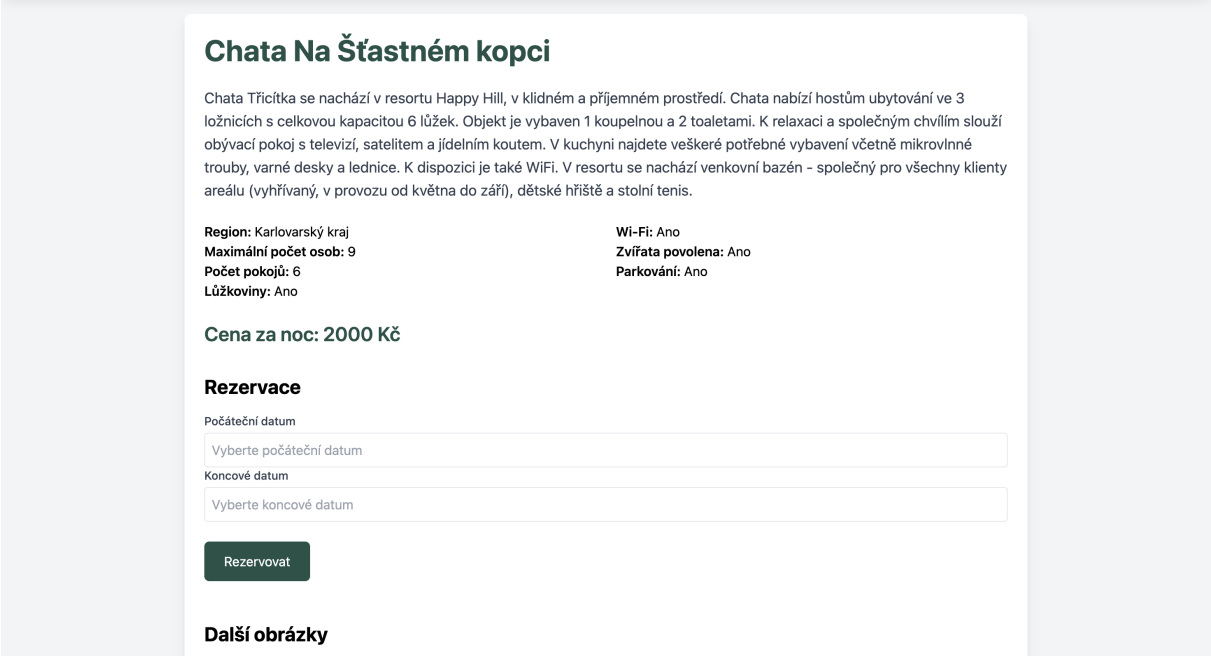
3.4.5.2 Detaily Chaty



Obrázek 3.11: Obrázek Detailu chaty

Na obrázku je zobrazen detail chaty a možnosti rezervace v aplikaci Chatky24. Chata s názvem Chata Na Šťastném kopci se nachází v resortu Happy Hill v Karlovarském kraji. Nabízí ubytování pro až 9 osob v 6 pokojích s 6 lůžky. Chata má 3 ložnice, 1 koupelnu a 2 toalety. K vybavení patří obývací pokoj s televizí a satelitem, plně vybavená kuchyně a Wi-Fi. V resortu je k dispozici venkovní vyhřívaný bazén (v provozu od května do září), dětské hřiště a stolní tenis. Chata povoluje domácí mazlíčky a nabízí parkování.

3.4.5.3 Cena za chatu



Obrázek 3.12: Obrázek Detailu chaty 2

Na obrázku je zobrazena sekce "Rezervace"pro chatu "Chata Na Šťastném kopci". Uživatelé mohou vybrat počáteční a koncové datum pobytu, aby si rezervovali chatu na požadované období. Cena za noc je 2000 Kč.

3.4.5.4 Obrázky



Obrázek 3.13: Obrázek Detailu chaty 3

Kromě toho jsou k dispozici "další obrázky chaty", které poskytují detailní vizuální představu o ubytování a jeho vybavení. Tyto fotky pomáhají uživatelům lépe si představit, jak chata vypadá a co všechno nabízí.

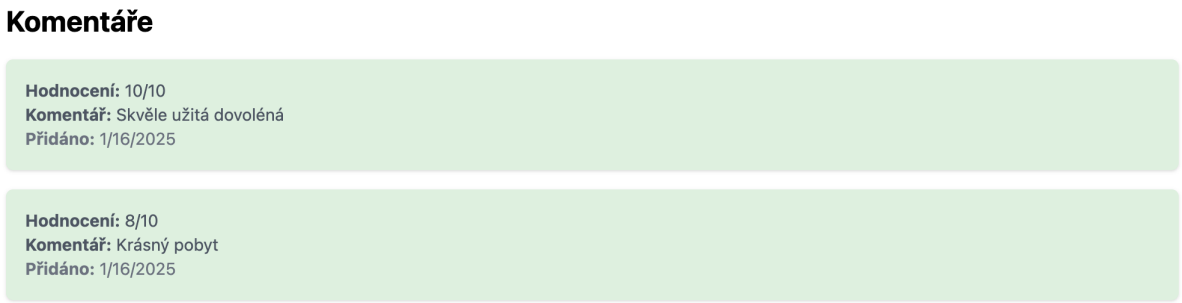
3.4.5.5 Mapa



Obrázek 3.14: Obrázek Detailu chaty 4

Na obrázku je zobrazena mapa s lokalitou chaty, včetně satelitního pohledu a okolních míst.

3.4.5.6 Komentáře

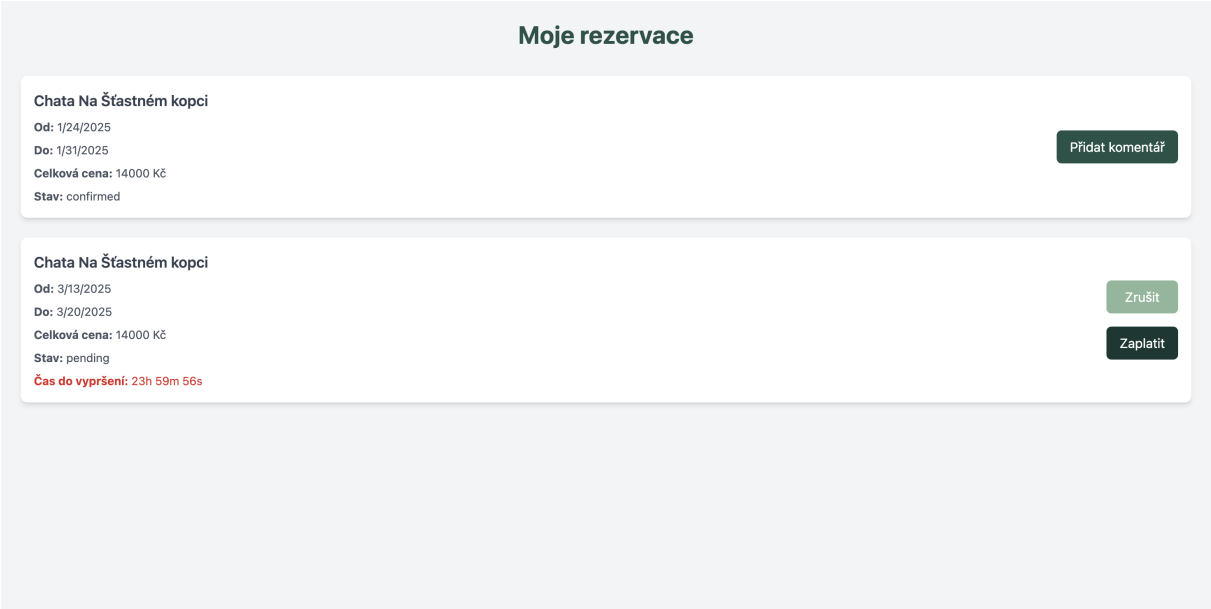


Obrázek 3.15: Obrázek Detailu chaty 5

Na obrázku je zobrazena sekce "Komentáře"k chatě "Chata Na Šťastném kopci". Uživatelé zde najdou hodnocení a komentáře od předchozích hostů. Jeden host ohodnotil chatu 10/10 a chválil skvěle strávenou dovolenou. Další host dal hodnocení 8/10 a pobyt označil jako krásný. Oba komentáře byly přidány 16. ledna 2025.

3.4.6 Rezervace

3.4.6.1 Moje rezervace

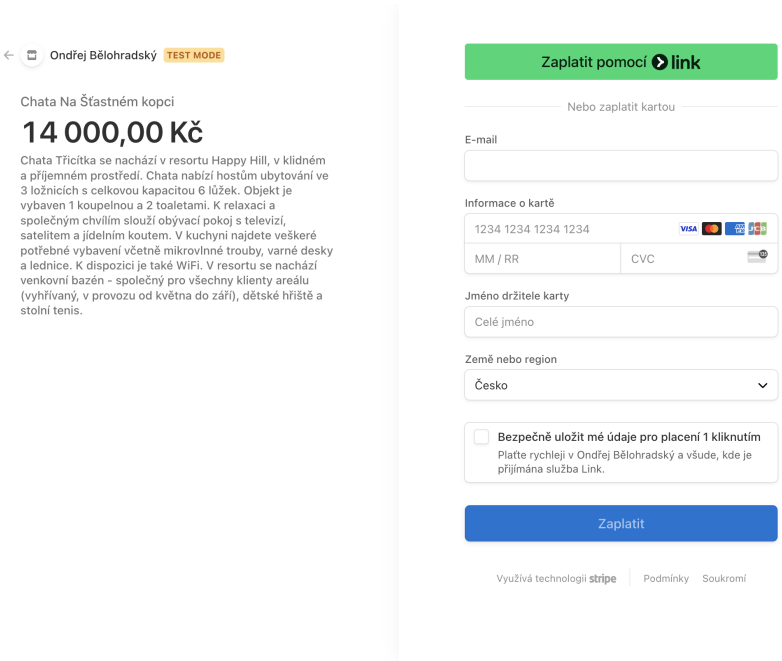


Obrázek 3.16: Obrázek Moje rezervace

Na obrázku je zobrazen přehled uživatelových rezervací. Vidíme zde dvě rezervace chaty *"Na Šťastném kopci"* – jednu již potvrzenou na leden a druhou čekající na březen.

Hned po vytvoření rezervace má uživatel možnost ji přímo z tohoto přehledu **zaplatit** nebo **zrušit**. Po uplynutí termínu pobytu se pak zobrazí možnost přidat ke své rezervaci *hodnotící komentář*. Toto rozhraní slouží jako praktický nástroj pro správu všech rezervací na jednom místě a zároveň umožňuje uživatelům sdílet své zkušenosti po skončení pobytu.

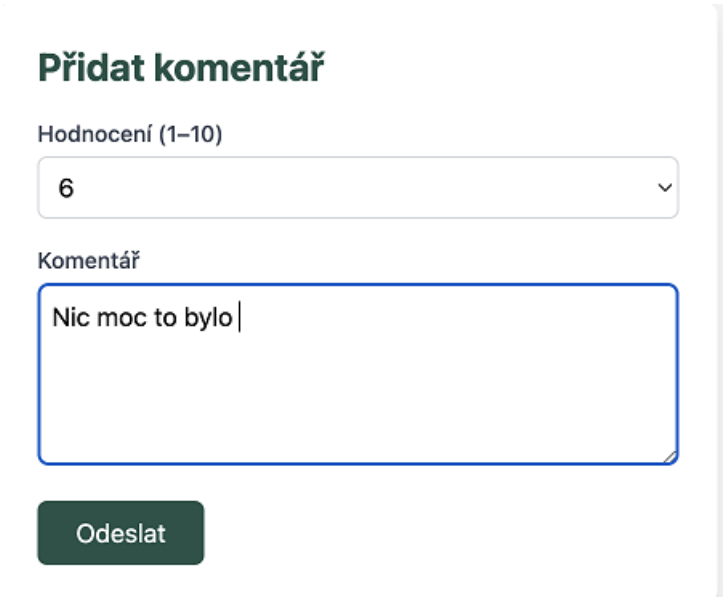
3.4.6.2 Platba



Obrázek 3.17: Obrázek Platební brány

Stránka je rozdělena na dvě části. Levá obsahuje popis chaty a cenu, pravá slouží k platbě. Uživatel může zaplatit přes Link nebo zadat údaje karty manuálně. Pokud zadá správné údaje, platba proběhne. V případě chyby musí údaje opravit. Možnost uložení platebních údajů urychluje budoucí transakce. Platba probíhá přes zabezpečenou bránu Stripe.

3.4.6.3 Přidání komentáře

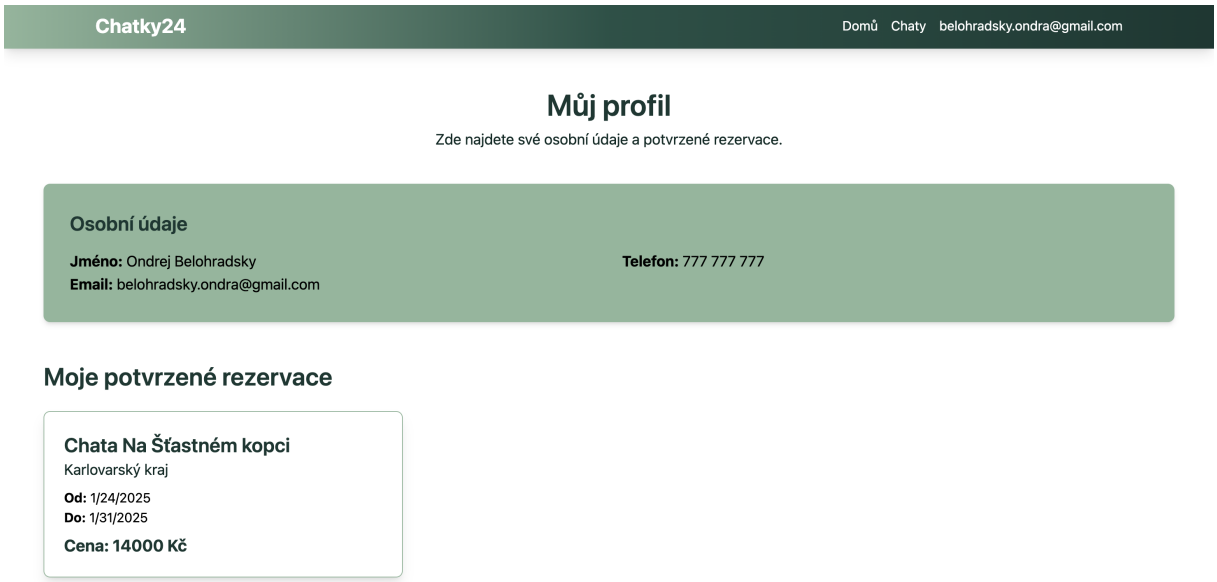


Obrázek 3.18: Obrázek Přidání komentáře

Uživatelé můžou ohodnotit pobyt v chatě pomocí škály **1–10** a napsat vlastní komentář. Po vyplnění mohou recenzi **odeslat** a podělit se tak o své zkušenosti s ubytováním.

3.4.7 Profil

3.4.7.1 Základní údaje



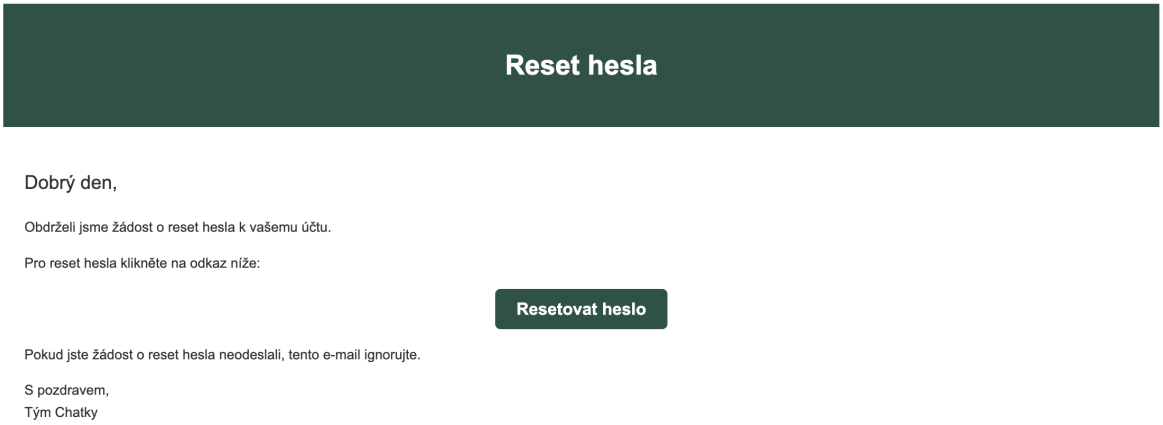
Obrázek 3.19: Obrázek Profilu Uživatele

V aplikaci je i **osobní profil uživatele**, kde jsou zobrazeny jeho základní informace a přehled potvrzených rezervací. Uživatel zde může snadno **spravovat své rezervace** a **aktualizovat kontaktní údaje**.

Dále je zde i přehledové rozhraní pro správu rezervací. Profilová sekce umožňuje uživatelům kontrolovat své **zaplacené a potvrzené rezervace**.

3.4.8 Email komunikace

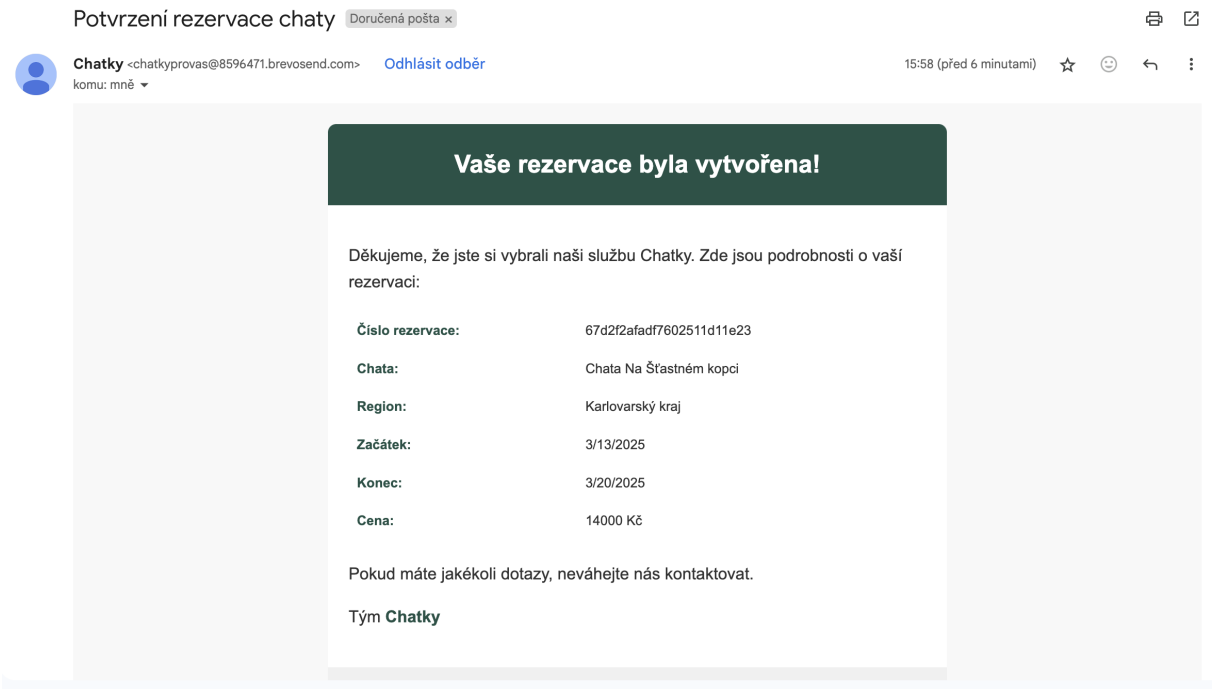
3.4.8.1 Obnova zapomenutého hesla



Obrázek 3.20: Obrázek Resetu hesla

Uživatel obdrží po žádosti o reset hesla. Systém automaticky zasílá tento e-mail na zadanou adresu s odkazem pro obnovu přístupu. Uživatel jednoduše klikne na tlačítko **"Resetovat heslo"** a může si nastavit nové heslo. Pokud žádost neinicioval, e-mail může bezpečně ignorovat – tato funkce zajišťuje bezpečnost účtu.

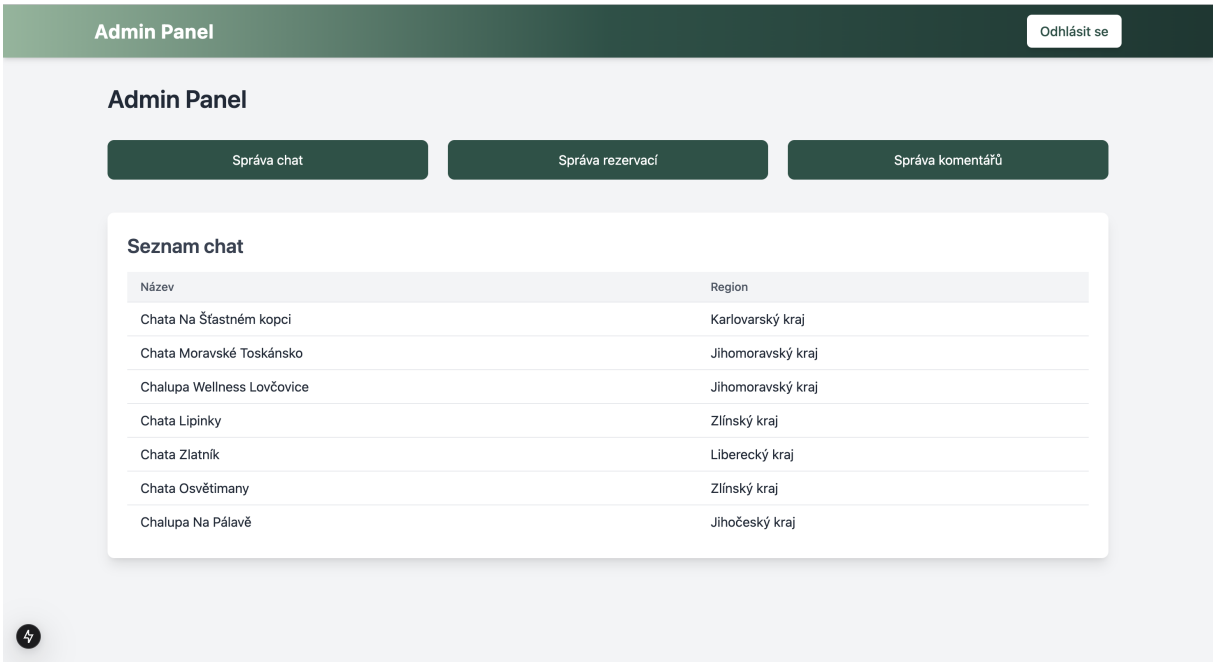
3.4.8.2 Potvrzovací e-mail o rezervaci



Obrázek 3.21: Obrázek Emailu o potvrzení chaty

Potvrzovací e-mail zasílaný uživateli po úspěšném vytvoření rezervace. Systém automaticky generuje tuto zprávu jako potvrzení o přijetí rezervace a shrnutí klíčových informací. E-mail slouží jako **oficiální doklad o rezervaci** a zároveň informuje uživatele o dalších krocích. Součástí je i nabídka pomoci v případě dotazů, což **zlepšuje celkovou uživatelskou zkušenost**.

3.4.9 Administrátor



Obrázek 3.22: Obrázek Admin stránky

Administrátorské rozhraní, které slouží k centrální správě systému. Administrátor zde může provádět kompletní správu chat, včetně **vytváření, úprav a mazání (CRUD operace)**. U rezervací má možnost **ručně aktualizovat jejich stav** a u uživatelských komentářů je k dispozici **funkce mazání nevhodných recenzí**. Panel nabízí **přehledný seznam registrovaných chat** s jejich základními údaji, což usnadňuje správu a aktualizaci nabídky ubytování.

4. Budoucnost

4.1 Budoucnost aplikace Chatky24

4.1.1 Co v aplikaci chybí

Aplikace Chatky24 nabízí širokou škálu funkcí pro rezervaci chat a chalup, ale stále existují oblasti, které je potřeba zlepšit a doplnit. Níže jsou uvedeny některé z hlavních nedostatků, na kterých je třeba dodelat.

4.1.1.1 Zabezpečení

I když aplikace využívá moderní technologie, stále existují některé bezpečnostní nedostatky, na kterých je potřeba zapracovat:

- **ID uživatele v localStorage:** Ukládání ID uživatele v localStorage není bezpečné, protože toto ID může být snadno přístupné a zneužitelné. Je třeba implementovat bezpečnější způsoby ukládání citlivých dat, například pomocí šifrování nebo alternativních metod.
- **Omezená ochrana proti brute-force útokům:** Momentálně chybí omezení počtu neúspěšných pokusů o přihlášení. To by mělo být implementováno, aby se zamezilo útokům typu brute-force.
- **Chybějící Content Security Policy (CSP):** Aplikace nemá nastavenu CSP, což může zvyšovat riziko XSS útoků. Zavedení této politiky by pomohlo chránit aplikaci před potenciálními bezpečnostními hrozbami.

4.1.1.2 Chybějící funkce a vylepšení

Aplikace by mohla nabídnout další užitečné funkce, které by zvýšily její atraktivitu a použitelnost pro širší okruh uživatelů:

- **Více jazykových verzí:** V současnosti aplikace podporuje pouze jeden jazyk. Zavedení více jazykových verzí by umožnilo širší přístup k aplikaci pro uživatele z různých zemí.
- **Tmavý režim:** Tmavý režim je stále velmi populární a zlepšuje uživatelský zážitek, zejména při používání aplikace v noci nebo v prostředích s nízkým osvětlením.

– **Pokročilejší filtrace:** V současnosti jsou možnosti filtrace rezervací a chat omezené. Uživatelé by ocenili širší možnosti filtrování podle dalších parametrů.

4.1.1.3 Optimalizace

Aplikace trpí i problémy s výkonem, které je nutné řešit, aby se zlepšil uživatelský zážitek:

– **Neefektivní načítání stránky:** Aplikace při prvním spuštění načítá více dat, než je nezbytné, což zpomaluje její start. Bylo by vhodné implementovat postupné načítání obsahu (*lazy loading*) a minimalizovat množství požadavků na server.
– **Absence efektivního kešování:** Aplikace nevyužívá dostatečně kešování API odpovědí, což vede k opakovanému dotazování databáze i tam, kde by bylo možné využít uložená data. Zavedení strategického kešování na straně klienta i serveru by mohlo zlepšit výkon.

4.1.2 Plány do budoucna

Rád bych vylepšil aplikaci o výše zmíněné funkce a opravil bezpečnostní problémy. Po dokončení těchto úprav plánuju aplikaci zpřístupnit veřejně, aby ji mohl využívat kdokoliv, kdo hledá ideální chatu nebo chalupu k rezervaci. Dále bych rád přidal možnost tmavého režimu a vícejazyčné podpory, což by mohlo pomoci oslovit širší okruh uživatelů.

Aplikaci budu nadále rozvíjet jako součást svého portfolia, ať už zůstane zdarma dostupná pro všechny, nebo se postupně rozšíří o další funkce a vylepšení, která ji posunou ještě dál.

Závěr

Projekt Chatky24 byl od začátku navržen jako platforma pro snadnou a efektivní rezervaci chat a chalup. Cílem bylo vytvořit moderní, přehlednou aplikaci s intuitivním uživatelským rozhraním, bezpečnou autentizací a flexibilní správou rezervací. Během vývoje jsem se zaměřil na integraci klíčových funkcí, jako je správa uživatelských účtů, pokročilé filtrování, online platby a administrátorský panel pro správu chat a rezervací.

Při zpětném pohledu mohu říct, že aplikace splnila většinu mých původních očekávání. Uživatelé mají k dispozici všechny důležité nástroje pro pohodlnou rezervaci a správu svých pobytů, včetně automatizovaných notifikací a platebního systému. Administrátoři pak mohou efektivně spravovat nabídku ubytování, kontrolovat rezervace a moderovat uživatelské recenze.

I když byly implementovány různé bezpečnostní mechanismy, některé aspekty zabezpečení nejsou 100% garantovány. Současné řešení poskytuje solidní úroveň ochrany, ale pro dosažení naprosté bezpečnosti bude nutné provést další vylepšení.

Na druhou stranu aplikace předčila moje očekávání v oblasti uživatelské zkušenosti a flexibility. Integrace moderních technologií, jako je React, Node.js a MongoDB, umožnila vytvořit responzivní a svižné rozhraní, které lze snadno rozšiřovat o nové funkce. Díky tomu je aplikace připravena na další rozvoj, včetně podpory více jazyků nebo rozšíření možností filtrování.

Během vývoje jsem získal cenné zkušenosti v oblasti backendu i frontendu, práce s API, zabezpečení uživatelských dat a integrace platebních systémů. Tento projekt mi pomohl prohloubit znalosti v oblasti webového vývoje a dal mi pevný základ pro další aplikace.

Ačkoliv je Chatky24 již funkční a připravená k používání, vnímám ji jako otevřený projekt s potenciálem pro budoucí vylepšení. Rád bych v dalších fázích přidal nové funkce, optimalizoval výkon aplikace a rozšířil její možnosti na základě zpětné vazby uživatelů.

Celkově jsem s výsledkem spokojený. Chatky24 splnila svůj účel a věřím, že uživatelům poskytne pohodlný a efektivní způsob rezervace ubytování. Tento projekt mi zároveň umožnil ukázat své schopnosti v návrhu a realizaci komplexní webové aplikace a stal se důležitou součástí mého portfolia.

Literatura

[1] Booking.com B.V. *Booking.com Official Site*. 10. břez. 2024. URL: https://www.booking.com/.

[2] Inc. Airbnb. *Airbnb: Vacation Rentals, Cabins, Beach Houses & More*. 10. břez. 2024. URL: https://www.airbnb.com/.

[3] Czech-Cottages s.r.o. *Ubytování v soukromí – chaty a chalupy*. 10. břez. 2024. URL: https://www.czech-cottages.cz/.

[4] E-chalupy.cz. *Pronájem chat a chalup v ČR*. 10. břez. 2024. URL: https://www.e-chalupy.cz/.

[5] Open Source Initiative. *MIT License*. 10. břez. 2024. URL: https://opensource.org/license/mit/.

[6] Inc. Vercel. *Next.js Documentation*. 10. břez. 2024. URL: https://nextjs.org/docs.

[7] Yoni Goldberg. *Node.js Best Practices*. 10. břez. 2024. URL: https://github.com/goldbergyoni/nodebestpractices.

[8] Inc. MongoDB. *MongoDB for Developers*. 10. břez. 2024. URL: https://university.mongodb.com/.

[9] Microsoft. *TypeScript Handbook*. 10. břez. 2024. URL: https://www.typescriptlang.org/docs/handbook/.

[10] Sarah Dayan. *Why I Love Tailwind CSS*. 10. břez. 2024. URL: https://frontstuff.io/why-i-love-tailwind.

[11] Inc. Stripe. *Stripe Documentation*. 10. břez. 2024. URL: https://stripe.com/docs.

[12] Brevo (ex Sendinblue). *Transactional Email API*. 10. břez. 2024. URL: https://developers.brevo.com/docs.

[13] Cloudinary Ltd. *Image Optimization Guide*. 10. břez. 2024. URL: https://cloudinary.com/guides.

[14] Google LLC. *Maps JavaScript API*. 10. břez. 2024. URL: https://developers.google.com/maps/documentation/javascript.

[15] Inc. npm. *npm Documentation*. 10. břez. 2024. URL: https://docs.npmjs.com/.

Seznam ukázek kódu

2.1 Ukázka integrace Tailwindu v aplikaci React s TypeScriptem 8
2.2 Ukázka integrace Stripe v Node.js 9
2.3 Ukázka integrace Google Mapy v aplikaci React s TypeScriptem 11
2.4 Příklad souboru `package.json` 13

Seznam obrázků

3.1 Architektura systému Chatky24 14
3.2 Schéma Databáze 16
3.3 Obrázek Rezervačního systému 22
3.4 Obrázek Filtrovacího systému 23
3.5 Obrázek Řadícího systému 23
3.6 Obrázek Přihlášeného navbaru 24
3.7 Obrázek Nepřihlášeného navbaru 24
3.8 Obrázek Login stránky 25
3.9 Obrázek Register stránky 25
3.10 Obrázek Nabídky chat 26
3.11 Obrázek Detailu chaty 27
3.12 Obrázek Detailu chaty 2 28
3.13 Obrázek Detailu chaty 3 28
3.14 Obrázek Detailu chaty 4 29
3.15 Obrázek Detailu chaty 5 29
3.16 Obrázek Moje rezervace 30
3.17 Obrázek Platební brány 31
3.18 Obrázek Přidání komentáře 32
3.19 Obrázek Profilu Uživatele 33
3.20 Obrázek Resetu hesla 34
3.21 Obrázek Emailu o potvrzení chaty 35
3.22 Obrázek Admin stránky 36