

EKONOMICKÁ UNIVERZITA V BRATISLAVE

FAKULTA HOSPODÁRSKEJ INFORMATIKY

Evidenčné číslo: 103004/I/2025/36124048427810564

APLIKÁCIA VIRTUÁLNEJ RECEPCIE PRE HOTEL

Diplomová práca

EKONOMICKÁ UNIVERZITA V BRATISLAVE

FAKULTA HOSPODÁRSKEJ INFORMATIKY

APLIKÁCIA VIRTUÁLNEJ RECEPCIE PRE HOTEL

Diplomová práca

Študijný program: Informačný manažment

Študijný odbor: Ekonómia a manažment

Školiace stredisko: Katedra aplikovanej informatiky

Vedúci záverečnej práce: Ing. Pavol Sojka, PhD.

Bratislava 2025

Bc. David Hanulák

Čestné vyhlásenie

Čestne vyhlasujem, že záverečnú prácu som vypracoval samostatne a že som uviedol všetku použitú literatúru.

Dátum:

.....

podpis študenta

Pod'akovanie

Chcel by som sa pod'akovať môjmu vedúcemu práce, Ing. Pavlovi Sojkovi, PhD., za jeho cenné rady, trpezlivosť a odborné vedenie počas celej práce. Jeho znalosti a skúsenosti mi veľmi pomohli pri riešení problémov a vylepšovaní mojej práce.

ABSTRAKT

HANULÁK, David: Aplikácia virtuálnej recepcie pre hotel. – Ekonomická univerzita v Bratislave. Fakulta hospodárskej informatiky, Katedra aplikovanej informatiky. – Vedúci záverečnej práce: Ing. Pavol Sojka, PhD.– Bratislava: FHI EU, 2025, 61 s.

Digitalizácia hotelových služieb sa stáva kľúčovým faktorom pri zvyšovaní efektivity a zlepšovaní zákazníckej skúsenosti. Tradičné recepčné služby môžu byť pre hotelové prevádzky nákladné a menej flexibilné, čo vedie k zvýšenému dopytu po automatizovaných riešeniach. Táto práca sa zaoberá návrhom a vývojom aplikácie virtuálnej recepcie, ktorá využíva pokročilé jazykové modely umelej inteligencie, konkrétne GPT-4, na interaktívnu komunikáciu s hosťami.

Aplikácia umožňuje hosťom dve hlavné formy interakcie s touto recepciou: priame konverzácie s modelom GPT-4 v reálnom čase a automatizované generovanie sprievodcov s informáciami o hoteli, okolí a odporúčaniami. Medzi ďalšie kľúčové funkcie patrí online check-in a check-out, digitálne kľúče od ubytovania, či virtuálny podpis rezervácie bez nutnosti papierovej formy.

Výsledky testovania aplikácie potvrdzujú jej schopnosť zlepšiť používateľskú skúsenosť, znížiť záťaž na recepčný personál a zvýšiť efektívnosť hotelových procesov. Virtuálna recepcia s umelou inteligenciou predstavuje inovatívny krok smerom k plne digitálnemu hotelovému manažmentu.

Kľúčové slová:

virtuálna recepcia, hotelový manažment, GPT-4, umelá inteligencia, digitálny check-in, automatizácia

ABSTRAKT

HANULÁK, David: Virtual Reception Application for a Hotel. – University of Economics in Bratislava, Faculty of Economic Informatics, Department of Applied Informatics. – Thesis Supervisor: Ing. Pavol Sojka, PhD. – Bratislava: FHI EU, 2025, 61 pages.

The digitalization of hotel services has become a key factor in increasing efficiency and improving customer experience. Traditional reception services can be costly and less flexible for hotel operations, leading to an increased demand for automated solutions. This thesis focuses on the design and development of a virtual reception application that utilizes advanced artificial intelligence language models, specifically GPT-4, for interactive communication with guests.

The application offers guests two primary forms of interaction with the reception. Real-time conversations with the GPT-4 model and automated guide generation with information about the hotel, its surroundings, and recommendations. Other key features include online check-in and check-out, digital room keys, and digital signing of reservations without the need for paper documentation.

The results of the application testing confirm its ability to enhance the user experience, reduce the workload of reception staff, and improve the efficiency of hotel processes. A virtual reception powered by artificial intelligence represents an innovative step toward fully digital hotel management.

Keywords:

virtual reception, hotel management, ChatGPT-4, artificial intelligence, digital check-in, automation

Obsah

ÚVOD	9
1 Súčasný stav riešenej problematiky doma a v zahraničí	11
1.1 Tradičný model recepcie v hoteloch a jeho limity	11
1.2 Automatizácia recepčných služieb	12
1.2.1 Typy systémov v hotelierstve	14
1.3 Umelá inteligencia	16
1.3.1 Súčasnosť	16
1.3.2 Generatívne modely	17
1.3.3 Etika	19
1.3.4 Aplikácia v hotelierstve	20
1.4 Predpoklady pre vývoj mobilných aplikácií	21
1.4.1 Frameworky	21
1.4.2 Prostredia	22
1.4.3 Databázy	24
1.4.4 API	25
2 Cieľ práce, metodika práce a metódy skúmania	27
2.1 Cieľ práce	27
2.2 Metódy skúmania	28
3 Výsledky práce a diskusia	30
3.1 Vytvorenie lokálneho serveru	30
3.2 Frontend	31
3.2.1 Login.....	32
3.2.2 Rezervácia	35
3.2.3 Detaily izby.....	41
3.2.4 Okolie podľa Google Maps	41
3.2.5 Miestny sprievodca	43
3.2.6 Komunikácia s modelom GPT-4.....	44
3.3 Backend.....	45

3.3.1 Login.....	45
3.3.2 Zobrazenie ubytovania	49
3.3.3 Rezervácia	50
3.3.4 Detaily rezervácie	51
3.3.5 Detaily izby.....	52
3.3.6 Okolie podľa Google Maps	53
3.3.7 Miestny sprievodca	53
3.3.8 Komunikácia s modelom GPT-4.....	54
3.4 WebView	55
3.5 Diskusia.....	57
ZÁVER	58
Zoznam použitej literatúry	59

ÚVOD

Digitalizácia a automatizácia služieb sú dnes neoddeliteľnou súčasťou moderného hotelového priemyslu. Neustále rastúce očakávania hostí, tlak na znižovanie nákladov a potreba optimalizácie pracovných procesov vedú hotelové prevádzky k implementácii inovatívnych technológií. Jednou z nich je aj virtuálna recepcia, ktorá umožňuje hosťom vybaviť celý proces registrácie, získania informácií a komunikácie s hotelom bez potreby priameho kontaktu s personálom.

Cieľom tejto práce je návrh a vývoj aplikácie virtuálnej recepcie pre hotelové prevádzky, ktorá využíva pokročilé jazykové modely umelej inteligencie, konkrétne GPT-4. Táto technológia umožňuje interaktívnu komunikáciu s hosťami a ponúka im informácie o hoteli, jeho okolí a dostupných službách. Medzi ďalšie funkcie patrí online check-in a check-out, virtuálne kľúče od ubytovania a možnosť digitálneho podpisu rezervácie, čím sa eliminuje potreba papierovej dokumentácie a zefektívňuje sa celý proces ubytovania.

Prvá kapitola sa zaoberá analýzou súčasného stavu riešenej problematiky doma a v zahraničí. Poskytuje prehľad existujúcich riešení, porovnanie ich výhod a nevýhod a identifikáciu trendov v oblasti automatizácie hotelových služieb. Osobitná pozornosť je venovaná implementácii umelej inteligencie v hotelierstve a jej aktuálnemu využitiu v praxi. Taktiež sme sa zamerali na predpoklady pre vývoj takejto aplikácie, zameranej na náhradu recepcie v hoteloch.

Druhá kapitola definuje ciele práce a metodiku jej riešenia. Opisuje prístup k návrhu a vývoju aplikácie, zvolenú architektúru systému, použité technológie a proces testovania. Zároveň popisuje metódy hodnotenia efektivity aplikácie v reálnych podmienkach.

Tretia kapitola predstavuje výsledky práce a diskusiu. Obsahuje zhodnotenie úspešnosti aplikácie, výsledky testovania a spätnú väzbu od používateľov. Diskutuje o výhodách a potenciálnych výzvach pri implementácii virtuálnej recepcie v hotelovej prevádzke a navrhuje možnosti ďalšieho rozvoja.

Hlavným prínosom tejto práce je zvýšenie efektivity, zníženie záťaže na recepčný personál a zlepšenie celkovej zákaznickej skúsenosti. Virtuálna recepcia s umelou

inteligenciou predstavuje moderný a inovatívny prístup k hotelovému manažmentu, ktorý môže prispieť k vyššej spokojnosti hostí a optimalizácii nákladov hotelových prevádzok.

1 Súčasný stav riešenej problematiky doma a v zahraničí

Automatizácia a digitalizácia v hotelierstve zohrávajú zásadnú úlohu pri zvyšovaní efektivity, znižovaní nákladov a poskytovaní prvotriedneho zážitku pre hostí. V súčasnej dobe, keď sa očakávania zákazníkov neustále zvyšujú, je nevyhnutné prispôbiť sa novým technológiám, ktoré umožňujú flexibilnejšiu a rýchlejšiu obsluhu. Implementácia digitálnych riešení ako sú online rezervácie, či mobilné aplikácie s virtuálnymi asistentmi pomáhajú optimalizovať procesy, komunikáciu s hosťami a správu služieb. Tieto technológie nielenže zvyšujú komfort pre hostí, ale tiež šetria čas a personál, čo prispieva k znižovaniu prevádzkových nákladov a zvyšovaniu spokojnosti zákazníkov.

Prvé online rezervačné systémy sa objavili v polovici 90. rokov, keď spoločnosti ako OpenTable pre reštauračné rezervácie a Doctor's Appointment Scheduler pre lekárske objednávky začali inovovať tento proces. Tieto technológie výrazne zjednodušili registráciu zákazníkov a minimalizovali chyby, čím im poskytli väčšie pohodlie. V súčasnosti sú online rezervačné systémy vyspelé platformy s mnohými pokročilými funkciami. Kľúčovú úlohu v ich rozvoji zohráva umelá inteligencia a strojové učenie. Tieto technológie umožňujú predpovedať najvhodnejšie časy pre rezerváciu, efektívne ich spravovať a prispôbiť sa zákazníkovi na základe ich preferencií. Medzi bežné funkcie patrí aj vylepšené riadenie marketingu, silné bezpečnostné opatrenia pre ochranu údajov a pokročilé analytické nástroje, ktoré zabezpečujú plynulý a bezpečný proces (Kamenchuk, 2024).

Smartfón sa stal kľúčovým faktorom v mnohých oblastiach, vrátane hotelového priemyslu, ovplyvňujúc prácu, či voľný čas. Ak dnes cestovatelia potrebujú nájsť nejaké informácie, siahajú práve po mobilnom zariadení. Ak sa im teda umožní ľahko nájsť potrebné údaje ako sú dostupnosť, vybavenie alebo turistické atrakcie a ponúkne sa im možnosť rýchlej rezervácie, je väčšia šanca, že si vyberú práve hotel, ktorý disponuje týmito funkciami pred konkurenciou. A ak bude hotelová aplikácia v ich telefóne, vzniknú ďalšie príležitosti, ako týmto zákazníkom odoslať informácie o špeciálnych ponukách, či predajoch (Hajji, 2023).

1.1 Tradičný model recepcie v hoteloch a jeho limity

Recepčné služby zohrávajú dôležitú úlohu, keďže sú prvým kontaktným bodom medzi hosťami a hotelom. Tradičná recepcia je zodpovedná za registráciu hostí, poskytovanie informácií, spracovanie platieb, riešenie požiadaviek či koordináciu ďalších hotelových

služieb. Efektívnosť recepcie výrazne ovplyvňuje celkovú spokojnosť hostí a ich ochotu sa do hotela vrátiť. S rastúcim dopytom po rýchlych a bezkontaktných službách však klasický model recepcie čelí viacerým výzvam.

Čakanie v radoch pri prihlasovaní a odhlasovaní z ubytovania môže byť pre hostí frustrujúce, najmä v čase špičky. Okrem toho, nepretržitá prevádzka recepcie znamená vysoké náklady na personál, ktoré môžu byť pre hotel značnou finančnou záťažou. Zákazníci tiež očakávajú okamžitú dostupnosť informácií a rýchlu reakciu na ich požiadavky, čo pri tradičnom recepčnom modeli nemusí byť vždy možné. Moderní cestovatelia sú čoraz viac zvyknutí na digitálne riešenia vo všetkých oblastiach života, čo vytvára tlak na hotely, aby prispôbili svoje služby novým trendom. Zjednodušenie procesov registrácie, poskytovanie informácií prostredníctvom digitálnych riešení a možnosť rýchlej interakcie s hotelom sú faktory, ktoré môžu výrazne zvýšiť spokojnosť hostí a zároveň zefektívniť hotelovú prevádzku. Výzvou zostáva nájsť riešenie, ktoré dokáže skombinovať pohodlie digitálnej komunikácie s osobným prístupom a kvalitnými službami, ktoré sú pre hotelierstvo nevyhnutné.

Hotelový sektor sa neustále mení pod vplyvom nových trendov a technologického pokroku. Recepčné služby a samotná úloha recepčného sa tomuto vývoju prispôbujú, pričom ich konkrétna podoba závisí od typu hotela a jeho potrieb. Preto sa modernizácia recepčných procesov v jednotlivých hoteloch môže líšiť, no existujú spoločné faktory, ktoré ovplyvňujú celé odvetvie. Hlavným hnacím motorom týchto zmien je digitalizácia, ktorá preniká do všetkých oblastí hotelových služieb a formuje ich budúci vývoj. Jedným z najvážnejších problémov, s ktorými sa dnes hotelový priemysel stretáva, je nedostatok kvalifikovaných pracovníkov, vrátane recepčných. Tento deficit často vedie k obmedzeniu služieb a preťaženiu súčasného personálu, čo môže mať negatívny dopad na ich spokojnosť, kvalitu poskytovaných služieb a celkový zážitok hostí. Výsledkom sú nielen finančné straty pre hotel, ale aj znížená atraktívnosť povolania recepčného, čo vedie k ďalšiemu odchodu zamestnancov a prehĺbovaniu celkového problému, čím sa vytvára začarovaný kruh (Müller, 2024).

1.2 Automatizácia recepčných služieb

Automatizácia recepčných služieb sa stáva nevyhnutným trendom v hotelovom priemysle. Hotely čoraz viac implementujú digitálne riešenia, ktoré zjednodušujú a urýchľujú procesy súvisiace s ubytovaním hostí. Hlavným cieľom automatizácie je zlepšenie efektivity,

zníženie prevádzkových nákladov a zvýšenie spokojnosti hostí prostredníctvom pohodlnejších a rýchlejších služieb.

Mnohé popredné hotelové reťazce už implementovali automatizované riešenia s cieľom zvýšiť komfort hostí, či zabezpečiť efektívnejší chod jednotlivých procesov, od rezervačných systémov až po rôzne zákaznícke služby. Tie zohrávajú čoraz dôležitejšiu úlohu v modernom hotelovom priemysle. Tí, ktorí sa chcú presadiť v oblasti hotelového manažmentu, musia pochopiť, ako tieto technologické inovácie môžu zlepšiť konkurencieschopnosť podnikov v sektore cestovného ruchu a turizmu. Integráciou do hotelových operácií a marketingových stratégií môžu hotely dosiahnuť množstvo výhod (Les Roches, 2024a).

- **Nákladová efektívnosť** – hotely môžu znížiť náklady na pracovnú silu. To umožňuje zamestnancom sústrediť sa na oblasti, ktoré zvyšujú produktivitu a zlepšujú zážitok hostí.
- **Zákaznícka skúsenosť** - vďaka mobilným aplikáciám môžu hostia jednoducho objednať službu alebo upratovanie. Implementovaním cielene zameraných systémov môžu hotelové podniky zabezpečiť presnosť pri plnení preferencií hostí. Podľa spoločnosti Zendesk (2020) zákaznícky zážitok ovplyvňuje vernosť značke u viac ako polovice spotrebiteľov.
- **Zvýšenie príjmov** - Podniky môžu zvýšiť zisky strategickou implementáciou automatizácie. Tá môže generovať ďalšie príjmové príležitosti ponúkaním nadštandardných služieb a prémiových doplnkov.

Aj keď automatizácia v hotelierstve prináša množstvo výhod, zároveň so sebou nesie výzvy v troch hlavných oblastiach: náklady na implementáciu, obavy o bezpečnosť dát a nedostatok kvalifikovaných odborníkov (Les Roches, 2024b).

- **Náklady na implementáciu** - jedným z hlavných aspektov, ktoré je potrebné zvážiť pri automatizácii v hotelierstve, sú investície do nových technológií a modernizácie existujúcich systémov. Tieto náklady môžu byť vysoké, pretože zahŕňajú nielen nákup technológií, ale aj zmenu infraštruktúry na ich podporu.

Napriek počiatočným nákladom môže automatizácia priniesť pozitívne výsledky ako zvýšenie efektivity a zníženie výdavkov na pracovnú silu.

- **Bezpečnosť dát** - pri implementácii pokročilých digitálnych riešení v hoteloch môžu vzniknúť problémy s ochranou osobných údajov hostí. Je nevyhnutné zabezpečiť integritu týchto informácií, pretože s rastúcou technológiou sa stávajú sofistikovanejšími aj podvodné aktivity. Kybernetická bezpečnosť si vyžaduje neustále aktualizácie, aby sa predišlo novým hrozbám. To znamená, že je potrebné neustále sledovanie a aktualizácie systémov na zabezpečenie maximálnej ochrany pred porušeniami, ktoré môžu ohroziť reputáciu značky a dôveru zákazníkov.
- **Nedostatok kvalifikovaných odborníkov** - zamestnanci musia mať vysokú úroveň kompetencií na obsluhu komplexného automatizovaného zariadenia alebo softvérových platforiem, ktoré riadia rôzne hotelové funkcie. Kvalifikovaní pracovníci sú tiež schopní lepšie pochopiť preferencie hostí na základe zhromaždených údajov, čo vedie k zlepšeniu celkového zážitku hostí. Riešením nedostatku zručností sú komplexné školiace programy, ktoré zamestnancov naučia výhody, funkcie a možné riziká spojené s používaním automatizovaných systémov, ako aj spôsoby efektívneho riešenia problémov.

1.2.1 Typy systémov v hotelierstve

V hotelovom svete sa využíva široká škála softvérových riešení, ktoré pomáhajú zefektívniť prevádzku, zlepšiť zákaznícku skúsenosť a optimalizovať riadenie hotelových služieb. Požívajú sa najmä na:

- **Správu majetku** - umožňujú efektívnu správu rezervácií a administratívnych úloh v hoteloch. Okrem automatizácie prevádzkových procesov v rôznych oddeleniach hotela sa tento systém stal aj nástrojom na zvýšenie viditeľnosti ubytovacích zariadení a získavanie nových rezervácií. Umožňuje spracovanie elektronických platieb a presnú správu dostupnosti izieb, čím sa predchádza nadmerným rezerváciám alebo duplicitám. Tiež často obsahuje funkciu automatického zasielania potvrdení e-mailom, čo je súčasťou modulu pre

správu rezervácií. Niektoré pokročilé riešenia umožňujú hosťom nielen rezervovať izbu, ale aj objednať ďalšie hotelové služby a aktivity, čím sa zvyšuje pohodlie a spokojnosť zákazníkov.

- **Riadenie príjmov** – softvér je navrhnutý na optimalizáciu cien hotelových izieb pomocou inteligentných algoritmov. Hlavným cieľom je zlepšiť cenové stratégie a maximalizovať výnosy z rezervácií. Keďže hotelový priemysel je mimoriadne dynamický, efektívne riadenie príjmov je kľúčové pre udržanie konkurencieschopnosti. Medzi hlavné výhody patrí strategické stanovovanie cien a vyššie tržby, no systém ponúka aj ďalšie benefity. Pomáha napríklad minimalizovať náklady tým, že zohľadňuje sezónne výkyvy v dopyte, čím umožňuje hotelovým manažérom efektívnejšie plánovať výdavky a predchádzať zbytočným nákladom.
- **Distribúciu kanálov** - umožňuje prepojenie hotelovej ponuky s viacerými online cestovnými portálmi a agentúrami, čím zabezpečuje efektívne riadenie cien a dostupnosti izieb naprieč rôznymi predajnými kanálmi. Tento systém pomáha predchádzať chybám, ktoré by mohli viesť k finančným stratám. Okrem toho poskytuje dôležité analytické údaje o zdrojoch rezervácií, čo hotelierom umožňuje identifikovať najvýhodnejšie distribučné kanály pre ich cieľovú skupinu a optimalizovať stratégiu predaja.
- **Sledovanie cien** - poskytuje jednoduché grafické rozhranie a prehľadné nástroje na tvorbu reportov, ktoré pomáhajú hotelovým majiteľom optimalizovať ceny izieb podľa aktuálneho dopytu. Tento softvér umožňuje získavať reálne údaje o konkurencii, vrátane ich cien a aktuálnych akcií. Okrem informácií o iných hoteloch tento systém umožňuje monitorovať aj umiestnenie hotela v miestnych rebríčkoch a získavať štatistiky o priemerných cenách v danej oblasti.
- **Správu rezervácií** - je navrhnutý na sledovanie spätnej väzby od online používateľov o službách a produktoch. Tento systém pomáha šíriť pozitívne recenzie zákazníkov prostredníctvom relevantných online platforiem. Budovanie reputácie pomocou pozitívnych recenzií je pre majiteľov hotelov prospešné, pretože môže prispieť k zvýšeniu predaja. Hotely tento softvér

využívajú na propagáciu svojho podnikania a získavanie negatívnych recenzií od zákazníkov. Je to tiež užitočný nástroj, ktorý pomáha monitorovať príspevky na sociálnych sieťach o ich službách. Následne môžu majitelia reagovať na tieto recenzie a rýchlo vyriešiť prípadné problémy (Barten, 2023).

1.3 Umelá inteligencia

Umelá inteligencia je technológia, ktorá za posledné desaťročie prešla výrazným vývojom a stala sa kľúčovým faktorom v mnohých oblastiach života. Od riešenia zložitých problémov po zefektívnenie pracovných procesov. Dnes ovplyvňuje každodenný život, ekonomiku a spoločnosť. Jej aplikácie sa nachádzajú v širokom spektre, od priemyselnej výroby, cez zdravotnú starostlivosť až po finančný sektor a zábavu. Nezameriava sa len na simuláciu ľudskej inteligencie, ale tiež na vytváranie schopností strojov a systémov učiť sa z dát, rozpoznávať vzory, predpovedať trendy a optimalizovať rozhodovacie procesy. Moderné pokroky v oblastiach ako strojové učenie, hlboké učenie a spracovanie prirodzeného jazyka umožnili umelým inteligenciám vykonávať činnosti, ktoré predtým vyžadovali ľudskú expertízu.

1.3.1 Súčasnosť

V súčasnosti je umelá inteligencia čoraz viac integrovaná do rôznych procesov, kde prináša významné zlepšenia v efektivite a kvalite poskytovaných služieb. Hotely ju využívajú na optimalizáciu cenovej politiky a správu dostupnosti izieb v reálnom čase. Systémy na analýzu veľkých dát pomáhajú predpovedať dopyt a určovať najvhodnejšie cenové stratégie, čo vedie k lepšiemu využitiu kapacít a maximalizácii príjmov. Je tiež kľúčová v oblasti správy prevádzky, kde umožňuje automatizáciu mnohých administratívnych úkonov, ako je spracovanie rezervácií, sledovanie údržby či efektívnejšie riadenie zásob. Systémy na báze umelej inteligencie umožňujú získať hlbší prehľad o prevádzke a optimalizovať prevádzkové náklady, čo prispieva k zlepšeniu celkovej efektivity. Implementácia umelej inteligencie prináša aj nedokonalosti, najmä v oblasti bezpečnosti dát, integrácie nových systémov s existujúcimi technológiami a potrebou školenia personálu, ktorý bude s týmito novými nástrojmi pracovať. Aj napriek týmto nedostatkom sa stáva neoddeliteľnou súčasťou

moderného hotelového manažmentu, čo pomáha zlepšiť konkurencieschopnosť a uspokojenie zákazníkov.

Nové technológie umelej inteligencie majú potenciál znovu určovať štandardné postupy a automatizovať pracovné procesy v oblasti hotelierstva. Prieskumy ukazujú, že hostia tento posun očakávajú. Podľa správy spoločnosti Hotel Tech Report (2024) päťdesiatosem percent hostí verí, že umelá inteligencia môže zlepšiť ich pobyt v hoteli. Strojové učenie pomáha firmám zlepšiť prediktívnu analýzu a poskytuje podrobnejšie informácie o zákazníkoch. Vďaka algoritmom môžu spoločnosti v hotelierstve analyzovať veľké množstvo dát z rôznych zdrojov, ako sú interakcie, hodnotiace formuláre alebo rezervácie. Taktiež vedú identifikovať trendy a individuálne preferencie. Tieto informácie umožňujú hotelom predvídať potreby hostí, či optimalizovať cenovú politiku. Personalizácia je kľúčová, pretože podporuje vernosť zákazníkov. Budúce trendy naznačujú sofistikovanejšie prispôsobenie, ktoré sa bude opierať o údaje a správanie hostí v reálnom čase. To zahŕňa prispôsobovanie nastavení izieb, odporúčania na stravovanie a dokonca aj celkové itineráre. V budúcnosti umelá inteligencia ešte viac posilní schopnosť hotelov a iných poskytovateľov služieb vytvárať jedinečné a nezabudnuteľné zážitky pre hostí (Amit, 2024).

1.3.2 Generatívne modely

Generatívna umelá inteligencia je typ umelej inteligencie, ktorý dokáže vytvárať nový obsah, napríklad text, obrázky, hudbu, kód či dokonca videá na základe vzorov naučených z existujúcich dát. Na rozdiel od klasických modelov, ktoré sa zameriavajú na analýzu a rozpoznávanie údajov, táto umelá inteligencia dokáže dáta kreatívne využívať a generovať nové výstupy, ktoré sa podobajú tým, na ktorých sa model učil. Tento typ využíva pokročilé neurónové siete a generatívne modely, napríklad GPT, DALL·E. Svoje využitie nachádza v rôznych oblastiach. Od automatizovaného písania článkov, tvorby realistických obrázkov až po vývoj virtuálnych asistentov. Má obrovský potenciál, ale prináša aj nedostatky ako je vytváranie falošných videí a nahrávok (deepfake), či etické otázky týkajúce sa autorských práv a dezinformácií.

Diskriminačné modelovanie sa používa na klasifikáciu existujúcich dátových bodov, napríklad rozpoznanie obrázkov mačiek a morčiat a ich zaradenie do príslušných kategórií.

Väčšinou patrí medzi úlohy strojového učenia pod dohľadom. Generatívne modelovanie sa snaží pochopiť štruktúru dátovej množiny a generovať podobné príklady ako napríklad vytvorenie realistického obrázka mačky alebo morčat'a. Väčšinou spadá do strojového učenia bez dohľadu. Diskriminačné modely strojového učenia sa používajú na klasifikáciu údajov na základe ich charakteristík a na predpovedanie, do akej kategórie konkrétny dátový bod patrí. Ich úlohou je rozlišovať medzi rôznymi triedami údajov na základe vzorcov naučených počas tréningu. Predstavme si, že máme súbor tréningových údajov, ktorý obsahuje obrázky mačiek a morčiat. Každý obrázok predstavuje vzorku s určitými vlastnosťami a pridelenou triedou, ktorá označuje, či ide o mačku alebo morča. Trénovaním neurónovej siete sa model učí rozpoznávať rozdiely medzi týmito dvoma kategóriami a vypočítať pravdepodobnosť, do ktorej z nich daný obrázok patrí. Počas tréningu sa predpovedané výsledky modelu porovnávajú so skutočnými triedami. Model na základe týchto porovnaní postupne upravuje svoje výpočty, aby sa presnejšie naučil rozlišovať medzi rôznymi triedami údajov. Ak by sme rozdiely medzi mačkami a morčatami definovali len pomocou dvoch vlastností, napríklad prítomnosť chvosta a tvar uší, mohli by sme tieto rozdiely vizuálne znázorniť v dvojrozmernom grafe. Po natrénovaní model jednoducho skúma, na ktorej strane tejto hranice sa nový obrázok nachádza, a na základe predchádzajúcich skúseností určuje, či ide o mačku alebo morča. Diskriminačný model v podstate zhromažďuje a analyzuje informácie o rozdieloch medzi jednotlivými kategóriami, avšak bez hlbšieho pochopenia samotnej podstaty objektov, ktoré klasifikuje. Generatívne algoritmy fungujú opačne. Namiesto predpovedania triedy na základe určitých vlastností sa snažia predpovedať vlastnosti na základe určitej triedy. Kým diskriminačné algoritmy sa zameriavajú na vzťahy medzi vlastnosťami a triedami, generatívne modely sa sústredia na to, ako získať vlastnosť z triedy. Generatívne modelovanie matematiky umožňuje zachytiť pravdepodobnosť, že sa hodnoty vlastností a tried vyskytnú spoločne. Tento prístup sa zameriava na pochopenie vlastností a ich vzťahov, čo pomáha určiť, čo robí mačky mačkami a morčatá morčatami. Výsledkom je, že generatívne algoritmy nielen rozlišujú medzi týmito dvoma zvieratami, ale môžu tiež vytvoriť alebo rekonštruovať ich obraz. Odpoveďou na otázku prečo sú vôbec diskriminačné algoritmy potrebné je, že sú jednoduchšie na sledovanie a poskytujú väčšiu interpretovateľnosť. Teda je ľahšie pochopiť, prečo model dospel k určitému záveru. Navyše, v mnohých prípadoch nezáleží na tom, ako boli dáta pôvodne generované. Dôležité je len to, do akej kategórie patria, a v tomto prípade

diskriminačné modely excelujú. Napríklad pri analýze sentimentu v recenziách hotelov je cieľom určiť, či je komentár pozitívny alebo negatívny, nie vytvárať falošné recenzie. Diskriminačné modely sú stále preferovanou voľbou pri rozpoznávaní obrázkov, klasifikácii dokumentov, detekcii podvodov a mnohých ďalších každodenných obchodných úlohách. Generatívna umelá inteligencia (AI) však ponúka ďalšie možnosti pre obchodné aplikácie, ktoré presahujú oblasti, kde sú diskriminačné modely využívané (Altexsoft, 2024).

1.3.3 Etika

V oblasti umelej inteligencie sa etika zaoberá morálnymi otázkami týkajúcimi sa vývoja, implementácie a jej používania. Hlavnými problémami sú ochrana osobných údajov, zodpovednosť za rozhodnutia, dopady na pracovný trh a potenciálna zneužitelnosť tejto technológie. S rastúcim množstvom údajov, ktoré umelá inteligencia spracováva, je dôležité zabezpečiť ochranu súkromia a transparentnosť rozhodovaní. Môže tiež nahradiť niektoré pracovné miesta, čo vedie k obavám o nezamestnanosť, no zároveň vytvára nové príležitosti. Technológie ako deepfake môžu byť zneužitú na šírenie dezinformácií a kybernetické útoky.

Etika umelej inteligencie zahŕňa súbor zásad, ktoré zabezpečujú zodpovedný vývoj a využívanie tejto technológie. Tieto zásady sa zameriavajú na bezpečnosť, ochranu súkromia, humánnosť a ochranu životného prostredia pri jej použití. Etické normy pre sa zameriavajú na zabránenie zaujatosti, ochranu osobných údajov a minimalizáciu environmentálnych rizík. Implementácia etických pravidiel sa uskutočňuje cez firemné etické rámce a regulačné politiky vlád, ktoré usmerňujú vývoj a aplikácie na národnej a globálnej úrovni. V posledných rokoch sa diskusia o etike presunula z akademických a neziskových organizácií do veľkých technologických spoločností ako IBM, Google a Meta, ktoré vytvorili tímy na riešenie etických problémov spojených so zberom údajov, pričom vlády a medzinárodné organizácie začali formulovať regulácie a etické smernice vychádzajúce z akademického výskumu (Coursera, 2025).

Pre organizácie neznamená etické využívanie umelej inteligencie len prijatie etických zásad. Tieto zásady musia byť implementované do všetkých technických a prevádzkových procesov spojených s ňou spojených. Aj keď môže byť pre organizácie, ktoré ju rýchlo zavádzajú, implementácia etických princípov náročná, skutočné prípady problémov spôsobených nedostatočne navrhnutými modelmi ukazujú, že ignorovanie etiky môže mať

vážne a nákladné následky. Mnohé spoločnosti vytvárajú výbory pod vedením vrcholových manažérov na formovanie politik riadenia AI. Napríklad v spoločnosti SAP sa zriadil tím, ktorý sa skladá z odborníkov na etiku a technológie, aby sa implementovali zásady do produktov a prevádzky (SAP, 2024).

1.3.4 Aplikácia v hotelierstve

V hotelierstve sa stále viac otvárajú nové možnosti využitia AI. Napríklad na zlepšenie služieb a optimalizáciu prevádzky. Aj keď ešte nie je bežne súčasťou všetkých hotelových prevádzok, jej potenciál je obrovský a predpokladá sa, že bude hrať kľúčovú úlohu pri transformácii celého odvetvia. Implementácia hotelových systémov by mohla viesť k zásadným zlepšeniam v rôznych oblastiach, ako je efektivita operácií, personalizácia zákazníckych služieb a celkové zjednodušenie komunikácie medzi hotelom a hosťami. Jednou z najväčších príležitostí, ktorú umelá inteligencia ponúka, je automatizácia rutinných úloh, ako je spracovanie rezervácií, poskytovanie informácií o hoteli či odpovedanie na najčastejšie otázky hostí. Systémy ňou vybavené môžu byť schopné okamžite reagovať na požiadavky hostí, odporúčať služby alebo aktivity podľa individuálnych preferencií, a tým zefektívniť komunikáciu, čo zároveň znižuje záťaž na recepcný personál. Takéto riešenia by mohli zlepšiť efektivitu a zároveň zvýšiť spokojnosť hostí, ktorí by mali prístup k informáciám kedykoľvek a bez čakania.

Analytici odhadujú, že trh s technológiami umelej inteligencie v oblasti hotelierstva dosiahol v roku 2023 približne deväťdesiat miliónov dolárov, pričom očakávajú, že tento trh bude rásť veľmi rýchlo, a to až o šesťdesiat percent ročne, pričom do roku 2033 by mal dosiahnuť viac než 8 miliárd dolárov. Tento rast je možný vďaka tomu, že neexistuje jediný typ umelej inteligencie pre hotelierstvo. Namiesto toho je integrovaná do rôznych technológií. Hotely ju zatiaľ využívajú na optimalizáciu rozvrhu upratovania, či na podporu dynamického stanovenia cien. Okrem týchto inovatívnych riešení nachádza uplatnenie aj v menej dramatických, no rovnako dôležitých oblastiach, ako je zlepšenie prevádzky, podpora udržateľnosti a znižovanie bezpečnostných rizík. Tento rýchly rozvoj v hotelierstve vyvoláva čoraz väčší záujem o tieto technológie. Výskumy ukazujú, že šesťdesiat percent hotelov a sedemdesiat percent cestovných agentúr plánuje implementovať umelú inteligenciu do svojich systémov (Schwarz, 2025).

Implementácia umelej inteligencie do hotelových systémov predstavuje veľký potenciál pre zlepšenie služieb a optimalizáciu prevádzky. Aj keď niektoré technológie ešte nie sú bežne používané, predpokladá sa, že ich využitie v oblasti hotelierstva bude čoraz bežnejšie a zlepši celkovú kvalitu poskytovaných služieb, čím sa zlepši konkurencieschopnosť na trhu.

1.4 Predpoklady pre vývoj mobilných aplikácií

Vývoj aplikácie v akomkoľvek odvetví si vyžaduje dôkladné pochopenie technických, prevádzkových a používateľských potrieb, ktoré sú nevyhnutné pre jej úspešné nasadenie a dlhodobú udržateľnosť. Základom je identifikácia kľúčových technologických komponentov a inovácií, ktoré umožnia vytvoriť efektívne, bezpečné a škálovateľné riešenie. V prípade vývoja aplikácií, ktoré majú za úlohu zefektívniť existujúce procesy alebo priniesť nové služby, je potrebné kombinovať viaceré technológie, ktoré spoločne zabezpečia bezproblémovú prevádzku.

Vývoj mobilných aplikácií je v súčasnosti nevyhnutný a kľúčový pre podniky všetkých veľkostí v digitálnom svete. Vstup smartfónov a rýchly rozvoj mobilných technológií zásadne zmenili spôsob, akým komunikujeme a fungujeme v každodennom živote. Mobilné aplikácie sa stali neoddeliteľnou súčasťou našich aktivít, od pracovnej produktivity až po nakupovanie a zábavu. Vytvorenie mobilných aplikácií prináša pre podniky množstvo výhod a príležitostí, ktoré im pomáhajú rozširovať základňu zákazníkov a zostať konkurencieschopnými. Prieskum, ktorý uskutočnila spoločnosť Clutch (2025) ukázal, že štyridsaťdva percent malých podnikov už má mobilnú aplikáciu, zatiaľ čo tridsať percent plánuje vyvinúť jednu v budúcnosti, čo naznačuje rastúci záujem o vývoj aplikácií naprieč rôznymi odvetvami (Shulgin, 2023).

1.4.1 Frameworky

Predpripravená štruktúra alebo platforma, ktorá poskytuje vývojárom nástroje, knižnice a rozhrania na rýchlejší a efektívnejší vývoj softvéru (Framework) slúži ako základ pre efektívny vývoj aplikácií. V mobilnom svete zjednodušujú proces vytvárania aplikácií tým, že ponúkajú opakovane použiteľný kód a funkcie, ktoré by inak musel vývojár písať od začiatku. Poskytujú hotové riešenia pre bežné problémy, čo výrazne urýchľuje celý proces vývoja, umožňujú dodržiavať osvedčené postupy a štandardy, čo pomáha udržať aplikácie konzistentné a menej náchylné na chyby. Zdieľanie kódu medzi rôznymi aplikáciami alebo

platformami šetrí čas pri implementácii bežných funkcií, čo šetrí náklady a umožňuje rýchlejšie uvedenie aplikácie na trh.

Flutter je populárny framework, ktorý bol uvedený spoločnosťou Google v roku 2017. Na vytváranie aplikácií je potrebné používať programovací jazyk Dart, ktorý je taktiež vyvinutý touto firmou. Medzi aplikácie, ktoré ho používajú patrí napríklad eBay, Alibaba, či Google Pay. Jednou z kľúčových vlastností je funkcia hot reload, ktorá umožňuje vývojárom okamžite vidieť zmeny v aplikácii bez nutnosti jej opätovnej kompilácie (JetBrains, 2025a).

React Native je dnes najpopulárnejšou voľbou vo svete mobilných aplikácií. Ako platforma pre vývoj je ideálna na vytváranie atraktívnych, intuitívnych a natívnych aplikácií pre Android a iOS. Ponúka nezávislé komponenty ako obrázky, texty a zobrazenia, čím zabezpečuje plynulý používateľský zážitok. Okrem toho využíva JavaScript, ktorý je jedným z najflexibilnejších programovacích jazykov pre vývoj mobilných aplikácií (Shah, 2024).

Ionic poskytuje komplexné riešenie na vývoj efektívnych, multiplatformových mobilných aplikácií. Vďaka širokej knižnici komponentov a funkcií, ako je zdieľanie kódu, môžu vývojári vytvárať kvalitné aplikácie a tým šetriť čas. Ponúka tiež automatizované testovanie. Okrem toho umožňuje integráciu s populárnymi frameworkami ako Angular a React Native. S cloudovými službami, rýchlymi časmi zostavovania, jednoduchým nasadzovaním a optimalizovaným procesom kontinuálnej integrácie umožňuje tento framework rýchlejší vývoj aplikácií než kedykoľvek predtým (Autor, 2024).

Kotlin Multiplatform je open-source technológia vyvinutá spoločnosťou JetBrains, ktorá umožňuje zdieľanie kódu medzi rôznymi platformami, pričom si zachováva výhody natívneho programovania. Vývojárom umožňuje opätovné využitie a bezproblémovú integráciu zdieľaného kódu do akéhokoľvek projektu. Kotlin je programovací jazyk používaný pre tento framework a umožňuje vývoj aplikácií pre Android, iOS, web, či desktop. Príklady mobilných aplikácií, ktoré túto technológiu používajú, sú McDonald's, Netflix“ Forbes alebo Philips (JetBrains, 2025b).

1.4.2 Prostredia

Je ťažké si to predstaviť, ale existovali časy, keď bolo všetko potrebné písať ručne v základnom textovom editore, bez možnosti okamžitého overenia syntaxe alebo integrovaných

ladiacich nástrojov. Aj keď je stále možné vytvárať a vyvíjať softvér takýmto spôsobom, existuje dnes už množstvo pokročilých vývojových prostredí, ktoré našu prácu značne uľahčujú. Je dôležité vybrať si správne nástroje pre konkrétnu úlohu.

Pre vývoj mobilných aplikácií existuje niekoľko prostredí, ktoré ponúkajú rôzne nástroje a funkcie na uľahčenie procesu tvorby.

- **Android Studio** - oficiálne integrované prostredie pre vývojárov Android aplikácií, navrhnuté špecificky pre mobilný vývoj. Je podporované spoločnosťami Google a JetBrains. Tento program sa stal prakticky štandardom pre vytváranie Android aplikácií. Ponúka efektívne nástroje ako inteligentný editor kódu s pokročilým dokončovaním, optimalizáciu, či analýzu kódu v reálnom čase. Má vstavaný Android emulátor, ktorý umožňuje testovanie aplikácií na rôznych zariadeniach vrátane telefónov, tabletov, televízorov a hodínok. Toto vývojové prostredie je bezplatné a neustále sa aktualizuje, aby poskytovalo moderné nástroje pre efektívny vývoj aplikácií (Lijewski, 2024).
- **Xcode** - je integrované vývojové prostredie od Apple určené na tvorbu aplikácií pre macOS, iOS, watchOS a tvOS. Tento nástroj je predinštalovaný na každom Macu zariadení a je k dispozícii od roku 2003. Poskytuje vývojárom všetky potrebné nástroje na vývoj aplikácií pre platformy Apple. Obsahuje rôzne knižnice, kompilátor, či nástroje na ladenie. Editor kódu ponúka funkcie ako zvyrazňovanie syntaxe, automatické dokončovanie a detekciu chýb v reálnom čase. Nástroj na ladenie umožňuje nastavenie zarážok, kontrolu premenných a monitorovanie výkonu aplikácie. (Sharma, 2023a).
- **Eclipse** - bezplatné a open-source integrované vývojové prostredie, ktoré bolo pôvodne vyvinuté spoločnosťou IBM a v súčasnosti je udržiavané organizáciou Eclipse Foundation. Pred príchodom Android Studia bol široko používaný v spojení s Android Development Tools (ADT) pre vývoj Android aplikácií. Je jedným z najpopulárnejších na trhu vďaka podpore viac ako sto programovacích jazykov, vrátane C++, JavaScriptu, Ady a Rustu. Medzi jeho hlavné vlastnosti patrí podpora nástrojov ako Gradle, Maven a Apache Ant,

široká škála pluginov na rozšírenie funkcionality a nástroje na automatické dokončovanie kódu, či kontrolu syntaxe (Sharma, 2023b).

1.4.3 Databázy

Databázy zohrávajú kľúčovú úlohu v mnohých mobilných aplikáciách, najmä keď ide o uchovávanie a správu dát, ktoré sú nevyhnutné pre správne fungovanie aplikácie. Od základných údajov používateľov až po zložité dáta, ktoré aplikácie potrebujú na vykonávanie rôznych funkcií. Poskytujú potrebnú infraštruktúru na organizáciu a spravovanie týchto informácií. Výber správnej databázy závisí od požiadaviek aplikácie. Napríklad výkonnosť, potreba pripojenia na internet a komplexnosť dátovej štruktúry.

Amazon sa stal lídrom na trhu v oblasti databáz a ponúka rôzne doplnkové služby, ako sú dátový sklad a služby na integráciu dát, ktoré uľahčujú ich správu. Medzi jeho aktuálne databázy patrí Amazon RDS. Tá funguje na serveroch Oracle, SQL alebo MySQL. Ďalším úložiskom údajov je Amazon SimpleDB, ktorá je primárne určená na menšie pracovné záťaž. V neposlednej rade sem patrí Amazon DynamoDB, ktorá je schopná automaticky replikovať pracovné záťaž cez tri zóny (Yehuda, 2022).

Azure je cloudová služba od Microsoftu, známa predovšetkým vďaka operačnému systému Windows. Ide o silného poskytovateľa cloudových riešení, ktorý disponuje rozsiahlym infraštruktúrnym zázemím. Ponúka SQL databázovú službu, ktorá umožňuje jednoduché vytvorenie aplikácie. Nastavenie databázy je rýchle a jednoduché. Stačí mať Microsoft účet. Po jej vytvorení ju možno kombinovať s ďalšími službami alebo integrovať s riešeniami od iných cloudových poskytovateľov. Táto služba je známa pokročilým zabezpečením, pretože tisíce odborníkov na kybernetickú bezpečnosť dohliadajú na jej ochranu. Umožňuje tiež plánované automatické zálohovanie dát, vďaka čomu je možné ich kedykoľvek obnoviť (Ionescu, 2025).

Oracle Database patrí medzi pokročilý relačný databázový systém od spoločnosti Oracle, ktorý efektívne spravuje veľké objemy štruktúrovaných dát. Podporuje nasadenie ako v lokálnom prostredí, tak aj v cloudovom, čím ponúka flexibilitu pre rôzne podnikové potreby. Využíva sa na spracovanie transakcií, dátové sklady a podnikové aplikácie. Okrem štandardných funkcií na správu dát ponúka aj širokú škálu bezpečnostných možností, vysokú

dostupnosť a pokročilé analytické nástroje. Vďaka týmto vlastnostiam je ideálnym riešením pre organizácie, ktoré potrebujú spoľahlivú a výkonnú databázovú platformu na správu kritických údajov a zložitých pracovných záťaží (GeeksForGeeks, 2024).

1.4.4 API

Application Programming Interface (API) je rozhranie, ktoré umožňuje aplikáciám komunikovať medzi sebou. Dokáže pristupovať k funkcionalitám alebo dátam iných systémov. Funguje na princípe žiadostí a odpovedí. Aplikácia pošle požiadavku na API server, ten ju spracuje a pošle odpoveď. Táto komunikácia prebieha väčšinou prostredníctvom Hypertext Transfer Protokolu (HTTP) a využíva formáty JavaScript Object Notation (JSON) alebo Extensible Markup Language (XML).

API môžeme prirovnať k elektrickým zásuvkám. Softvérová aplikácia, napríklad ovládanie prehrávania videa na webovej stránke, funguje ako spotrebič. Dáta alebo služby, ktoré aplikácia potrebuje, napríklad YouTube, predstavujú elektrickú sieť. API, v tomto prípade YouTube prehrávač, zohráva úlohu štandardnej zásuvky, ktorá umožňuje rôznym zariadeniam pripojiť sa k sieti a využívať jej energiu. Pracujú na základe presne definovaných pravidiel a protokolov, ktoré zabezpečujú spoľahlivú výmenu údajov a efektívnu spoluprácu medzi systémami. Existujú rôzne druhy určené pre špecifické účely a potreby vývojárov (Nguyen,2023).

Verejné API sú dostupné pre každého vývojára alebo firmu, ktorá ich chce využívať. Spoločnosti, ktoré chcú umožniť prístup k svojim aplikáciám a dátam iným subjektom ich vytvárajú ako súčasť svojej obchodnej stratégie. Existujú určité mechanizmy na overenie identity a kontrolu prístupu, ale nie sú príliš prísne. Niektoré spoločnosti môžu svoje API spoplatniť, pričom si účtujú poplatok za každé použitie (Bigelow, 2024a).

Interné API sa používajú výhradne internými vývojármi v organizácii. Uľahčujú prenos dát medzi rôznymi časťami systému, čím podporujú automatizáciu procesov. Umožňujú rôznym aplikáciám a systémom v organizácii vzájomne komunikovať, čo ich robí nezávislými pokiaľ ide o úpravy alebo údržbu. Zavádzanie inovácií a nových služieb do existujúcich systémov je taktiež jednoduchšie (Shepard, 2023).

Partnerské API sú určené len pre konkrétne a schválené externých vývojárov alebo organizácie, ktoré sa podieľajú na obchodných vzťahoch medzi firmami. Napríklad, ak firma chce zdieľať údaje o zákazníkoch iba s vybranými externými partnermi, umožní sa pripojenie iba medzi nimi, pričom iné pripojenie nie je povolené. Partneri majú prístup k týmto dátam na základe jasne stanovených práv a licencií. Preto zahŕňajú prísnejšie bezpečnostné, autentifikačné a autorizačné mechanizmy (Bigelow, 2024b).

Kombinované API umožňujú vykonať viacero požiadaviek naraz. To šetrí dáta a zlepšuje efektivitu aplikácie, pretože sa znižuje počet volaní. Príkladom je GraphQL dotaz, ktorý umožňuje poslať všetky požiadavky cez jeden bod a získať presne požadované dáta bez nadmerného načítavania. Postman Collections je ďalším príkladom, kde sa viacero volaní zoskupí do jedného dotazu. Mnoho verejných kombinovaných API je k dispozícii na prehľadávanie a môžu poskytnúť ďalšie nápady na efektívne využitie (Simpson, 2022).

2 Cieľ práce, metodika práce a metódy skúmania

V predchádzajúcej kapitole sme sa venovali analýze súčasného stavu riešenej problematiky doma i v zahraničí. Preskúmali sme tradičný model recepcie v hoteloch, jeho limity a výzvy, ktorým čelí v súvislosti s rastúcimi očakávaniami hostí a potrebou digitalizácie hotelových služieb. Zároveň sme sa zamerali na automatizáciu recepčných služieb, dostupné technologické riešenia a možnosti využitia umelej inteligencie v hotelierstve. V neposlednom rade sme sa venovali aj predpokladom pre vývoj mobilných aplikácií, pričom sme analyzovali najčastejšie používané frameworky, vývojové prostredia, databázové riešenia a API. V tejto kapitole definujeme hlavný cieľ práce, ktorým je návrh a vývoj aplikácie virtuálnej recepcie pre hotelové prevádzky. Zároveň sa zameriame na metodiku riešenia, opis výskumného prístupu a použité metódy skúmania, ktoré nám umožnia vyhodnotiť efektivitu aplikácie v reálnych podmienkach. V rámci tejto kapitoly sa teda sústredíme na teoretický základ potrebný pre realizáciu praktickej časti práce a poskytneme prehľad o výskumných postupoch, ktoré nám pomôžu dosiahnuť stanovené ciele.

2.1 Cieľ práce

Cieľom tejto diplomovej práce je navrhnúť a implementovať aplikáciu virtuálnej recepcie pre hotelové prevádzky s využitím umelej inteligencie, konkrétne modelu ChatGPT-4. Aplikácia by mala umožniť hosťom efektívne komunikovať s hotelom, získavať informácie o ubytovaní a vybaviť procesy, ako sú online check-in a check-out, bez potreby priameho kontaktu s recepčným personálom. Praktickým výstupom práce je funkčná webová aplikácia so všetkými potrebnými prvkami na uľahčenie hotelového manažmentu a zlepšenie zákazníckej skúsenosti.

Aplikácia bola taktiež rozšírená o automatizovaný systém odporúčaní, ktorý hosťom poskytuje informácie o hoteli, dostupných službách a zaujímavostiach v okolí. Okrem interaktívnej komunikácie a zjednodušenia hotelových procesov bola virtuálna recepcia navrhnutá a implementovaná s dôrazom na používateľsky prívetivé rozhranie a bezpečnosť dát. Používateľské prostredie je intuitívne a jednoducho ovládateľné, aby hostia mohli rýchlo a pohodlne vybaviť všetky potrebné úkony počas svojho pobytu.

V rámci vývoja aplikácie boli použité moderné prostredia a programovacie jazyky, ako je HyperText Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript, Hypertext Preprocessor (PHP) a Android Studio.

Aby sme dosiahli hlavný cieľ, stanovili sme si nasledujúce čiastkové ciele:

- Analýza existujúcich riešení
- Návrh architektúry aplikácie
- Implementácia a testovanie aplikácie
- Hodnotenie efektivity aplikácie
- Identifikácia možností ďalšieho rozvoja

2.2 Metódy skúmania

Pri vypracovaní tejto diplomovej práce sme využili viacero metód, ktoré nám pomohli efektívne dosiahnuť stanovené ciele. Jednou z kľúčových metód bolo štúdium dostupných informácií z verejných zdrojov. Táto metóda nám umožnila získať potrebné teoretické poznatky o digitalizácii hotelových služieb, umelej inteligencii a súčasných technologických trendoch v hotelierstve.

Ďalšou použitou metódou bola analýza konkurencie, ktorá nám poskytla prehľad o existujúcich riešeniach v oblasti automatizácie recepčných služieb. Prostredníctvom tejto metódy sme mohli porovnať dostupné systémy, identifikovať ich silné a slabé stránky a určiť najlepšie prístupy pri návrhu vlastného riešenia. Skúmanie konkurencie nám zároveň pomohlo pochopiť aktuálne trendy v odvetví a identifikovať oblasti, v ktorých by naša aplikácia mohla priniesť konkurenčnú výhodu.

Ďalšou metódou využitou v práci bola analýza dát, ktorá slúžila na zber a spracovanie štatistických údajov o hotelových prevádzkach a zákazníckom správaní. Táto metóda zahŕňala zber relevantných údajov, ich modelovanie a následnú interpretáciu pomocou rôznych analytických nástrojov. Výsledky tejto analýzy nám umožnili lepšie pochopiť potreby používateľov a optimalizovať funkcionality aplikácie na základe reálnych dát.

Taktiež sme využili metódu implementácie, ktorá zahŕňala systematický postup pri návrhu a vývoji aplikácie. Tento proces pozostával z viacerých fáz, od analýzy požiadaviek, návrhu architektúry riešenia, cez samotnú implementáciu až po testovanie a nasadenie do prevádzky. Pri implementácii sme dbali na výber vhodných technológií a nástrojov, ktoré zabezpečili stabilitu a efektívnosť aplikácie.

Správna kombinácia týchto výskumných metód nám umožnila efektívne vyhodnotiť dostupné informácie, analyzovať konkurenciu, interpretovať získané dáta a úspešne realizovať vývoj aplikácie. Tento prístup prispel k dosiahnutiu hlavného cieľa práce a poskytol cenné poznatky pre ďalší rozvoj v oblasti automatizácie hotelových služieb.

3 Výsledky práce a diskusia

V predchádzajúcej kapitole sme sa zamerali na stanovenie cieľov práce, metodiku riešenia a výskumné metódy, ktoré sme použili pri návrhu a vývoji aplikácie virtuálnej recepcie. Definovali sme hlavný cieľ práce, ktorým je vytvorenie inteligentnej recepčnej aplikácie využívajúcej umelú inteligenciu na automatizovanú komunikáciu s hotelovými hosťami. Opísali sme tiež jednotlivé čiastkové ciele, ako analýzu existujúcich riešení, návrh architektúry aplikácie, implementáciu kľúčových funkcionalít a testovanie efektivity aplikácie v reálnych podmienkach.


V tejto kapitole sa budeme venovať praktickému výstupu práce – procesu návrhu a vývoja aplikácie virtuálnej recepcie. Podrobne popíšeme jednotlivé kroky realizácie, od výberu vhodných technológií a návrhu používateľského rozhrania až po samotnú implementáciu kľúčových funkcionalít, ako sú online check-in a check-out, digitálne kľúče od ubytovania, či automatizované odporúčania pre hostí. Následne sa zameriame na testovanie aplikácie, jej funkcionality a celkovej efektivity v hotelovej prevádzke.

Okrem opisu vývoja aplikácie sa v tejto kapitole sústredíme aj na diskusiu o dosiahnutých výsledkoch. Zhodnotíme, do akej miery sa nám podarilo splniť stanovené ciele, aké výzvy sme počas vývoja riešili a aké sú možné smerovania pre ďalší vývoj aplikácie. Diskutovať budeme aj o výhodách a nevýhodách implementovaného riešenia v porovnaní s tradičnými recepčnými modelmi a automatizovanými alternatívami na trhu.

3.1 Vytvorenie lokálneho serveru

Na vývoj a testovanie aplikácie virtuálnej recepcie sme využili lokálny server, ktorý nám umožnil simulovať prostredie webovej aplikácie bez nutnosti priameho pripojenia na vzdialený hosting. Pre tento účel sme použili softvérový balík WampServer, ktorý poskytuje kompletne riešenie na spustenie webového servera na lokálnom počítači. Zahŕňa webový server Apache, databázový systém MySQL a skriptovací jazyk PHP, čo sú kľúčové technológie potrebné na prevádzku našej aplikácie.

Po nainštalovaní WampServeru sme spustili Apache server, ktorý umožňuje spracovanie PHP skriptov a požiadaviek. Následne sme nakonfigurovali MySQL databázu, do ktorej sa ukladajú dôležité údaje o používateľoch a rezerváciách.



Tabuľka	Akcia	Riadkov	Typ	Zotriedenie	Veľkosť	Naviac
<input type="checkbox"/> checkins	★ Prechádzať Štruktúra Hľadať Vložiť Vyprázdniť Odstrániť	8	MyISAM	utf8mb3_unicode_ci	28,0 KiB	-
<input type="checkbox"/> login_data	★ Prechádzať Štruktúra Hľadať Vložiť Vyprázdniť Odstrániť	4	MyISAM	utf8mb3_unicode_ci	2,6 KiB	52 B
2 tabuľky Celkom		12	MyISAM	utf8mb3_unicode_ci	30,6 KiB	52 B

Obrázok 1: Databáza MySQL
Zdroj: Vlastné spracovanie

Aby mohla aplikácia efektívne komunikovať s databázou na lokálnom serveri, bolo potrebné vytvoriť konfiguračný súbor, ktorý k nej zabezpečuje pripojenie. Tento súbor obsahuje údaje potrebné na nadviazanie spojenia medzi PHP skriptmi a databázou bežiacou na WampServeri.

```
<?php
$host = 'localhost';
$username = 'root';
$password = '';
$dbname = 'vr';

$conn = new mysqli($host, $username, $password, $dbname);

if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$conn->set_charset( charset: 'utf8');
?>
```

Obrázok 2: Konfiguračný súbor
Zdroj: Vlastné spracovanie

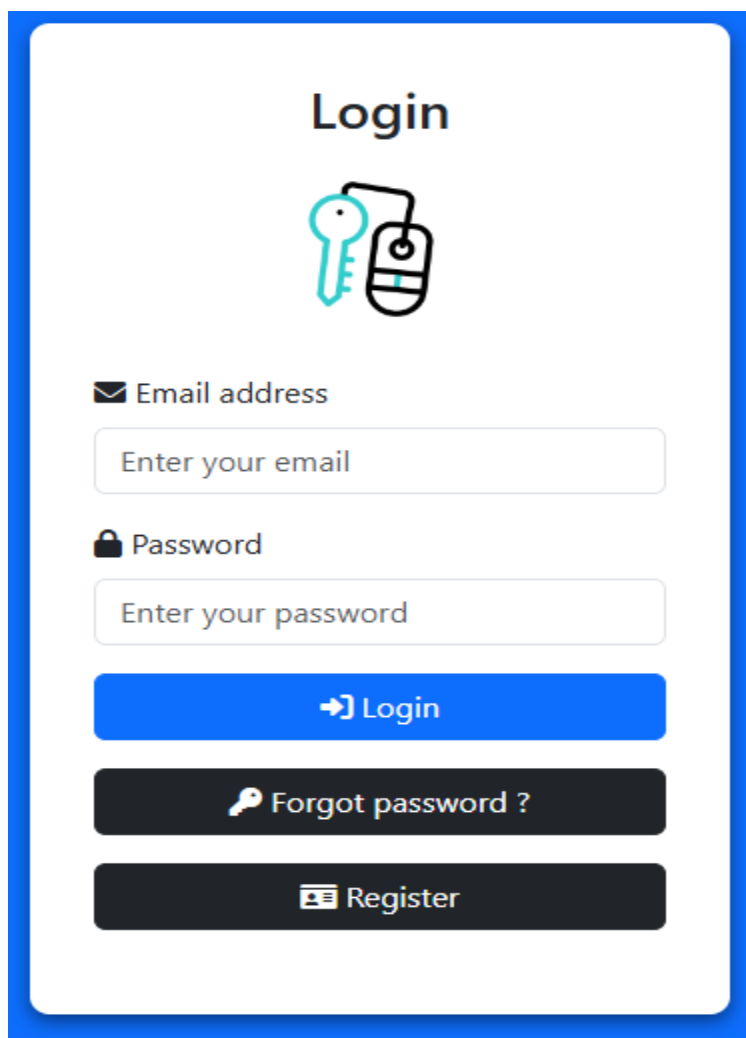
3.2 Frontend

Frontend predstavuje používateľské rozhranie aplikácie, ktoré umožňuje používateľom jednoduchú a intuitívnu interakciu s virtuálnou recepciou. Pri návrhu sme kládli dôraz na prehľadnosť, prispôsobivý dizajn a používateľskú skúsenosť, aby bola aplikácia ľahko

ovládateľná na rôznych zariadeniach, napríklad počítačoch, tabletoch, či mobiloch. Pri vývoji sme využili kombináciu HTML, Bootstrapu, CSS a JavaScriptu.

3.2.1 Login

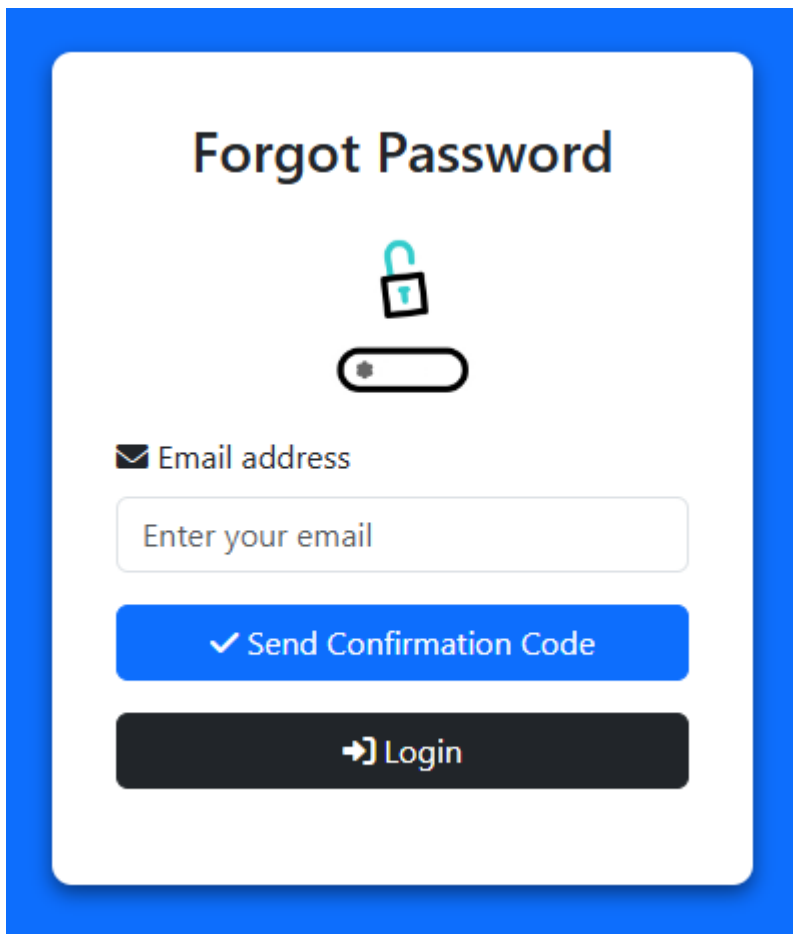
Prihlasovanie je kľúčovou súčasťou aplikácie virtuálnej recepcie, ktorá umožňuje bezpečný prístup k účtu, či samotnú interakciu s aplikáciou. Prihlasovacia stránka obsahuje pole pre e-mail, kde používateľ zadáva svoju e-mailovú adresu slúžiacu ako identifikátor účtu. Ďalej je tu vstupné pole pre heslo, ktorým sa overuje identita používateľa.




The image shows a login form with a blue border. At the top, the word "Login" is displayed in a bold, black font. Below it is an icon of a key and a smartphone. The form contains two input fields: "Email address" with a placeholder "Enter your email" and "Password" with a placeholder "Enter your password". Below the input fields are three buttons: a blue "Login" button with a right-pointing arrow icon, a dark grey "Forgot password ?" button with a key icon, and a dark grey "Register" button with a document icon.

Obrázok 3: Login
Zdroj: Vlastné spracovanie

Po vyplnení týchto údajov používateľ klikne na tlačidlo „Prihlásiť sa“, ktorým sa dostane do samotnej aplikácie. V prípade zabudnutého hesla je k dispozícii tlačidlo „Zabudnuté heslo?“, ktoré umožňuje resetovať heslo prostredníctvom e-mailu.



Forgot Password



Email address

Enter your email

✓ Send Confirmation Code

→ Login

Obrázok 4: Zabudnuté heslo
Zdroj: Vlastné spracovanie

Pre tých, ktorí ešte nemajú účet, je k dispozícii tlačidlo „Registrácia“, ktorým si vedľa účtu vytvoriť.

Register



✉ Email address

🔒 Password

🔒 Confirm Password

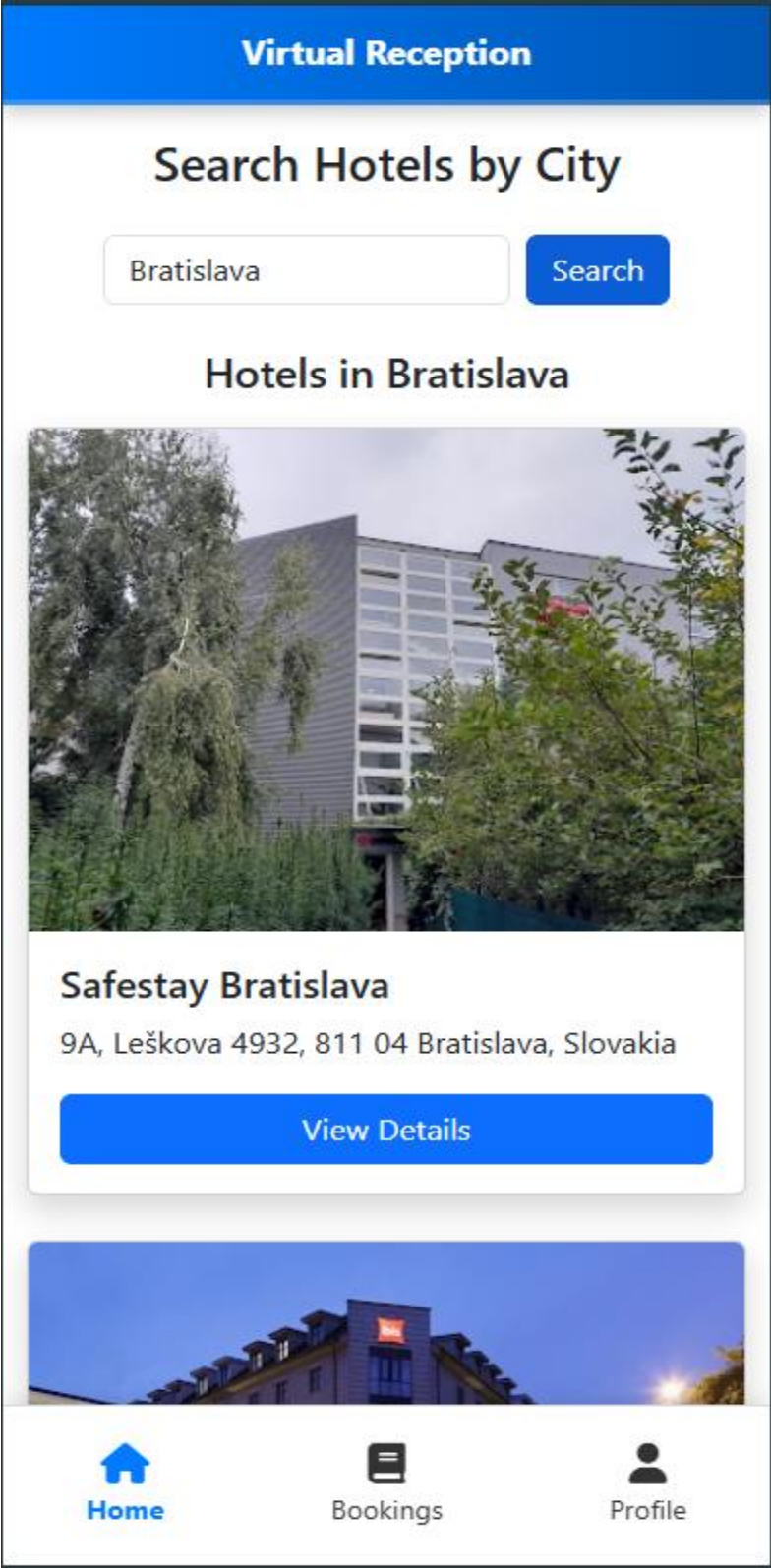
✓ Register

➔ Login

Obrázok 5: Registrácia
Zdroj: Vlastné spracovanie

3.2.1 Rezervácia

Aplikácia virtuálnej recepcie je navrhnutá tak, aby poskytovala používateľovi jednoduchý a efektívny spôsob správy rezervácií a interakcie s hotelovými službami. V záložke „domov“ môže používateľ vyhľadávať hotely podľa mesta, pričom po zadaní požiadavky sa zobrazí zoznam dostupných hotelov v zadanom mieste, spolu s kľúčovými informáciami, ako sú názov, lokalita, či hodnotenie. Pokiaľ žiadne mesto nevyberie, zobrazia sa dostupné možnosti v okolí. Po výbere konkrétneho hotela sa používateľ dostane k podrobnostiam a môže pokračovať v procese rezervácie.



Obrázok 6: Záložka domov
Zdroj: Vlastné spracovanie

Tlačidlo, ktoré sa nachádza pri každom hoteli v aplikácii, umožňuje používateľovi získať podrobnosti o vybranom ubytovaní. Po stlačení tlačidla sa zobrazí detailný prehľad, ktorý obsahuje informácie ako je webová adresa hotela, telefónne číslo na recepciu a hodnotenie hotela od ostatných používateľov. Tieto informácie sú nevyhnutné pre lepšiu orientáciu a rozhodovanie používateľa pri výbere ubytovania. Webová adresa poskytuje priamy prístup na stránku hotela, kde si môže používateľ prečítať ďalšie informácie. Telefónne číslo je k dispozícii pre prípad, že používateľ chce kontaktovať recepciu hotela a získať odpovede na konkrétne otázky. Hodnotenie hotela, ktoré zobrazuje spätnú väzbu od predchádzajúcich hostí, pomáha používateľovi pri hodnotení kvality ubytovania.

Hotel Details



Name: Safestay Bratislava

Address: 9A, Leškova 4932, 811 04 Bratislava-Staré Mesto, Slovakia

Rating: 4.1 ★

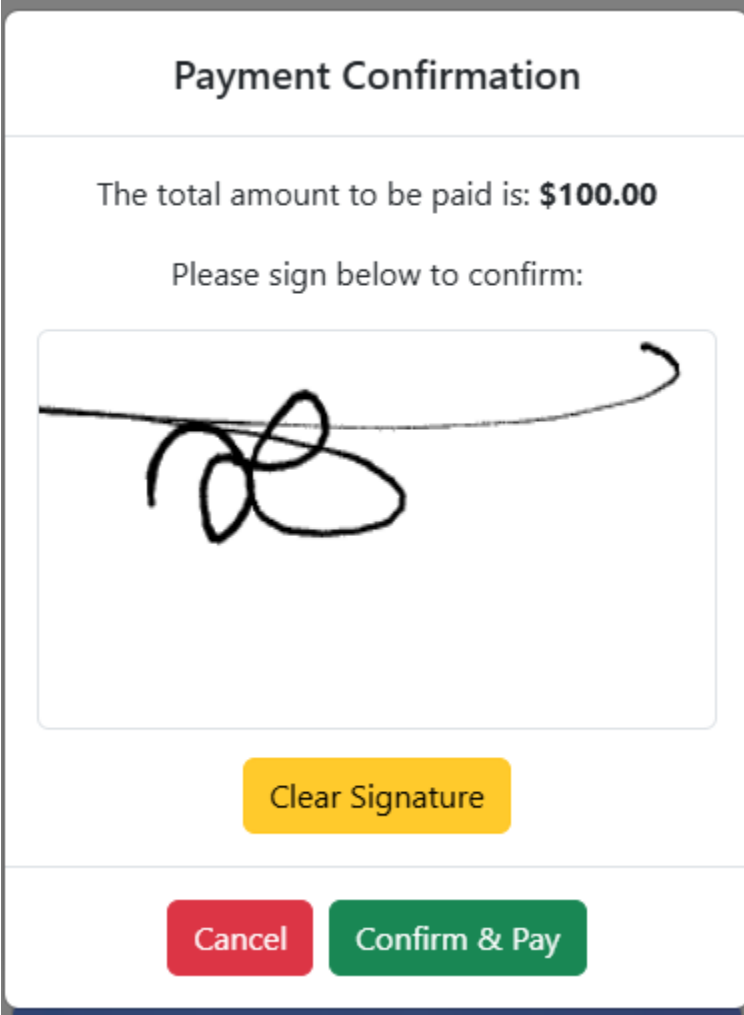
Phone: 02/330 002 37

Website:
<https://www.safestay.com/venue/safestay-bratislava-presidential-palace/>

[Close](#) [Online Reservation](#)

Obrázok 7: Detaily
Zdroj: Vlastné spracovanie

Po výbere hotela používateľ vyplní formulár, ktorý obsahuje informácie ako dátum príchodu, dátum odchodu, či počet osôb. Po zadaní všetkých potrebných údajov a potvrdení výberu prejde na platobnú bránu. Tam sa zobrazí prehľad rezervácie vrátane detailov o cene za ubytovanie. Pre demonštračné účely sme nastavili pevnú cenu ubytovania. Následne zákazník túto platbu elektronicky podpíše. Po úspešnom spracovaní platby je rezervácia potvrdená a používateľ obdrží potvrdenie o rezervácii cez e-mail, ktorý obsahuje všetky potrebné informácie o pobyte. Tento proces zabezpečuje prehľad o všetkých krokoch a podmienkach rezervácie, a zároveň poskytuje bezpečný spôsob platby za zvolené ubytovanie.



Payment Confirmation

The total amount to be paid is: **\$100.00**

Please sign below to confirm:

[Handwritten signature]

Clear Signature

Cancel Confirm & Pay

Obrázok 8: Potvrdenie rezervácie
Zdroj: Vlastné spracovanie

Po úspešnej rezervácii je používateľ presmerovaný do sekcie „rezervácie“, kde sa mu zobrazí zoznam všetkých jeho zaplatených ubytovaní. Každá rezervácia obsahuje detailné informácie o termíne príchodu, odchodu a stave rezervácie. Pokiaľ sa nachádzame v časovom rámci pre „check-in“, tlačidlo na potvrdenie príchodu je aktívne a je možné ho stlačiť, čím dokončí proces prihlásenia do hotela. Ak však používateľ nie je v rozmedzí stanoveného časového okna, tlačidlo je nedostupné a musí počkať. Okrem toho je tu aj tlačidlo, ktoré umožňuje zobrazit' zaplatené rezervácie. Po jeho stlačení sa zobrazí potvrdenie o zaplatenej rezervácii, kde sú uvedené všetky platobné detaily a potvrdenie o úhrade. Tento prehľad poskytuje jasný prehľad o všetkých jeho zaplatených a potvrdených rezerváciách, čo umožňuje efektívne sledovať pobyty a vyhnúť sa nedorozumeniam.

Reservation Details

Hotel Name: Safestay Bratislava

Hotel Address: 9A, Leškova 4932, 811 04
Bratislava-Staré Mesto, Slovakia

Full Name: David Hanulák

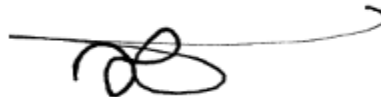
Email: test@testing.com

Phone: 0915443818

Guest Number: 2

Amount Paid: 100.00 €

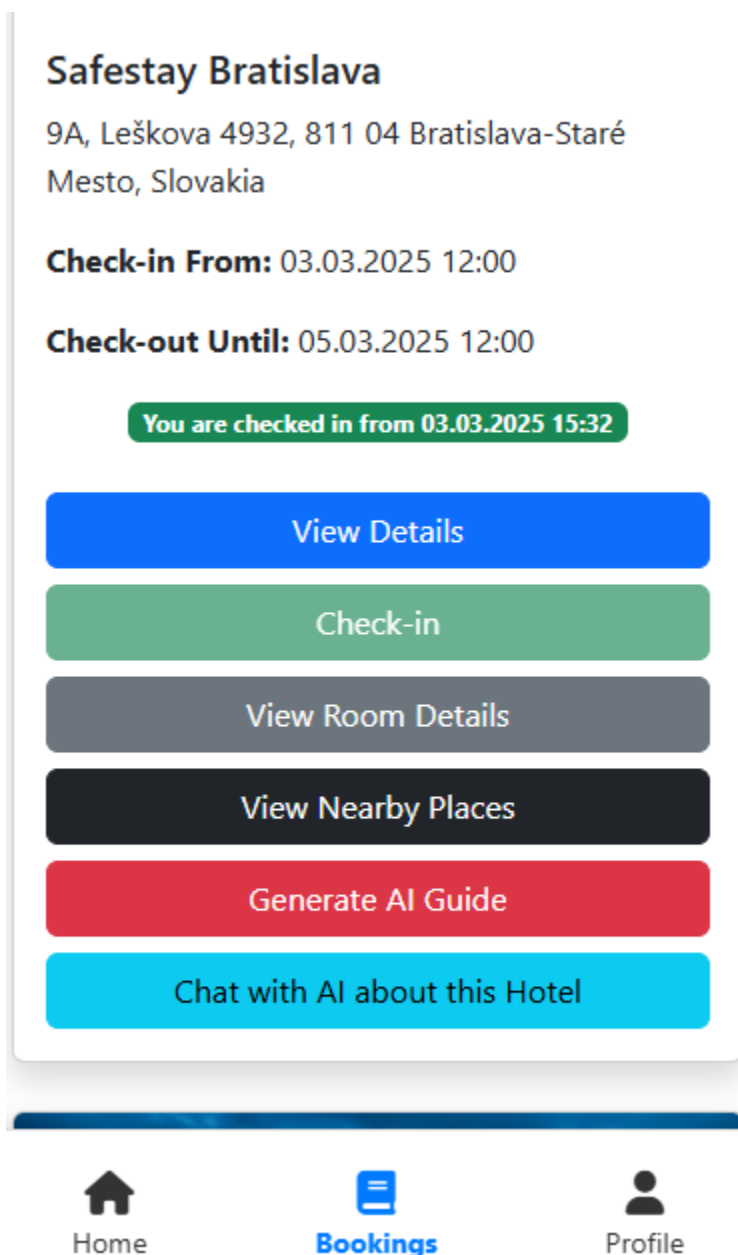
Signature:



Close

Obrázok 9: Detaily rezervácie
Zdroj: Vlastné spracovanie

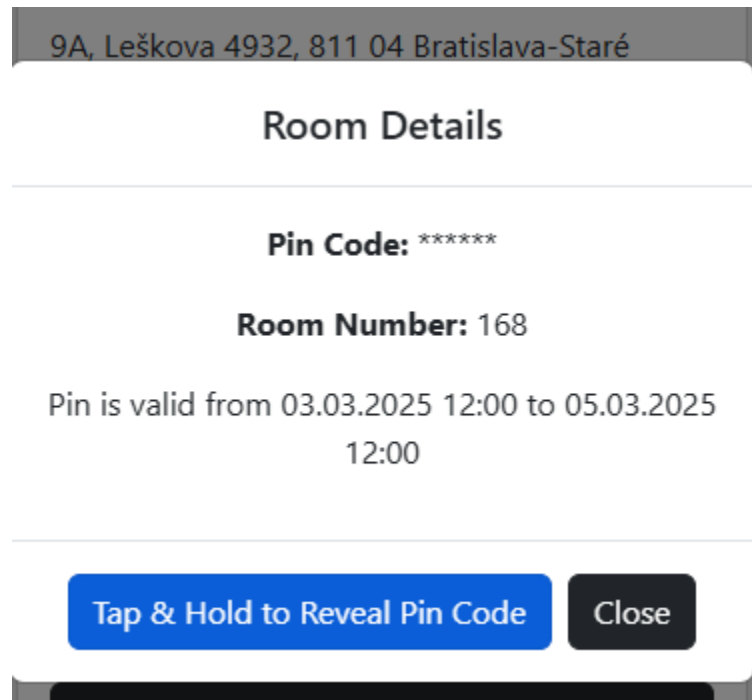
Keď používateľ vykoná „check-in“, zobrazia sa mu ďalšie možnosti ohľadom ubytovania. Môže si pozrieť izbu, v ktorej bude bývať, okolité miesta podľa Google Maps, či miestneho sprievodcu vygenerovaného pomocou umelej inteligencie. Taktiež vie priamo komunikovať s daným modelom o čomkoľvek, čo ho zaujíma.



Obrázok 10: Ďalšie možnosti
Zdroj: Vlastné spracovanie

3.2.2 *Detaily izby*

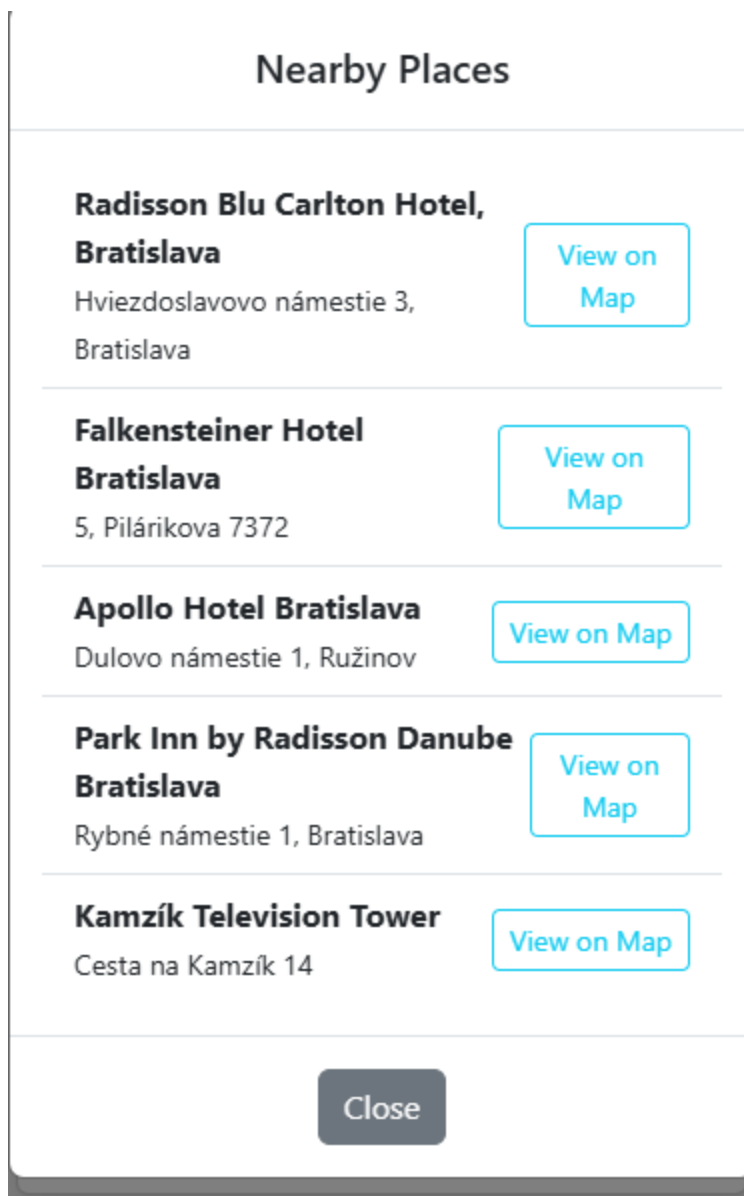
Tlačidlo s detailmi o izbe po stlačení zobrazuje kľúčové informácie, ktoré sú potrebné na pohodlný a bezpečný pobyt. Zobrazí sa jedinečný PIN kód, ktorý je potrebný na otvorenie dverí a platí iba počas pobytu. Navyše sme ho ochránili tak, že je dostupný iba počas držania tlačidla na odhalenie. Okrem toho je vidno aj číslo izby. Ukáže sa aj platnosť kódu, teda časový rámec, počas ktorého je tento kód platný.



Obrázok 11: *Detaily izby*
Zdroj: *Vlastné spracovanie*

3.2.3 *Okolie podľa Google Maps*

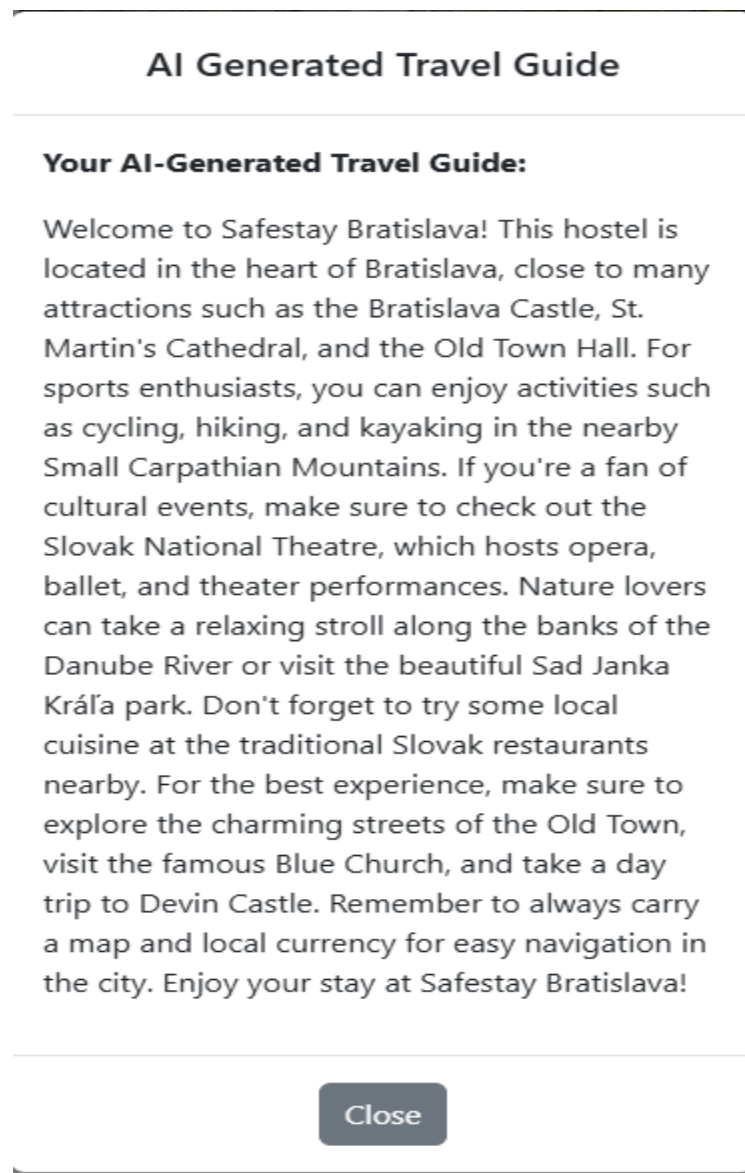
Ďalej je dostupná možnosť okolia ubytovania. Po stlačení sa ukáže používateľovi zoznam okolí hotela, ako sú reštaurácie, turistické atrakcie, obchody, kaviarne a iné zaujímavé miesta v blízkosti. Každé miesto je zobrazené spolu s krátkym popisom a odkazom na Google Maps, ktorý umožňuje používateľovi získať podrobné informácie o lokalite. Týmto spôsobom používateľ získa prehľad o možnostiach v okolí ubytovania a môže si naplánovať ďalšie aktivity počas svojho pobytu.



Obrázok 12: Okolie ubytovania
Zdroj: Vlastné spracovanie

3.2.4 Miestny sprievodca

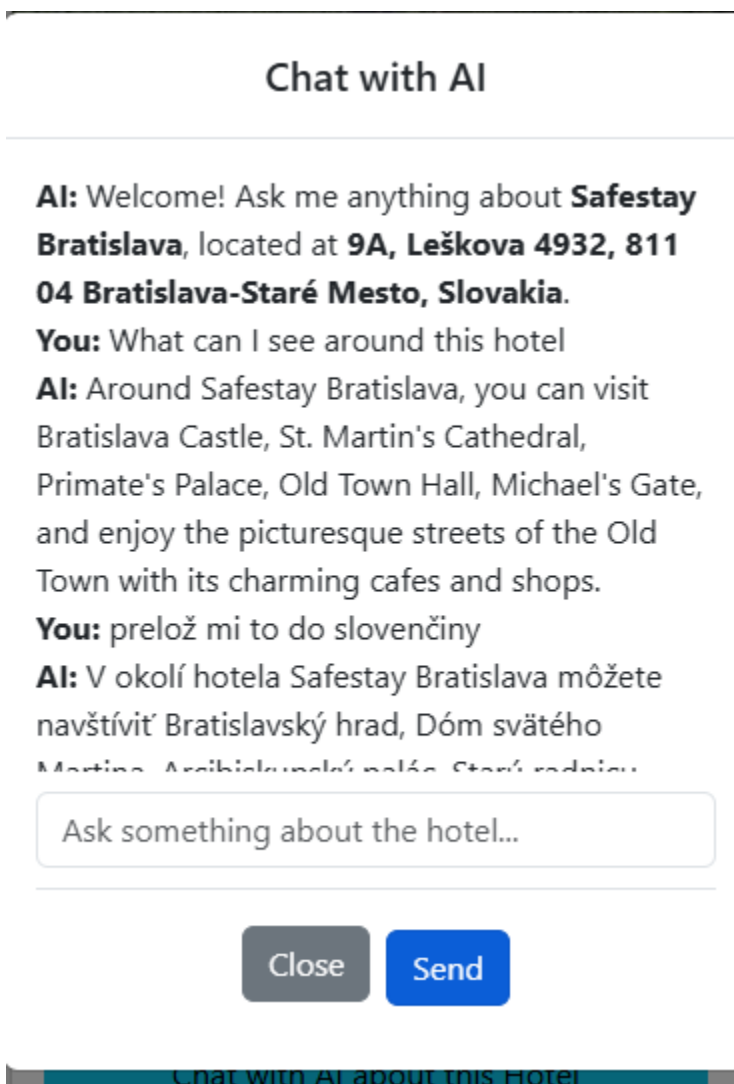
Tlačidlo generovania sprievodcu spustí jeho proces na základe daného ubytovania. Tento sprievodca je generovaný pomocou modelu GPT-4, ktorý analyzuje lokalitu, dostupné aktivity, atrakcie, reštaurácie a ďalšie faktory týkajúce sa konkrétneho hotela a jeho okolia. Výsledkom je odporúčanie na mieru, ktoré zahŕňa rôzne miesta, ktoré môžu byť pre používateľa zaujímavé, od turistických atrakcií až po skryté klenoty v okolí. Tento sprievodca sa neustále prispôsobuje na základe preferencií, čím mu poskytuje lepší zážitok a pomáha mu v orientácii na danom mieste.



Obrázok 13: Miestny sprievodca generovaný modelom GPT-4
Zdroj: Vlastné spracovanie

3.2.5 Komunikácia s modelom GPT-4

Používateľ vie po stlačení tohto tlačidla otvoriť interaktívny chat, kde môže komunikovať s modelom GPT-4, ktorý je špeciálne natrénovaný na informácie týkajúce sa daného hotela. Tento model poskytuje okamžité odpovede na otázky týkajúce sa ubytovania, dostupných služieb, vybavenia izby, okolitých miest a akýchkoľvek ďalších informácií, ktoré ho môžu zaujímať počas pobytu. Používateľ môže klásť otázky akéhokoľvek typu, v hociktorom jazyku. Model GPT-4 poskytne relevantné a presné odpovede na základe informácií o konkrétnom mieste a jeho okolí. Tento chat slúži ako rýchly a efektívny nástroj na získanie informácií a pomáha s akýmikoľvek požiadavkami počas pobytu.



Obrázok 14: Komunikácia s modelom GPT-4
Zdroj: Vlastné spracovanie

3.3 Backend

Backend aplikácie virtuálnej recepcie zabezpečuje spracovanie používateľských požiadaviek, komunikáciu s databázou a vykonávanie kľúčových operácií potrebných pre správne fungovanie systému. Na jeho implementáciu bol použitý serverový jazyk PHP a JavaScript, ktoré umožňujú dynamickú manipuláciu s údajmi a efektívnu správu rezervácií, používateľských účtov a interakcií s hotelovými službami. PHP zabezpečilo spracovanie požiadaviek, autentifikácie používateľov a prácu s databázou, zatiaľ čo JavaScript bol využitý na interaktívne prvky aplikácie ako je podpis rezervácie alebo napojenie na jednotlivé API. Na odosielanie e-mailových notifikácií, napríklad registrácie alebo resetovanie hesla bol použitý PHPMailer, ktorý spravuje spoľahlivú a bezpečnú komunikáciu prostredníctvom e-mailu.

3.3.1 Login

V tejto časti sme sa zamerali na autentifikáciu používateľov, ich registráciu a zabudnuté heslo. Po odoslaní prihlasovacieho formulára systém spracuje vstupné údaje a overí ich platnosť v databáze. Najprv sa skontroluje existencia e-mailovej adresy v tabuľke používateľov, následne sa overí zhodnosť zadaného hesla s bezpečne uloženým šifrovaným kódom (hashom). V prípade správnych prihlasovacích údajov sa e-mailová adresa uloží v relácii, čím sa zabezpečí pretrvávanie prihlásenia počas používania aplikácie. Nasleduje presmerovanie na hlavnú stránku. Ak sú údaje nesprávne, uloží sa chybové hlásenie a používateľ sa nedostane ďalej a formulár sa zobrazí znovu. Na zvýšenie bezpečnosti je implementovaná ochrana proti SQL injekciám prostredníctvom funkcie prepare, ako aj zabezpečené ukladanie hesiel pomocou hashovania.

```

<?php
session_start();
include('config.php');

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $email = $_POST['email'];
    $password = $_POST['password'];

    $sql = "SELECT * FROM login_data WHERE email = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param( types: "s", &var1: $email);
    $stmt->execute();
    $result = $stmt->get_result();

    if ($result->num_rows > 0) {
        $row = $result->fetch_assoc();

        if (password_verify($password, $row['password'])) {

            $_SESSION['email'] = $row['email'];

            header( header: "Location: homepage.php");
            exit();
        } else {
            $_SESSION['error'] = "Incorrect password!";
        }
    } else {
        $_SESSION['error'] = "No user found with that email!";
    }
    header( header: "Location: index.php");
    exit();
}
?>

```

Obrázok 15: Prihlasovacia funkcia
Zdroj: Vlastné spracovanie

Ďalší formulár zabezpečuje registráciu. Po jeho odoslání sa najskôr overí, či sa zadané heslá zhodujú, a následne skontroluje, či e-mail už neexistuje v databáze. Ak je e-mail nový, heslo sa zahashuje a používateľ sa uloží do databázy spolu s overovacím tokenom. Následne sa pomocou PHPMaileru odošle na zadaný e-mail potvrdzovacia správa s tokenom na overenie účtu. Ak registrácia prebehne úspešne, používateľ je presmerovaný na stránku, kde ho overí.

```
<?php

session_start();

include('config.php');

use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'phpmailer/src/Exception.php';
require 'phpmailer/src/PHPMailer.php';
require 'phpmailer/src/SMTP.php';

2 usages
function sendConfirmationEmail($email, $token) {
    $message = '';
    $error = '';

    try {
        $mail = new PHPMailer( exceptions: true);

        $mail->isSMTP();
        $mail->Host = 'smtp.gmail.com';
        $mail->SMTPAuth = true;
        $mail->Username = 'hanulakdavid@gmail.com';
        $mail->Password = 'xatthlfxuxuqkvak';
        $mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;
        $mail->Port = 587;|
```

Obrázok 16: PHP Mailer
Zdroj: Vlastné spracovanie

Posledný formulár v tejto časti zabezpečuje reset hesla prostredníctvom e-mailovej autentifikácie. Po odoslaní so zadanou e-mailovou adresou sa overí jej platný formát. Ak je správny, vygeneruje sa šesťmiestny overovací kód, ktorý sa uloží do relácie spolu s e-mailovou adresou používateľa. Následne sa podobne ako pri registrácii odošle e-mail s týmto kódom na danú adresu. Ak sa správa úspešne odošle, používateľ je presmerovaný na stránku, kde musí zadať overovací kód. V prípade chyby pri odosielaní e-mailu alebo neplatného formátu sa zobrazí chybová správa a používateľ je presmerovaný naspäť.

```
function sendConfirmationEmail($email) {
    global $message, $error;

    if (filter_var($email, filter: FILTER_VALIDATE_EMAIL)) {

        $confirmationCode = mt_rand(100000, 999999);

        $_SESSION['confirmation_code'] = $confirmationCode;
        $_SESSION['email'] = $email;

        try {
            $mail = new PHPMailer( exceptions: true);

            $mail->isSMTP();
            $mail->Host = 'smtp.gmail.com';
            $mail->SMTPAuth = true;
            $mail->Username = 'hanulakdavid@gmail.com';
            $mail->Password = 'xatthlfxuxugkvak';
            $mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;
            $mail->Port = 587;

            $mail->setFrom( address: 'hanulakdavid@gmail.com', name: 'Virtual Reception');
            $mail->addAddress($email);

            $mail->isHTML( isHtml: true);
            $mail->Subject = 'Password Reset Confirmation Code';
            $mail->Body = "Here is your confirmation code: <strong>$confirmationCode</strong>";
            $mail->AltBody = "Here is your confirmation code: $confirmationCode";
```

Obrázok 17: Zabudnuté heslo
Zdroj: Vlastné spracovanie

3.3.2 Zobrazenie ubytovania

Používateľ má dve možnosti, ako vyhľadávať. Buď podľa svojej aktuálnej polohy alebo manuálne zadaním mesta do vyhľadávacieho poľa. Po prihlásení sa vždy najskôr vyskúša hľadanie podľa polohy. Získa sa aktuálna poloha používateľa a následne sa začnú vyhľadávať hotely v blízkosti. Funkcia Najskôr skontroluje, či prehliadač podporuje lokalizáciu. Ak áno, získa zemepisnú šírku a dĺžku aktuálnej polohy. Tieto súradnice sa potom odovzdajú ďalšej funkcii, ktorá vykoná vyhľadanie hotelov na základe polohy. Ak sa poloha nepodarí získať, zobrazí sa chybové hlásenie.

```
function getUserLocationAndFetchHotels() :void {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition( successCallback: position : GeolocationPosition => {
      const latitude : number = position.coords.latitude;
      const longitude : number = position.coords.longitude;
      fetchNearbyHotels(latitude, longitude);
    }, errorCallback: error : GeolocationPositionError => {
      alert("Unable to retrieve your location.");
    });
  } else {
    alert("Geolocation is not supported by this browser.");
  }
}
```

Obrázok 18: Vyhľadávanie podľa GPS
Zdroj: Vlastné spracovanie

Druhou funkciou je hľadanie podľa mesta, ktoré používateľ zadá. Ak ide o nové vyhľadávanie, vymaže predchádzajúce výsledky. Následne sa vykoná požiadavka na server, ktorý získa hotely v danom meste pomocou Google Places API. Po získaní výsledkov sa pridajú do zoznamu a zobrazia na stránke. Ak sú k dispozícii ďalšie výsledky, tak sa načítajú po tom, čo používateľ prejde na koniec.

```
function fetchHotelsByCity(city, loadMore : boolean = false) :void {
  if (isFetching) return;
  isFetching = true;

  if (!loadMore) {
    hotels = [];
    displayedHotels = 0;
    lastFetchedPage = 0;
    cityName = city;
    document.getElementById('hotels-list').innerHTML = "<h4 class='text-center'>Hotels in " + city + "</h4><div class='row mt-3'></div>";
  }
}
```

Obrázok 19: Vyhľadávanie podľa mesta
Zdroj: Vlastné spracovanie

3.3.3 Rezervácia

Táto časť spracováva rezerváciu ubytovania. Po prijatí údajov používateľom sa tieto hodnoty bezpečne spracujú, aby sa zabránilo neoprávneným dopytom. Následne sa údaje vložia do databázy. V prípade chyby pri vkladaní sa vráti správa o neúspechu. Ak chýbajú povinné údaje, funkcia informuje o chýbajúcich poliach.

```
<?php
include 'config.php';

if (isset($_POST['hotel_id'], $_POST['full_name'], $_POST['email'], $_POST['tel_number'], $_POST['checkin_date'], $_POST['checkout_date'], $_POST['guest_number'])) {

    $hotel_id = mysqli_real_escape_string($conn, $_POST['hotel_id']);
    $full_name = mysqli_real_escape_string($conn, $_POST['full_name']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $tel_number = mysqli_real_escape_string($conn, $_POST['tel_number']);
    $checkin_date = mysqli_real_escape_string($conn, string: $_POST['checkin_date'] . ' 12:00:00');
    $checkout_date = mysqli_real_escape_string($conn, string: $_POST['checkout_date'] . ' 12:00:00');
    $guest_number = mysqli_real_escape_string($conn, $_POST['guest_number']);

    $query = "INSERT INTO checkins (hotel_id, full_name, email, phone, checkin_date, checkout_date, guest_number)
VALUES ('$hotel_id', '$full_name', '$email', '$tel_number', '$checkin_date', '$checkout_date', '$guest_number')";

    if (mysqli_query($conn, $query)) {
        $checkin_id = mysqli_insert_id($conn);

        echo json_encode([
            'success' => true,
            'id' => $checkin_id,
            'hotel_id' => $hotel_id,
            'full_name' => $full_name
        ]);
    } else {
        echo json_encode(['success' => false, 'message' => 'Failed to insert reservation.']);
    }

    mysqli_close($conn);
} else {
    echo json_encode(['success' => false, 'message' => 'Missing required fields.']);
}
?>
```

Obrázok 20: Potvrdenie rezervácie
Zdroj: Vlastné spracovanie

Súčasťou je aj spracovanie platby s požiadavkou na podpis pred jej potvrdením. Pri odoslaní platby sa najskôr skontroluje, či používateľ poskytol podpis. Ak nie, zobrazí sa výzva na jeho pridanie. Ak bola platba podpísaná, jej dáta sa odošlú spolu s ďalšími na server cez POST požiadavku. Server následne spracuje platbu, vráti odpoveď a ak je platba úspešná, zobrazí okno s potvrdením úspechu. Po chvíli je používateľ presmerovaný na stránku rezervácií. V prípade neúspechu platby sa zobrazí chybové hlásenie.

```

var canvas :HTMLInputElement = document.getElementById( elementId: "signaturePad");
var signaturePad :SignaturePad = new SignaturePad(canvas);

document.getElementById( elementId: "clearSignature").addEventListener( type: "click", listener: function () :void {
    signaturePad.clear();
});

document.getElementById( elementId: "confirmPayment").addEventListener( type: "click", listener: function () :void {
    var checkinId = document.getElementById( elementId: "hiddenCheckinId").value;
    var amount = document.getElementById( elementId: "hiddenAmount").value;

    // Ensure the user has signed
    if (signaturePad.isEmpty()) {
        alert("Please provide a signature before confirming payment.");
        return;
    }

    var signatureImage :string = signaturePad.toDataURL( type: "image/png");

    var paymentData :FormData = new FormData();
    paymentData.append( name: "checkin_id", checkinId);
    paymentData.append( name: "amount", amount);
    paymentData.append( name: "paid", value: true);
    paymentData.append( name: "signature", signatureImage); |

```

Obrázok 21: Platba
Zdroj: Vlastné spracovanie

3.3.4 Detaily rezervácie

Táto časť umožňuje zobraziť detaily o rezervácii a ďalšie hotelové informácie. Na začiatku sa overí, či je používateľ prihlásený. Ak nie, je presmerovaný na prihlásenie. Získa sa ID rezervácie a hotela z parametrov a následne sa vykoná dopyt do databázy na získanie rezervácie, kde sa skontroluje, či sa nachádza v databáze. Ak neexistuje, vráti sa chybové hlásenie. Ak existuje, skript pomocou Google Places API získa podrobnosti o hoteli. Ak API odpovie úspešne, tieto údaje sú spracované a priradené k rezervácii. Nakoniec sú všetky údaje skombinované a odoslané späť na frontend, aby sa zobrazili používateľovi.

```

<?php
session_start();
include('config.php');

if (!isset($_SESSION['email'])) {
    $_SESSION['error'] = "You must be logged in to view your bookings.";
    header( "Location: index.php" );
    exit();
}

$email = $_SESSION['email'];
$reservationId = $_GET['reservation_id'];
$hotelId = $_GET['hotel_id'];

$sql = "SELECT * FROM checkins WHERE id = ? AND email = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param( 'is', &var: $reservationId, &var: $email );
$stmt->execute();
$result = $stmt->get_result();
$reservation = $result->fetch_assoc();

if (!$reservation) {
    echo json_encode(['error' => 'Reservation not found']);
    exit();
}

$googleApiKey = 'AIzaSy8HCAaztWA0wVFS4HcoZcaznKTZZz6d8o!';
$url = "https://maps.googleapis.com/maps/api/place/details/json?place_id={$hotelId}&fields=name,rating,formatted_phone_number,website,formatted_address,photos,geometry&key={$googleApiKey}";

```

Obrázok 22: Funkcia na zobrazenie detailov
Zdroj: Vlastné spracovanie

3.3.5 Detaily izby

Každá rezervácia obsahuje možnosť zobrazenia detailov, akými sú číslo izby a prístupový kód. Funkcia zobrazí modálne okno z Bootstrap knižnice s obsahom po kliknutí na príslušné tlačidlo. Dátumy príchodu a odchodu sme naformátovali, aby boli zrozumiteľné. Kód zabezpečuje aj maskovanie, pričom skutočný PIN je uložený, ale zobrazí sa až po podržaní príslušného tlačidla, aby sa ochránila bezpečnosť.

```

window.showRoomDetails = function(pinCode, roomNumber, checkinDate, checkoutDate, button) :void {

    document.getElementById( elementId: 'roomNumber' ).innerText = roomNumber;
    const options :{...} = { day: '2-digit', month: '2-digit', year: 'numeric', hour: '2-digit', minute: '2-digit' };

    document.getElementById( elementId: 'checkindate' ).innerText = new Date( checkinDate )
        .toLocaleString( locales: 'en-GB', options )
        .replace( searchValue: /\//g, replaceValue: '.' )
        .replace( searchValue: ', ', replaceValue: '' );

    document.getElementById( elementId: 'checkoutdate' ).innerText = new Date( checkoutDate )
        .toLocaleString( locales: 'en-GB', options )
        .replace( searchValue: /\//g, replaceValue: '.' )
        .replace( searchValue: ', ', replaceValue: '' );

    document.getElementById( elementId: 'maskedPinCode' ).innerText = '*****';
    document.getElementById( elementId: 'actualPinCode' ).innerText = pinCode;

    lastFocusedButton = button;

    |
    var myModal :bootstrap.Modal = new bootstrap.Modal( document.getElementById( elementId: 'roomDetailsModal' ) );
    myModal.show();
};

```

Obrázok 23: Maskovanie PIN kódu
Zdroj: Vlastné spracovanie

3.3.6 Okolie podľa Google Maps

Pomocou tejto funkcie sa zobrazí zoznam okolia daného hotela pomocou Google Places API. Najprv sa skontroluje, či sú dostupné geografické súradnice hotela. Následne sa prostredníctvom požiadavky na PHP skript získa zoznam miest v okolí zadaných súradníc. Po úspešnom získaní dát zo servera sa získa zoznam miest. Zobrazených ich je maximálne päť, pričom každá položka v zozname obsahuje názov miesta, adresu a tlačidlo, ktoré umožňuje zobraziť dané miesto na mape. Ak API nevráti žiadne výsledky alebo dôjde k chybe počas načítania dát, zobrazí sa chybové hlásenie.

```
function showNearbyRestaurants(hotelId, latitude, longitude) :void {
  if (!latitude || !longitude) {
    alert("Hotel location not available.");
    return;
  }

  // Show the modal first with a loading spinner
  document.getElementById( 'elementId: 'nearbyPlaces').innerHTML = "";
  document.getElementById( 'elementId: 'loadingSpinner').style.display = "flex";

  var myModal :bootstrap.Modal = new bootstrap.Modal(document.getElementById( 'elementId: 'nearbyRestaurantsModal'));
  myModal.show();

  // Fetch data from the proxy PHP script
  const url :string = `google_places_proxy.php?latitude=${latitude}&longitude=${longitude}`;

  fetch(url) Promise<Response>
    .then(response : Response => response.json()) Promise<any>
    .then(data => {
      document.getElementById( 'elementId: 'loadingSpinner').style.display = "none"; // Hide spinner

      if (data.status === "OK") {
        let placesList :string = "<ul class='list-group list-group-flush'>";

```

Obrázok 24: Okolité miesta podľa Google Maps
Zdroj: Vlastné spracovanie

3.3.7 Miestny sprievodca

Procedúra slúži na generovanie cestovného sprievodcu pre turistov, ktorí navštívia konkrétny hotel, pomocou modelu GPT-4. Vytvorí sa vstup, ktorý obsahuje názov hotela a jeho adresu. Na jeho základe model vygeneruje text o turistických atrakciách v okolí hotela, vrátane historických pamiatok, športových aktivít, kultúrnych udalostí, prírodných miest a tipov na najlepšie zážitky. Kód následne vykoná požiadavku na API, aby sa získala odpoveď. Po jej

získaní sa vygenerovaný text zobrazí v príslušnom okne. Ak sa pri generovaní sprievodcu vyskytne chyba alebo ak model nevráti žiadny text, používateľ uvidí chybové hlásenie.

```
function generateAIGuide(hotelName, hotelAddress) :void {  
  
  const prompt :string = `Generate a travel guide for tourists visiting ${hotelName}, located at ${hotelAddress}.  
  
  const apiData :{...} = {  
    model: "gpt-4",  
    messages: [{  
      role: "user",  
      content: prompt  
    }],  
    max_tokens: 300,  
  };  
};
```

Obrázok 25: Nastavenie modelu GPT-4
Zdroj: Vlastné spracovanie

3.3.8 Komunikácia s modelom GPT-4

V tejto práci je implementovaná aj funkcia na vytvorenie komunikačného rozhrania, kde používateľ môže komunikovať s AI asistentom o konkrétnom hoteli. Funkcia otvorí okno, ktoré obsahuje úvodnú správu. Taktiež drží aj históriu správ. Ďalej spracováva požiadavku používateľa a odosiela ju do modelu GPT-4. Po získaní odpovede sa táto odpoveď pridá do rozhrania a zobrazuje sa v okne. Okrem toho sa odpoveď uloží do histórie chatu, čím sa zabezpečí kontinuita konverzácie.

```

let chatHistory :any[] = [];

Show usages
function openChatModal(hotelName, hotelAddress) :void {

    chatHistory = [
        {
            role: "system",
            content: `You are an assistant who knows everything about the hotel named ${hotelName}, located at ${hotelAddress}.`
        }
    ];

    var myModal :bootstrap.Modal = new bootstrap.Modal(document.getElementById( 'chatModal'));
    myModal.show();

    const chatMessages :HTMLElement = document.getElementById( 'chatMessages');
    chatMessages.innerHTML = `<div><strong>AI:</strong> Welcome! Ask me anything about <b>${hotelName}</b>, located at <b>${hotelAddress}</b>.`;
}

Show usages
function sendMessage() :void {
    const userMessage = document.getElementById( 'userMessage').value;

    if (userMessage.trim() === "") return;

    const chatMessages :HTMLElement = document.getElementById( 'chatMessages');

    chatMessages.innerHTML += `<div><strong>You:</strong> ${userMessage}</div>`;
    document.getElementById( 'userMessage').value = "";
}

```

Obrázok 26: Komunikačná funkcia s modelom
Zdroj: Vlastné spracovanie

3.4 WebView

V poslednom štádiu práce sme túto aplikáciu presunuli na webový hosting a následne na operačný systém Android. Použili sme komponent WebView na zobrazenie webovej stránky a zároveň sme umožnili prístup k funkciám ako GPS a JavaScript. Pri spustení aplikácie sa najprv skontroluje, či má aplikácia povolenie na prístup k lokalizačným službám. Ak ešte nebolo udelené, požaduje ho od používateľa. Po získaní povolení aplikácia zobrazí webovú stránku v rámci komponentu WebView. Ten je nakonfigurovaný tak, aby umožnil spúšťanie JavaScriptu, používanie lokalizačných služieb, či prístup k lokálnemu úložisku. Aplikácia je nastavená aj na spracovanie Secure Sockets Layer (SSL) chýb, čo je užitočné pri práci so stránkami s neplatnými certifikátmi.

```

AndroidView(
    modifier = Modifier.fillMaxSize(),
    factory = { context ->
        WebView(context).apply {

            setBackgroundColor(android.graphics.Color.parseColor("colorString: "#0d6efd"))

            settings.javaScriptEnabled = true
            settings.domStorageEnabled = true
            settings.setGeolocationEnabled(true)
            settings.loadWithOverviewMode = true
            settings.useWideViewPort = true
            settings.allowFileAccess = true
            settings.allowContentAccess = true

            webViewClient = object : WebViewClient() {
                @SuppressWarnings("WebViewClientOnReceivedSslError")
                override fun onReceivedSslError(
                    view: WebView?,
                    handler: SslErrorHandler?,
                    error: SslError?
                ) {
                    handler?.proceed()
                }
            }

            webChromeClient = object : WebChromeClient() {
                override fun onGeolocationPermissionsShowPrompt(
                    origin: String?,
                    callback: GeolocationPermissions.Callback?
                ) {
                    callback?.invoke(origin, allow: true, retain: false)
                }
            }

            loadUrl(uri: "https://diplo.msvelehradska.sk")
        }
    }
)

```

Obrázok 27: WebView component
Zdroj: Vlastné spracovanie

3.5 Diskusia

Vytvorenie aplikácie pre virtuálnu recepciu predstavuje významný pokrok v zjednodušení a zefektívnení interakcií medzi zákazníkmi a hotelmi alebo inými ubytovacími zariadeniami. Táto aplikácia umožňuje používateľom flexibilne a intuitívne spravovať svoje rezervácie a získať informácie o hoteloch či okolitých reštauráciách a zaujímavých miestach. Výsledky ukazujú, že použitie pokročilých technológií ako je integrácia s Google Places API a použitie modelov umelej inteligencie na generovanie cestovných sprievodcov alebo poskytovanie podpory cez správy ponúka vynikajúcu používateľskú skúsenosť.

Dôležitým aspektom tejto aplikácie je možnosť dynamicky generovať informácie podľa preferencií používateľov, ako sú napríklad odporúčania pre turistov na základe konkrétnych kritérií, ako sú lokalizačné údaje alebo údaje o hoteli. Táto flexibilita poskytuje používateľom cenné informácie v reálnom čase, čím sa zlepšuje celková spokojnosť s ubytovaním a poskytovanými službami. Použitie GPT-4 na poskytovanie informácií o hoteli a jeho službách je ďalšou zaujímavou možnosťou, ktorá prispieva k zlepšeniu používateľskej skúsenosti a efektívnosti.

Avšak, aj napriek výhodám existujú aj niektoré výzvy a obmedzenia. Napríklad, schopnosť generovať presné informácie závisí od kvality natrénovania jednotlivých modelov, čo môže ovplyvniť presnosť a úplnosť. Okrem toho, implementácia pokročilých funkcií ako je generovanie sprievodcov alebo poskytovanie podpory cez správy môže byť náročná na technické zručnosti a vyžaduje si správnu konfiguráciu a neustálu údržbu.

Záver z tejto práce ukazujú, že vytvorenie virtuálnej recepcie môže významne zlepšiť interakciu medzi používateľmi a hotelmi, čím sa zjednoduší proces rezervácie, získavania informácií a podpory. Budúci výskum by sa mohol zamerať na rozšírenie možností tejto aplikácie, ako je integrácia s ďalšími systémami hotela alebo vylepšenie rozhrania pre používateľov. Vhodným krokom by tiež bolo skúmanie využitia ďalších technológií, ako sú rozšírená realita alebo odporúčania na základe správania používateľov, ktoré by ešte viac vylepšili celkový zážitok tejto virtuálnej recepcie.

ZÁVER

V tejto diplomovej práci sme sa zamerali na návrh a implementáciu virtuálnej recepcie, ktorá umožňuje používateľom efektívne spravovať rezervácie, komunikovať s modelom umelej inteligencie a získavať informácie o ubytovaní a okolí. Cieľom bolo vytvoriť riešenie, ktoré zjednoduší interakcie medzi zákazníkmi a hotelmi, čím sa zlepší celkový používateľský zážitok. Implementovali sme rôzne funkcie ako generovanie sprievodcov a komunikáciu pomocou správ s umelou inteligenciou, ktoré používateľom umožňujú rýchlo nájsť potrebné informácie a získať asistenciu.

Práca preukázala, že kombinácia moderných technológií, ako sú API pre lokalizáciu, Google Places a GPT-4 výrazne zlepšuje používateľský zážitok a zefektívňuje proces rezervácií a interakcie s hotelmi. Virtuálna recepcia tak ponúka komfort, rýchlosť a flexibilitu, čo prispieva k vyššej spokojnosti zákazníkov. Správne navrhnuté používateľské rozhranie zabezpečuje rýchly prístup k informáciám a efektívnu komunikáciu.

V budúcnosti by bolo možné túto aplikáciu ďalej rozširovať, napríklad pridaním funkcií ako odporúčania alebo integráciou hlasových asistentov či rozšírenej reality. Tento výskum môže slúžiť ako prínos pre oblasť virtuálnych recepcií a prispieť k zlepšeniu služieb v cestovnom ruchu. Tiež môže byť užitočný pre vývojárov, ktorí sa zaujímajú o využitie technológie umelej inteligencie a API v hotelových systémoch.

Na záver môžeme konštatovať, že virtuálna recepcia je efektívnym nástrojom, ktorý výrazne zlepšuje komunikáciu medzi hotelmi a zákazníkmi. Výsledky tejto práce môžu prispieť k ďalšiemu rozvoju tohto inovatívneho riešenia a poskytnúť inšpiráciu pre budúcich vývojárov a podnikateľov v oblasti cestovného ruchu a technológií.

Zoznam použitej literatúry

1. Altexsoft. Generative AI Models Explained. [online] 2024 [cit. 2024-09-05] Dostupné na internete: <https://www.altexsoft.com/blog/generative-ai/>
2. Amit, Jolene. The Future of AI in Hospitality: 6 Trends Pushing Innovation. [online] 2024 [cit. 2024-09-10] Dostupné na internete: <https://aiola.ai/blog/future-of-ai-in-hospitality/>
3. Autor, Clara. 8 Best Mobile App Development Platforms for Building an App. [online] 2024 [cit. 2024-10-22] Dostupné na internete: <https://www.designrush.com/agency/mobile-app-design-development/trends/mobile-application-development-platform>
4. Barten, Martijn. Hotel Software: The Most Important Software Solutions for Hotels. [online] 2024 [cit. 2024-02-03] Dostupné na internete: <https://www.revfine.com/hotel-software/>
5. Bigelow, Stephen. What are the types of APIs and their differences. [online] 2024 [cit. 2024-12-23] Dostupné na internete: <https://www.techtarget.com/searcharchitecture/tip/What-are-the-types-of-APIs-and-their-differences>
6. Coursera. AI Ethics: What It Is, Why It Matters, and More. [online] 2025 [cit. 2025-02-03] Dostupné na internete: <https://www.coursera.org/articles/ai-ethics>
7. GeeksForGeeks Oracle Database - An Introduction. [online] 2024 [cit. 2024-09-19] Dostupné na internete: <https://www.geeksforgeeks.org/oracle-database-an-introduction/>
8. Hajji, Johan. Exploring Online Booking And Mobile Apps For Hotel Management In The Digital Age. [online] 2023 [cit. 2023-03-07] Dostupné na internete: <https://www.forbes.com/councils/forbesbusinesscouncil/2023/03/07/exploring-online-booking-and-mobile-apps-for-hotel-management-in-the-digital-age/>
9. HotelTechReport. The 2024 State of Hotel Guest Tech Report Unveils Key Trends Shaping the Future of Hospitality. [online] 2024 [cit. 2024-08-12] Dostupné na internete: https://hoteltechreport.com/news/2025-state-of-hotel-guest-technology-report?utm_source=chatgpt.com
10. Ionescu, Stefan. Best cloud database of 2025. [online] 2025 [cit. 2025-02-12] Dostupné na internete: <https://www.techradar.com/best/best-cloud-databases>

11. JetBrains. The Six Most Popular Cross-Platform App Development Frameworks. [online] 2025 [cit. 2025-02-14] Dostupné na internete: <https://www.jetbrains.com/help/kotlin-multiplatform-dev/cross-platform-frameworks.html>
12. Kamenchuk, Antonia. The future of online booking platforms. [online] 2024 [cit. 2024-05-03] Dostupné na internete: <https://easyweek.io/the-future-of-online-booking-systems.html>
13. LesRoches. Understanding automation in the hospitality industry. [online] 2024 [cit. 2024-06-15] Dostupné na internete: <https://lesroches.edu/blog/what-is-automation-in-hospitality-industry/>
14. Lijewski, Dawid. Best IDEs for Android Development in 2024. [online] 2024 [cit. 2024-06-14] Dostupné na internete: <https://www.netguru.com/blog/best-ides-for-android-development>
15. Müller, Tim. The hotel reception in transformation. [online] 2024 [cit. 2024-04-17] Dostupné na internete: <https://straiv.io/en/blog/the-hotel-reception-in-transformation/>
16. Nguyen, Tam. What are APIs? A computer scientist explains the data sockets that make digital life possible. [online] 2023 [cit. 2023-09-26] Dostupné na internete: <https://theconversation.com/what-are-apis-a-computer-scientist-explains-the-data-sockets-that-make-digital-life-possible-213042>
17. SAP. What is AI ethics. [online] 2024 [cit. 2024-08-09] Dostupné na internete: <https://www.sap.com/resources/what-is-ai-ethics>
18. Schwartz, Lisa. AI in Hospitality: Advantages & Use Cases. [online] 2025 [cit. 2025-01-06] Dostupné na internete: <https://www.netsuite.com/portal/resource/articles/business-strategy/ai-hospitality.shtml>
19. Shepard, Alexandria. Types of APIs and Use Cases. [online] 2023 [cit. 2023-05-02] Dostupné na internete: <https://konghq.com/blog/learning-center/different-api-types-and-use-cases>
20. Shulgin, Aleksandr. The Relevance of Mobile App Development for Businesses. [online] 2023 [cit. 2023-11-22] Dostupné na internete: <https://www.projecttimes.com/articles/the-relevance-of-mobile-app-development-for-businesses/>
21. Sharma, Akhilesh. What Is IDE and How It Is Important In Mobile App Development. [online] 2023 [cit. 2023-06-29] Dostupné na internete:

- <https://www.a3logics.com/blog/what-is-ide-and-how-it-is-important-in-mobile-app-development/>
22. Yaniv, Yehuda. Top 7 Cloud Databases. [online] 2022 [cit. 2022-09-15] Dostupné na internete: <https://www.dbmaestro.com/blog/database-automation/top-7-cloud-databases>
 23. Zendesk. Zendesk Customer Experience Trends Report 2020. [online] 2020 [cit. 2020-03-10] Dostupné na internete: <https://d1eipm3vz40hy0.cloudfront.net/pdf/cxtrends/cx-trends-2020-full-report.pdf>
 24. Zhang, Mounil. Best Mobile App Development Platforms to Build Top-Notch Apps. [online] 2024 [cit. 2024-11-14] Dostupné na internete: <https://radixweb.com/blog/mobile-app-development-platforms>