

EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY

Evidenčné číslo: 103004/B/2024/36145173628206084

Ubytovací systém pre študentov Ekonomickej Univerzity
v Bratislave
Bakalárska práca

2024

Magdaléna Rodáková

EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY

Ubytovací systém pre študentov Ekonomickej Univerzity
v Bratislave
Bakalárska práca

Študijný program: Hospodárska informatika
Študijný odbor: Ekológia a manažment
Školiace pracovisko: Katedra aplikovanej informatiky
Vedúci záverečnej práce: Ing. Peter Procházka

Bratislava 2024

Magdaléna Rodáková



Ekonomická univerzita v Bratislave
Fakulta hospodárskej informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Magdaléna Rodáková
Študijný program: hospodárska informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: ekonómia a manažment
Typ záverečnej práce: Bakalárska záverečná práca
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický
Názov: Ubytovací systém pre študentov Ekonomickej Univerzity v Bratislave.
Anotácia: Študent v BP vytvorí webovú aplikáciu prepojenú s databázou prostredníctvom ktorej si študenti, ktorým bolo pridelené ubytovanie, budú vedieť vybrať internát a taktiež aj konkrétnu izbu, kde budú ubytovaní v ďalšom akademickom roku. Práca bude riešiť celý proces od poslania žiadosti o ubytovanie až po následné vybratie si izby. Študent aplikáciu otestuje, poukáže na jeho výhody a porovná s inými riešeniami.

Vedúci: Ing. Peter Procházka
Katedra: KAI FHI - Katedra aplikovanej informatiky
Vedúci katedry: Ing. Mgr. Peter Schmidt, PhD.

Dátum zadania: 03.03.2023

Dátum schválenia: 10.03.2023

osoba zodpovedná za realizáciu študijného programu

ČESTNÉ PREHLÁSENIE

Čestne prehlasujem, že som záverečnú prácu vypracovala samostatne, a že všetku použitú literatúru, ktorú som použila, som uviedla v zozname použitých zdrojov.

Dátum:

.....
(podpis študenta)

POĎAKOVANIE

Týmto spôsobom by som sa chcela poďakovať školiteľovi Ing. Petrovi Procházkovi za pomoc, rady, usmernenie a vedenie v mojej bakalárskej práci. Jeho odborné a praktické znalosti boli obrovským prínosom. V neposlednom rade sa chcem poďakovať všetkým, ktorí prispeli svojimi informáciami a radami k vzniku tejto práce.

ABSTRAKT

RODÁKOVÁ, Magdaléna: *Ubytovací systém pre študentov Ekonomickej Univerzity v Bratislave*. – Ekonomická univerzita v Bratislave. Fakulta hospodárskej informatiky; Katedra aplikovanej informatiky. – Vedúci záverečnej práce: Ing. Peter Procházka, PhD - Bratislava: FHI, 2024, 48s.

Cieľom tejto bakalárskej práce je navrhnúť a implementovať webovú aplikáciu pre študentov, ktorá im umožní vybrať si na internáte konkrétnu izbu pre daný akademický rok. Práca dôkladne analyzuje súčasné riešenia dostupné na trhu, porovnáva ich výhody a nevýhody a navrhuje nový postup k riešeniu tohto problému. Práca je rozdelená do niekoľkých kapitol. Prvá kapitola sa venuje súčasnému stavu problému vrátane dostupných riešení na trhu a ich podrobnému porovnaniu. V druhej časti sa popisuje cieľ práce a vyhodnotenie dotazníka, na základe ktorého sa vytvorí návrh nového riešenia danej problematiky. Práca poskytuje podrobný opis krokov od podania žiadosti o ubytovanie po výber konkrétnej izby. Posledná kapitola popisuje proces vytvárania samotnej webovej aplikácie a jej integráciu s databázou. Záver sumarizuje celý proces riešenia problému a hodnotí výsledky práce. Výsledkom riešenia je plne funkčná webová aplikácia, ktorá umožňuje študentom efektívne a pohodlne vybrať si ubytovanie a izbu na akademický rok.

Kľúčové slová: webová aplikácia, react.js, rezervačný systém

ABSTRACT

RODÁKOVÁ, Magdaléna: *Ubytovací systém pre študentov Ekonomickej Univerzity v Bratislave*. – University of Economics in Bratislava. Faculty of Economic Informatics; Department of Applied Informatics. – Thesis supervisor: Ing. Peter Procházka, PhD - Bratislava: FHI, 2024, 48p.

The aim of this bachelor thesis is to design and implement a web application for students, which will allow them to select a specific room in the dormitory for the academic year. The thesis thoroughly analyzes current solutions available on the market, compares their advantages and disadvantages, and proposes a new approach to solving this problem. The thesis is divided into several chapters. The first chapter deals with the current state of the problem, including available solutions on the market and their detailed comparison. The second part describes the goal of the thesis and evaluates the questionnaire, based on which a proposal for a new solution to the problem is created. The thesis provides a detailed description of the steps from submitting an accommodation request to selecting a specific room. The last chapter describes the process of creating the web application itself and its integration with the database. The conclusion summarizes the entire process of problem-solving and evaluates the results of the thesis. The result of the solution is a fully functional web application that allows students to effectively and conveniently choose accommodation and a room for the academic year.

Kľúčové slová: web application, react.js, reservation system

Obsah

Úvod	9
1 Súčasný stav riešenia danej problematiky	10
1.1 Súčasný stav riešenia na Ekonomickej univerzite	10
1.2 Súčasný stav riešenia na Univerzite Komenského	12
1.3 Súčasný stav riešenia na Slovenskej technickej univerzite	14
1.4 Porovnanie vybraných ubytovacích systémov	16
2 Cieľ práce, metodika práce a metódy skúmania	18
2.1 Dotazník	18
2.2 Návrh nového riešenia	21
3 Výsledky práce a diskusia	24
3.1 Príprava vývojového prostredia	24
3.2 Databáza	24
3.3 Klientska časť	28
3.3.1 Stránka pre prihlasovanie	29
3.3.2 Domovská stránka	30
3.3.3 Stránka s informáciami o internátoch	33
3.3.4 Stránka pre výber izby	35
3.3.5 Stránka pre správu a sledovanie žiadostí	36
3.3.6 Stránka pre import csv súboru	37
3.4 Serverová časť	38
3.4.1 Koncové body	39
3.5 Zálohovanie	42
3.6 Výzvy	42
3.7 Ďalšie možnosti rozšírenia	43
Záver	45
Zoznam použitej literatúry	46

Zoznam ilustrácií

Obrázok 1: Grafické zobrazenie navštevovaných univerzít respondentov z dotazníka (zdroj: vlastné spracovanie).....	19
Obrázok 2: Grafické zobrazenie miest, kde bývajú študenti počas svojho štúdia (zdroj: vlastné spracovanie).....	19
Obrázok 3: Grafické znázornenie konkrétnych ubytovaní študentov počas ich štúdia (zdroj: vlastné spracovanie).....	20
Obrázok 4: Otázka o výhodách ubytovacieho systému (zdroj: vlastné spracovanie).....	20
Obrázok 5: Otázka o nevýhodách ubytovacieho systému (zdroj: vlastné spracovanie).....	21
Obrázok 6: Otázka o možnostiach vylepšenia ubytovacieho procesu (zdroj: vlastné spracovanie).....	21
Obrázok 7: Príklad príkazu pre vytvorenie tabuľky v SQL (zdroj: vlastné spracovanie) ...	25
Obrázok 8: Príklad príkazu pre vloženie do tabuľky v SQL (zdroj: vlastné spracovanie)..	25
Obrázok 9: ER diagram (zdroj: vlastné spracovanie v dbdiagram.io).....	26
Obrázok 10: Vytvorenie vlastného dátového typu v PostgreSQL (zdroj: vlastné spracovanie)	28
Obrázok 11: Stránka pre prihlasovanie (zdroj: vlastné spracovanie)	30
Obrázok 12: Domovská stránka pre admina (zdroj: vlastné spracovanie)	31
Obrázok 13: Domovská stránka pre študenta (zdroj: vlastné spracovanie).....	31
Obrázok 14: Používateľská lišta pre admina (zdroj: vlastné spracovanie).....	32
Obrázok 15: Používateľská lišta pre študenta (zdroj: vlastné spracovanie)	32
Obrázok 16: Ukážka menu pre admina (zdroj: vlastné spracovanie)	32
Obrázok 17: Ukážka zobrazenia internátu na domovskej stránke (zdroj: vlastné spracovanie)	33
Obrázok 18: Ukážka zobrazenia informácií o internáte (zdroj: vlastné spracovanie).....	34
Obrázok 19: Ukážka zobrazenia fotiek internátu (zdroj: vlastné spracovanie)	34
Obrázok 20: Ukážka zobrazenia polohy internátu na mape (zdroj: vlastné spracovanie)...	34
Obrázok 21: Ukážka podstránky pre výber izby (zdroj: vlastné spracovanie).....	36
Obrázok 22: Stránka pre sledovanie žiadostí pre študenta (zdroj: vlastné spracovanie).....	37
Obrázok 23: Stránka pre sledovanie žiadostí pre admina (zdroj: vlastné spracovanie)	37
Obrázok 24: Stránka pre import csv súboru (zdroj: vlastné spracovanie).....	38
Obrázok 25: Ukážka menu pre typ csv súboru (zdroj: vlastné spracovanie).....	38

Úvod

V súčasnej dobe, v ére neustáleho technologického pokroku a digitalizácie, je nevyhnutné, aby aj univerzity využívali moderné nástroje a technológie na zlepšenie svojich procesov. V tejto bakalárskej práci sa budeme zameriavať práve na ubytovací proces, ktorý predstavuje kľúčovú súčasť študentského života na univerzitách. Konkrétne sa budeme venovať popisu a implementácii webovej aplikácie, a to ubytovacieho rezervačného systému Ekonomickej univerzity v Bratislave.

Súčasný postup prideľovania ubytovania pre študentov, vykonávaný manuálne určitou skupinou ľudí podľa príslušnej fakulty, je v 21. storočí považovaný za neefektívny a zastaralý. S cieľom riešiť tieto problémy a prispieť k modernizácii procesov rezervácií ubytovania na univerzite vytvoríme nový systém. Tento systém nie len eliminuje súčasné nedostatky iných, už existujúcich systémov, ale pridá hodnotu v podobe inovácií, ktoré zlepšujú používateľský zážitok a efektívnosť celého procesu. Cieľom je automatizovať a zefektívniť procesy rezervácií ubytovania, a taktiež poskytnúť používateľom nové možnosti a funkcionality, ktoré sú v súčasnosti nedostupné.

V prvej kapitole podrobne zanalyzujeme súčasné ubytovacie systémy a porovnáme ich medzi sebou. Tento komplexný pohľad nám umožní identifikovať ich silné a slabé stránky a poskytne nám cenné východisko pre návrh a vytvorenie vlastného inovatívneho systému pre Ekonomickú univerzitu v Bratislave.

V druhej kapitole zhrnieme a podrobne opíšeme ciele našej práce. Okrem toho vytvoríme a následne zanalyzujeme dotazník pre študentov bývajúcich na internáte, aby sme lepšie porozumeli ich potrebám a požiadavkám týkajúcich sa ubytovacieho rezervačného systému. Na základe tohto dotazníka vytvoríme návrh nového riešenia, kde zohľadníme relevantné požiadavky respondentov. Týmto spôsobom sa ubezpečíme, že navrhovaný systém bude čo najviac prispôbený potrebám potenciálnych používateľov.

Následne v tretej kapitole pristúpime k samotnej implementácii ubytovacieho systému podľa požiadaviek študentov. Cieľom tejto fázy bude nielen overiť funkčnosť a efektívnosť nového systému, ale aj zhodnotiť úspech implementovaných riešení a identifikovať prípadné oblasti na ďalšie vylepšenia. Počas implementácie budeme klásť dôraz na sledovanie spätnej väzby od budúcich používateľov a reagovať tak na ich potreby a pripomienky s cieľom neustále zdokonaľovať a optimalizovať ubytovací systém.

1 Súčasný stav riešenia danej problematiky

V tejto kapitole sme sa zamerali na aktuálny stav v oblasti ubytovacích systémov pre študentov na troch najväčších univerzitách v Bratislave. Ubytovanie pre študentov predstavuje jednu z kľúčových oblastí, ktorá ovplyvňuje ich študentský život a komfort počas štúdia na svojej univerzite. Cieľom tejto kapitoly bolo poskytnúť podrobný prehľad súčasnej situácie v oblasti ubytovacích systémov na univerzitách a poukázať na pozitívne stránky, ale aj hlavné problémy a nedostatky, s ktorými sa tieto systémy momentálne stretávajú. Detailná analýza súčasného stavu nám umožnila lepšie porozumieť fungovaniu týchto systémov a navrhnúť tak potenciálne zlepšenia, ktoré by mohli prispieť k efektívnejšiemu procesu pridelovania ubytovania pre študentov.

1.1 Súčasný stav riešenia na Ekonomickej univerzite

Ako bolo spomenuté v úvode práce, aktuálne riešenie pridelovania ubytovania na Ekonomickej univerzite spočíva v manuálnom procese, kde hrá silnú rolu ľudský faktor, ktorý so sebou prináša množstvo chýb a nedorozumení.

Postup tohto procesu začína v bode, keď sa podávajú žiadosti o ubytovanie. Študenti EU v Bratislave si môžu požiadať o ubytovanie v Študentských domovoch Ekonomickej univerzity v Bratislave prostredníctvom elektronickej žiadosti v určených termínoch cez Akademický informačný systém (AiS). V AiSe sa im následne rovno vypočítajú body na základe priemeru, vzdialenosti a bonusových aktivít, čím sa odbreňuje ubytovacia komisia, ktorá to nemusí manuálne dopočítavať. Novoprijatí uchádzači majú rovnakú možnosť podať si žiadosť v rámci stanovených termínov, ktoré sú vopred určené. Žiadosti o ubytovanie sú však zamietnuté v prípade neúplných alebo nepravdivých údajov alebo ak študenti nevyhovujú štandardným podmienkam. Žiadosť o ubytovanie si študenti zvyčajne podávali v máji daného akademického roka a v júni toho istého roka si mohli na stránke univerzity pozrieť vyhodnotenie žiadostí, či im žiadosť bola akceptovaná alebo nie. Tento akademický rok sa však celý proces trochu posunul a študenti si o ubytovanie žiadali už v marci. Pridelovanie ubytovania je riadené ubytovacou komisiou príslušnej fakulty, ktorá zabezpečuje kontrolu podaných žiadostí a prípadné overenie doplňujúcich údajov. Žiadosti musia byť podané v stanovenom termíne a po tomto termíne už nie sú akceptované. V žiadosti si môžu študenti zažiadať o konkrétnu izbu na konkrétnom internáte a v poznámke uviesť preferovaného spolubývajúceho alebo spolubývajúcu ideálne z rovnakej fakulty.

Avšak podľa skúseností väčšiny študentov im zvyčajne nie je vyhovené, keďže každý rok dostanú fakulty iné izby pre svoju skupinu pridelovaných študentov. Je taktiež aj veľmi náročné pri takomto manuálnom procese dohliadnuť na to, aby každý býval s kým si žiadal v žiadosti. Ubytovacia komisia žiadosti vyhodnocuje na základe rozdelenia ubytovacích kapacít. Toto rozdelenie je riadené podľa vopred určených kritérií. Od celkového počtu ubytovacích miest, ktoré sú vyčlenené pre jednotlivé fakulty, sa odpočítavajú miesta rezervované pre rektora, dekana alebo miesta pre študentov so špeciálnymi potrebami. Zostávajúce miesta sú následne určené pre študentov nastupujúcich do prvého ročníka alebo pokračujúcich vo svojom štúdiu na príslušnej fakulte. Tieto miesta sú pridelované podľa poradovníka, ktorý je zostavený na základe bodov študentov. Po obdržaní zoznamu správne podaných žiadostí ubytovacia komisia na základe kapacít internátov určí podľa fakúlt, ktorí študenti dostanú internát, ktorí nespĺňajú podmienky určené v smernici a ktorí majú nedostatok bodov. Bodové hodnotenie, podľa ktorého je zostavený poradovník, sa vypočíta na základe študijných výsledkov, vzdialenosti trvalého bydliska od univerzity a doplnkových aktivít. Celkový počet bodov sa vypočíta ako súčet bodov z týchto troch kategórií. Všetky vzorce, podľa ktorých sa body spočítavajú sú uvedené na stránke univerzity www.euba.sk v Zásadách pridelovania ubytovania v študentských domovoch. Je dôležité spomenúť, že poradie podania žiadosti o ubytovanie nemá žiadny vplyv na pridelenie ubytovania, a teda žiadosť o ubytovanie podaná ako prvá alebo posledná je posudzovaná rovnako. Následne sa pred nasledujúcim akademickým rokom študenti, ktorým bolo pridelené ubytovanie, dozvedia, aký dostali internát a ich konkrétnu izbu, ktorú im pridelila ubytovacia komisia z ich fakulty. [1]

Samotný proces, ktorým ubytovacia komisia prechádza počas pridelovania konkrétnych izieb študentom, ktorí majú schválenú žiadosť, spočíva v excel tabuľke, kde majú rozdelené izby, ktoré budú pridelovať. Tie sú rozdelené podľa fakúlt a následne podľa toho, či je daná bunka dievčenská alebo chlapčenská. Na základe toho ubytovacia komisia prideluje študentov, ktorým schválili žiadosť o ubytovanie, na dané izby. Pri tomto procese sú často znevýhodnení najmä študenti, ktorí ostali po akademickom roku bývať na svojej izbe aj počas letných prázdnin a v nasledujúcom akademickom roku im bola pridelená iná izba alebo v horšom prípade aj iný internát. Vzhľadom na to, že sa posledný týždeň v auguste letné ubytovanie neposkytuje kvôli upratovaniu a ich novo pridelená izba nemusí byť pre nich po letnom ubytovaní ešte pripravená, musia sa študenti na týždeň vysťahovať a v prvý septembrový pracovný deň sa môžu nastťahovať na svoju novú izbu v rovnakom alebo úplne

inom internáte. Výnimkou, kedy sa študenti nemusia vyst'ahovať, je len vtedy, ak dostali rovnakú izbu, čo je pri aktuálnom procese veľmi ojedinelý prípad, alebo ich nová izba je už uprataná a pripravená.

Poslednou, ale nesmierne dôležitou nevýhodou vecou je, že celý tento proces je manuálny. V dnešnom svete plnom aplikácií a automatizácie procesov je to veľmi zastaralý a nemoderný postup. Možnosť bývať na študentom zvolenej izbe a prípadne aj so spolubývajúcim, ktorého si sám vybral, môže dopomôcť k lepšiemu výkonu študenta počas akademického roka. Väčšina študentov bývajúcich na internáte tam trávi väčšinu svojho štúdia a prostredie, v akom študent trávi čas, má veľký vplyv aj na výsledky jeho štúdia. Automatizácia tohto procesu by nielen eliminovala mnohé chyby a nedostatky spojené s manuálnym pridelením ubytovania, ale vytvorila by aj efektívnejší a spravodlivejší spôsob pridelenia miest. Vďaka automatizácii by sa tiež zvýšila transparentnosť celého procesu a minimalizovala by sa potreba ľudskej faktoru, čo by ubytovacej komisii ušetrilo čas a úsilie. Okrem toho, modernizácia procesu pridelenia ubytovania by prispela k zlepšeniu celkového študentského prostredia a zvýšeniu ich spokojnosti. Študenti by mali možnosť presnejšie vyjadriť svoje preferencie týkajúce sa ubytovania, čo by zlepšilo ich celkový zážitok zo života na internáte. Zavedenie online systému by umožnilo študentom sledovať stav ich žiadosti a transparentne získať informácie o pridelenom ubytovaní, čím by sa zvýšila aj ich kontrola nad procesom.

Navyše, automatizácia procesu by umožnila ubytovacej komisii venovať viac času na riešenie zložitejších otázok a individuálnu podporu študentov, namiesto toho, aby sa zaoberali manuálnym spracovaním žiadostí o ubytovanie. To by v konečnom dôsledku prispelo k zlepšeniu kvality poskytovaných služieb a udržaniu vysokej úrovne študentského života na Ekonomickej univerzite.

1.2 Súčasný stav riešenia na Univerzite Komenského

Na Univerzite Komenského funguje ubytovací systém prostredníctvom moderného informačného systému, ktorý umožňuje riadenie ubytovania v rámci univerzitných zariadení.

Riaditelia týchto zariadení pravidelne aktualizujú informácie o dostupných izbách a bunkách. Komisia následne rozdeľuje izby podľa rôznych kritérií, ako sú špeciálne potreby študentov, požiadavky zamestnancov a potreby zahraničných študentov. Rozdelenie reálnej

ubytovacej kapacity sa zverejňuje vopred, aby o nej boli študenti informovaní. V prípade výnimočných situácií sa môže rozdelenie upraviť. Ceny za ubytovanie sú tiež pravidelne aktualizované. Pred samotným ubytovaním sa vykonáva kapacitný test elektronického systému, aby bol zabezpečený plynulý priebeh procesu ubytovania. Študenti si musia žiadosti podávať cez elektronický systém, pričom pri prihlásení používajú svoje univerzitné prihlasovacie údaje. To je podporované centrom podpory informačných technológií (CePIT). Študent musí potvrdiť, že sa oboznámil s podmienkami pridelenia ubytovania. Môže podať iba jednu žiadosť a sledovať jej stav v elektronickom systéme. Študenti, ktorí majú prerušené štúdium v aktuálnom akademickom roku, nemôžu požiadať o ubytovanie. Ak ubytovacie zariadenie v minulosti ukončilo zmluvu o ubytovaní so študentom, taktiež nemá nárok na pridelenie ubytovania. Všetci študenti sú zodpovední za správnosť poskytnutých údajov v žiadosti a sú povinní informovať univerzitu o akejkolvek zmene, ktoré by mohli ovplyvniť pridelenie ubytovania. Informácie o hodnotení žiadosti sú dostupné vo vopred určenom termíne a študenti si môžu podať námietky v prípade zistených chýb. V rámci vyhodnocovania žiadostí podaných študentmi platia určité postupy. Fakultný referent posudzuje námietky v žiadostiach prostredníctvom elektronického systému a opravuje chybné údaje v určených termínoch. Žiadosti pokračujúcich študentov sa hodnotia na základe smernice dekana fakulty, pričom sa berie do úvahy súčet bodov, ktorý sa vypočíta na základe rôznych kategórií. Po vyhodnotení žiadostí sa vytvárajú fakultné poradovníky, ktoré sa schvaľujú a zverejňujú v elektronickom systéme. Ubytovanie sa najprv prideli študentom so špeciálnymi potrebami, novoprijatým uchádzačom a zahraničným študentom. V prípade voľnej ubytovacej kapacity sa ubytovanie prideli ďalším študentom. Nárok na pridelené ubytovanie môže zaniknúť v rôznych situáciách, vrátane zrušenia žiadosti zo strany študenta alebo porušenia pravidiel. Po vyhodnotení žiadostí nasleduje kľúčová fáza, ktorá určuje, kto získava ubytovanie a akú izbu dostane. Tento proces je rozdelený do štyroch dní, kedy si študenti postupne vyberajú svoje izby. V prvý deň majú túto možnosť študenti s najvyšším bodovým stavom. Postupne si každý deň v rovnakom čase vyberajú izbu ďalší študenti podľa ich bodového stavu, a to až do štvrtého dňa. Každý deň trvá proces od 8:00 do 20:00, kedy sa študenti prihlasujú do e-ubytovania cez svoje študentské kontá. Po prihlásení majú možnosť vybrať si typ izby - jednolôžkovú, dvojľôžkovú alebo trojlôžkovú. V minulosti mali študenti možnosť vidieť, s kým budú na izbe, no nové regulácie túto možnosť zrušili. Nevýhodou tohto systému je aj fakt, že po vybraní izby nie je možné rezerváciu zmeniť. Oznámenie o pridelení alebo nepridelení ubytovania je zverejnené v elektronickom systéme a doručené všetkým žiadateľom e-mailom na ich elektronické

adresy. Obsah oznámenia o pridelení ubytovania zahŕňa informácie o pridelení izby, cene za danú izbu, charakteristike izby, a výzvu na úhradu kaucie, platobné údaje pre bankový prevod, termín platby a následky pri nesplnení povinností. Proti oznámeniu o nepridelení ubytovania je možné podať odvolanie, musí však byť odôvodnené. [2]

Proces má svoje výhody. Najmä to, že je plne automatizovaný prostredníctvom moderných webových aplikácií, ktoré študentom umožňujú samostatne vyberať izby na nadchádzajúci akademický rok. Ďalšou výhodou je možnosť stabilizácie vybranej izby, čo znamená, že študent má istotu, že si svoju už vybranú izbu udrží a v nasledujúcich rokoch jeho štúdia má na dané miesto prioritu výberu.

1.3 Súčasný stav riešenia na Slovenskej technickej univerzite

Na Slovenskej technickej univerzite platia určité kritériá pre pridelenie ubytovania študentom denného štúdia v študentských domovoch v Bratislave. Tieto kritériá slúžia na zostavenie poradovníka študentov, ktorí si požiadali o ubytovanie v prípade, že miesta nie sú dostatočné.

Oprávnenými žiadateľmi o ubytovanie sú študenti denného štúdia, ktorí nespĺňajú podmienku trvalého bydliska v Bratislave a podávajú žiadosť v stanovenom termíne s požadovanými dokladmi, ktoré je potrebné k žiadosti priložiť. Bodové hodnotenie študentov sa vykonáva na základe ich študijných výsledkov a ďalších kritérií, pričom obdobie hodnotenia sa posudzuje od 1. júna predchádzajúceho akademického roku do 31. mája aktuálneho akademického roka. Žiadatelia majú právo podať námietku proti svojmu bodovému hodnoteniu, ktorú následne skontroluje príslušná študijná referentka. Spracovanie žiadostí a zostavenie poradovníka žiadateľov je zodpovednosťou Ubytovacej komisie príslušnej fakulty alebo ústavu na STU. Pri pridelení ubytovania sa dáva prednosť zahraničným študentom a doktorandom, a následne študentom z poradovníka. Všetky relevantné informácie o ubytovaní a kontakty na príslušné osoby sú zverejnené v informačnom systéme a na webových stránkach príslušných fakúlt a ústavov STU. Pri pridelení ubytovania na Slovenskej technickej univerzite sa berú do úvahy aj kritériá týkajúce sa dodržiavania ubytovacieho poriadku. Študenti, ktorí boli vylúčení zo študentských domovov kvôli porušeniu tohto poriadku, budú pri pridelení ubytovania na nasledujúci akademický rok penalizovaní znížením ich bodov. Za hrubé porušenie ubytovacieho poriadku sa bodové hodnotenie znižuje, a to v závislosti od závažnosti porušenia tohto poriadku. Pri zistení, že študent má pridelené ubytovanie aj inde, bude jeho

ubytovanie zrušené a v budúcnosti mu nebude poskytnutá ďalšia možnosť žiadať o ubytovanie na univerzite. Študenti, ktorým je pridelené ubytovanie, sa musia ubytovať v stanovenom termíne, inak môžu byť vylúčení z poradovníka žiadateľov o ubytovanie a pri ďalšom pridelení ubytovania môže byť ich bodové hodnotenie znížené. Študenti, ktorí si žiadajú o ubytovanie zo zdravotných dôvodov, musia predložiť lekársku správu vo vopred stanovenom termíne. Na základe doložených lekárskejších správ sú študenti zaradení do skupín a body sú im pridelené podľa závažnosti ich zdravotného stavu. Pre študentov nastupujúcich do prvého ročníka štúdia sa vyčleňuje špecifická kapacita miest pre ubytovanie. Celkové bodové hodnotenie pri pridelení ubytovania je ovplyvnené aj výsledkami za štúdium a dodržiavanie pravidiel ubytovacieho poriadku. Toto bodové hodnotenie sa vypočíta na základe váženého študijného priemeru a je ovplyvnené výsledkami študenta v jednotlivých predmetoch. Okrem študijných výsledkov sa však berú do úvahy aj ďalšie faktory, ako je časová vzdialenosť trvalého bydliska študenta od univerzity, športová reprezentácia fakulty alebo univerzity, účasť v rôznych aktivitách schválených univerzitou, darovanie krvi či sociálne kritériá. Maximálny počet bodov, ktoré môže študent získať na základe týchto kritérií, je 100 bodov. Na Slovenskej technickej univerzite to prebieha podobne ako na Univerzite Komenského, a to cez webovú aplikáciu, kde majú prihlasovacie údaje rovnaké ako do AIS. Budúci prváci majú internát garantovaný a taktiež majú možnosť zvoliť si izbu. Pri samotnom výbere izby študent vidí, koľko ľudí si izbu zvolilo a ktorý je v poradí. Je možné zvoliť si viacero izieb a na základe vopred stanovených podmienok sa študentovi vyberie jedna izba z jeho zvolených. Prvým kolo pri výbere ubytovania je možnosť stabilizácie izby a prenechanie izby inému študentovi. Toto kolo sa týka len študentov, ktorí už na univerzite študujú a bývajú v danom roku na internáte. V nasledujúcich kolách si študenti postupne vyberajú izby na základe počtu bodov, a teda znova majú výhodu tí, ktorí majú väčší počet bodov. Nevýhodou určite je, že študenti môžu čakať na danú izbu niekoľko v poradí, čo robí proces menej zautomatizovaný, pretože na konci procesu, kedy si vyberajú študenti, niekto musí izby pridelit' na základe poradia študentov na daných izbách. To robí celý proces omnoho komplikovanejší. Výhodou je stabilizácia izby, čo ocenia najmä študenti, ktorí ostávajú na internáte aj počas letných prázdnin. Študenti majú právo podať odvolanie proti neprideleniu ubytovania zo závažných zdravotných alebo sociálnych dôvodov. Ubytovacia komisia fakulty a ústavu sa zaoberá odvolaniami iba v prípade, ak sú podložené relevantnými dokladmi v súlade s platnými kritériami pre pridelenie ubytovania. [3]

Ako už bolo spomenuté, výhodu, ktorú tento systém zdieľa s Univerzitou Komenského, avšak nie s Ekonomickou univerzitou, je automatizovaný systém, ktorý prebieha cez webovú aplikáciu. Avšak aj napriek webovej aplikácii aj na konci tohto procesu je silno dôležitý ľudský faktor, čo môže viesť k väčšej chybovosti.

1.4 Porovnanie vybraných ubytovacích systémov

Každá univerzita má svoj vlastný spôsob pridelovania ubytovania a riadenia ubytovacích kapacít, a preto je dôležité analyzovať a porovnať tieto systémy.

Ekonomická univerzita v Bratislave používa silno manuálny proces pridelovania ubytovania, ktorý je riadený ubytovacou komisiou príslušnej fakulty. Študenti majú možnosť podať žiadosť o konkrétnu izbu na konkrétnom internáte, pričom pridelovanie ubytovania sa riadi bodovým hodnotením a poradím podľa tohto hodnotenia. Výhodou tohto systému je, že ubytovacia komisia sa snaží prihliadať na bodové hodnotenie študentov a následne podľa toho prideliť žiadanú izbu. Tento prístup môže motivovať študentov počas štúdia získavať viac bodov, čo im zvyšuje šancu na získanie ubytovania v preferovanom internáte. Tento systém teda podporuje študentov v zlepšovaní svojich študijných výsledkov a aktívnej účasti na univerzitných aktivitách, čo v konečnom dôsledku prispieva k celkovému rozvoju a úspechu študentov. Nevýhody tohto systému sú spojené predovšetkým s jeho manuálnym charakterom. Proces pridelovania je náročnejší a menej efektívny v porovnaní s automatizovanými systémami. Okrem toho môže byť ťažké zabezpečiť spravodlivé a transparentné pridelovanie ubytovania pri manuálnom procese, čo môže viesť k obvineniam z preferencie niektorých študentov pred ostatnými.

Na druhej strane Univerzita Komenského má moderný ubytovací systém, ktorý umožňuje elektronické podávanie žiadostí a online sledovanie stavu žiadosti o ubytovanie. Tento systém prináša výhodu rýchleho a pohodlnejšieho spôsobu podania žiadosti o ubytovanie, čo študentom uľahčuje celý proces. Navyše, možnosť online sledovania stavu žiadosti poskytuje študentom transparentnosť a informovanosť o priebehu ich žiadosti, čo znižuje ich stres. Študenti si tiež môžu vyberať izby podľa svojich preferencií, čo prispieva k ich spokojnosti aj počas štúdia. Ďalšou výhodou je istota, že si vybranú izbu udržia kľudne aj počas celého štúdia vďaka možnosti stabilizácie si izby, čo zvyšuje pocit istoty študentov. Napriek týmto výhodám však existujú aj nevýhody tohto systému. Jednou z nich môže byť nadmerná konkurencia o konkrétne izby alebo internáty, čo môže viesť k frustrácii a sklamaniam u študentov, ktorí nedostanú svoju plánovanú voľbu. Okrem toho, elektronický

system môže byť náchylný na technické problémy, čo môže spôsobiť oneskorenie alebo chyby v procese podávania žiadostí. Taktiež je študentom umožnené filtrovanie len na základe počtu lôžok na izbe, čím sa im znemožňuje výber izby presne podľa svojich kritérií.

Slovenská technická univerzita uplatňuje bodové hodnotenie študentov na základe rôznych kritérií, vrátane študijných výsledkov či účasti na rôznych aktivitách podporovaných univerzitou. Tento prístup k pridelovaniu ubytovania umožňuje univerzite brať do úvahy nielen akademické úspechy študentov, ale aj ich aktívnu účasť vo verejnom živote a spoločenské angažovanie. Študenti majú tak možnosť aktívne ovplyvniť svoje bodové hodnotenie aj mimo akademického kontextu. Medzi výhody tohto systému patrí fakt, že študenti sú motivovaní angažovať sa nielen vo svojom štúdiu, ale aj vo verejnom živote a rôznych spoločenských aktivitách, čo prispieva k ich rozvoju. Okrem toho, možnosť výberu izby z dostupných možností dáva študentom istú mieru kontroly nad ich ubytovaním a umožňuje im vybrať si takú izbu, ktorá najviac vyhovuje ich potrebám a preferenciám. Napriek týmto výhodám však existujú aj nevýhody. Tou je, že študenti majú neistotu ohľadom konečného výberu izby. Keďže študenti si môžu vybrať viacero izieb a finálne rozhodnutie je na ubytovacej komisii, študenti nemajú istotu, ktorú izbu im komisia v skutočnosti vyberie. Táto neistota môže viesť k znepokojeniu u študentov, ktorí majú svoje preferencie ohľadom ubytovania a chcú mať istotu, že budú bývať na izbe, ktorá im najviac vyhovuje.

Napriek vyššie zhrnutým výhodám a nevýhodám týchto systémov na jednotlivých univerzitách, každý systém prináša svoje vlastné pozitíva a prispieva k celkovému cieľu - poskytnúť ubytovanie pre študentov, ktoré podporuje ich akademický rast a spokojnosť počas štúdia.

2 Cieľ práce, metodika práce a metódy skúmania

Cieľom našej práce bolo vyvinúť a implementovať moderný a efektívny ubytovací systém pre Ekonomickú univerzitu v Bratislave, ktorý zodpovedal súčasným požiadavkám a štandardom v oblasti informačných technológií. Konkrétne sme sa snažili dosiahnuť tieto ciele:

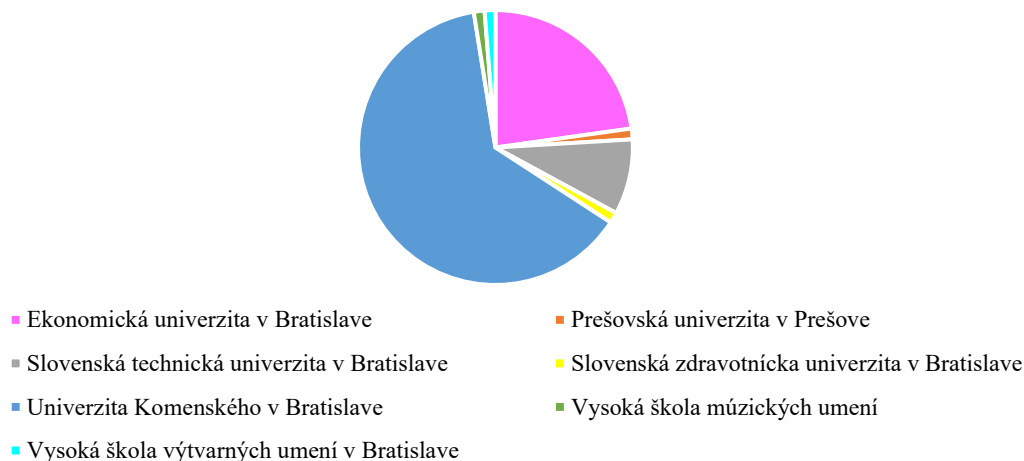
1. Automatizácia procesu ubytovania študentov: Navrhnuť a implementovať systém, ktorý umožní študentom zarezervovať si internátnu izbu podľa svojich požiadaviek.
2. Zlepšenie používateľského zážitku: Poskytnúť intuitívne rozhranie, ktoré umožní študentom jednoducho prehľadávať dostupné internáty, filtrovať si izby podľa svojich požiadaviek a následne si vybrať požadovanú izbu.
3. Optimalizácia procesov správy ubytovania: Vytvoriť nástroje pre administrátorov, ktoré im umožnia efektívne spravovať a monitorovať všetky rezervácie, priradenie jednotlivých izieb študentom či importovať potrebné údaje do databázy priamo v aplikácií, čím sa zníži manuálnosť procesu a zlepši sa celková efektivita tohto ubytovacieho procesu.
4. Analyzovanie a hodnotenie výsledkov: Poskytnúť dôkladnú analýzu implementácie nového ubytovacieho systému na základe reálnej používateľskej spätnej väzby, čo umožní identifikovať silné stránky, ale aj nedostatky a oblasti na ďalšie vylepšenia.

2.1 Dotazník

V rámci metodiky práce a metód skúmania sme sa venovali analýze výsledkov z dotazníka, ktorý sme realizovali medzi vysokoškolskými študentmi. Cieľom dotazníka bolo získať rôzne pohľady, názory a návrhy od študentov z rôznych univerzít na problematiku ubytovacích systémov ich univerzity a ich preferencie a nápady na vylepšenia ohľadom webovej aplikácie na pridelovanie ubytovania. Celkovo sme získali spätnú väzbu od 80 študentov, čo nám umožnilo lepšie pochopiť súčasné potreby a očakávania študentov v oblasti pridelovania ubytovania na univerzitách a taktiež sme získali aj vedomosť o aktuálnych nedostatkoch, ktorým sme sa pri implementovaní chceli vyvarovať. Analyzovať tieto výsledky nám pomohlo identifikovať funkcionality a oblasti, na ktoré sme

sa zamerali pri návrhu a vývoji webovej aplikácie pre efektívnejšie a používateľsky prívetivejšie riadenie ubytovacích kapacít pre študentov.

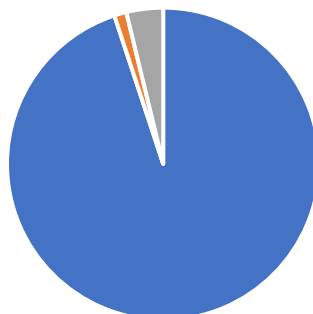
Akú školu študuješ?



Obrázok 1: Grafické zobrazenie navštevovaných univerzít respondentov z dotazníka (zdroj: vlastné spracovanie)

Výsledky prvej otázky z nášho dotazníka vypovedajú, že respondenti sú z rôznych univerzít na Slovensku. Zistili sme, že najväčšiu časť respondentov tvorili študenti Univerzity Komenského, ktorí predstavovali vyše 61%. Necelých 24% študentov študuje na Ekonomickej univerzite. Tretiu najpočetnejšiu skupinu tvorili študenti zo Slovenskej technickej univerzity. Okrem toho sme zaznamenali prítomnosť menších skupín študentov z rôznych ďalších univerzít na Slovensku. Pestrosť respondentov nám umožnila získať široké spektrum pohľadov a názorov študentov.

Kde počas štúdia bývaš?

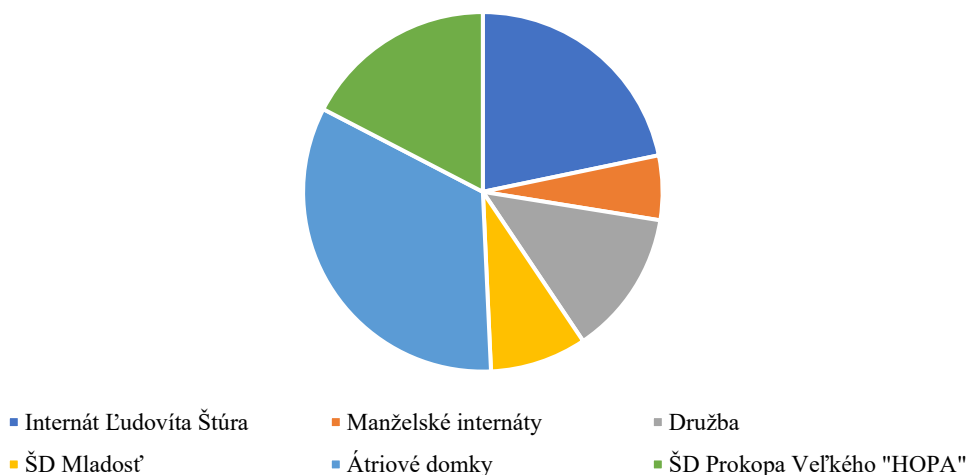


Obrázok 2: Grafické zobrazenie miest, kde bývajú študenti počas svojho štúdia (zdroj: vlastné spracovanie)

94% našich respondentov, čo predstavuje počet 75 z celkových 80, uviedlo, že aktuálne býva na internáte. Táto vzorka študentov, ktorí bývajú na internáte, bola následne oslovená v otázkach týkajúcich sa výhod, nevýhod a navrhovaných zmien týkajúcich sa ich

ubytovania. Otázka týkajúca sa konkrétneho internátu, na ktorom študenti bývajú, bola otvorená, čo umožnilo respondentom uviesť ich vlastné skúsenosti a postrehy. Po spracovaní odpovedí sme identifikovali najčastejšie uvedené internáty, kde respondenti bývajú a výsledky sme zhrnuli do grafu.

Na akom internáte bývaš?



Obrázok 3: Grafické znázornenie konkrétnych ubytovaní študentov počas ich štúdia (zdroj: vlastné spracovanie)

Hlavným dôvodom, prečo sme sa rozhodli vytvoriť dotazník, bolo získanie informácií o výhodách, nevýhodách a návrhoch na zlepšenie rôznych ubytovacích systémov. Z tohto dôvodu boli v nasledujúcich otázkach možnosti voľnej odpovede, aby sa študenti vedeli lepšie vyjadriť.

...

STRUČNE ZHRŇ VÝHODY UBYTOVACIEHO SYSTÉMU NA TVOJEJ UNIVERZITE. *

Text dlhej odpovede

Obrázok 4: Otázka o výhodách ubytovacieho systému (zdroj: vlastné spracovanie)

Medzi výhody, ktoré študenti najčastejšie uvádzali, patrila možnosť výberu konkrétnej izby a stabilizácia danej izby na nasledujúci akademický rok. Študenti tiež ocenili spravodlivé rozdelenie bodov na základe študijných výsledkov a vzdialenosti od univerzitného mesta. S tým súvisí aj často spomínaná výhoda, a to, že výber izby postupuje hierarchicky od najvyššieho počtu bodov po najnižší. Respondenti uviedli ako pozitívum aj to, že prváci majú internát garantovaný.

STRUČNE ZHRŇ NEVÝHODY UBYTOVACIEHO SYSTÉMU NA TVOJEJ UNIVERZITE. *

Text dlhej odpovede

Obrázok 5: Otázka o nevýhodách ubytovacieho systému (zdroj: vlastné spracovanie)

Ďalej sme sa pýtali aj na nevýhody. Najčastejšou nevýhodou, ktorú respondenti uviedli, je nestabilný systém v dôsledku veľkého náporu študentov. Často spomínaným problémom však bolo, že súčasný systém je málo prehľadný alebo príliš manuálny. Študentom chýba aj grafická orientácia pre lepší výber izby. Študenti sa sťažovali aj na zastaraný systém či nedostatočnú transparentnosť celého procesu.

ČO TI CHÝBA NA UBYTOVACOM SYSTÉME TVOJEJ UNIVERZITY? *

Text dlhej odpovede

Obrázok 6: Otázka o možnostiach vylepšenia ubytovacieho procesu (zdroj: vlastné spracovanie)

Posledná otázka v našom dotazníku sa týkala návrhov na zlepšenie súčasného systému. Študenti z univerzít, kde nie je možná stabilizácia izby, by uvítali túto možnosť. Často uvádzanými návrhmi boli zlepšenia grafického rozhrania systému a pridanie informácií o každej izbe, vrátane poschodia, orientácie a umiestnenia v chodbe. Študenti by tiež uvítali pridanie fotografií izieb, aby mohli vidieť, či sú zrekonštruované. Ďalším častým návrhom bolo zobrazenie obsadenosti izby a počtu zaregistrovaných študentov.

2.2 Návrh nového riešenia

V rámci návrhu nového ubytovacieho systému sme sa sústredili na vytvorenie webovej aplikácie, ktorá výrazne zlepší proces rezervácie ubytovania pre študentov. Na základe detailnej analýzy potrieb a návrhov z dotazníka, ktorý zahŕňal 80 študentov z rôznych univerzít, sme identifikovali kľúčové funkcionality pre efektívnejšie a modernejšie riešenie.

V prvotnom procese prejavu záujmu o ubytovanie sme sa rozhodli, že bude zachovaný aktuálny proces, a to zakliknutie priamo cez AiS. Zachoval sa tým aj proces prideľovania bodov študentom, pri ktorom sa body udeľujú na základe študijného prospechu, vzdialenosti trvalého bydliska od univerzity či doplnkových aktivít. Tento výpočet vykonáva

AiS automaticky priamo z údajov v ňom. Súčasný proces sme sa rozhodli zachovať v dôsledku toho, že AiS nepodporuje prepojenie s externými aplikáciami. V dôsledku toho sme nevedeli získať údaje potrebné pre výpočet bodov študenta, ako napríklad priemer či miesto trvalého bydliska. Výpočet bodov prebiehal tak ako doteraz podľa smernice, ktorú vopred určil rektor v zásadách pridelenia ubytovania. Tie sú vždy zverejnené na stránke univerzity. Študenti sme ponechali možnosť získavať dodatočné body za preddefinované aktivity, ako sú napríklad účasť v kvapke krvi či dobrovoľnícke upratovacie akcie v okolí internátov. Z AiS si pred každým pridelovaním samotného ubytovania komisia vytvorí csv súbor všetkých relevantných záujemcov o ubytovanie, ktorý bude následne importovaný priamo do webového rozhrania podľa potrieb databázy. Títo študenti budú mať následne prístup do aplikácie a tým pádom im bude umožnené požiadať si o ubytovanie a teda aj izbu vo vopred stanovenom termíne. Prístup im bude umožnený pomocou školského emailu a hesla, ktoré bude ID študenta a je jedinečné.

Preventívnym opatrením zameraným na zníženie rizika častého pádu systému bude implementácia viacerých kôl v procese výberu ubytovania. Nový systém bude obsahovať viac kôl s menším počtom študentov. Tým sa predíde preťaženiu systému, čo prispieva k plynulejšiemu procesu výberu izby a celkovo príjemnejšiemu zážitku pre študentov. Tieto kolá si rozvrhne parlament podľa počtu študentov v danom akademickom roku. Taktiež sa tým docieli to, že prioritne si budú vyberať študenti s najvyšším počtom bodov. Oстане tak zachovaná snaha študentov získať počas štúdia čo najviac bodov. V každom kole sa bude prístup postupne umožňovať študentom čoraz s nižším počtom bodov. V prvom kole si študenti s najvyšším počtom bodov budú vyberať spomedzi všetkých izieb a bude tak väčšia pravdepodobnosť, že si vyberú požadovaný internát či izbu. Oстане tak zachovaná hierarchia študentov, ktorí majú vyššie počty bodov, a tým pádom s veľkou pravdepodobnosťou majú lepší priemer alebo bývajú ďaleko od univerzity. [4]

Na základe požiadaviek študentov sme sa rozhodli, že webová aplikácia bude obsahovať podrobné fotky izieb na internáte podľa blokov, okolia internátu či všetky potrebné informácie, ktoré by mohli ovplyvniť študentov pri výbere izby. Pri danom internáte budú môcť vidieť aj jeho polohu na mape. Študenti budú môcť sledovať svoju žiadosť, či je schválená, nevybavená alebo zamietnutá.

Dôležitou časťou, ktorú sme sa rozhodli implementovať, bol filter, ktorý poskytne študentom rozšírené možnosti pri výbere ubytovania podľa ich konkrétnych preferencií. Bude im umožňovať filtrovať izby podľa názvu internátu, počtu lôžok, budú si môcť vybrať

izbu podľa orientácie voči svetovej strane, ale aj orientáciu na chodbe daného internátu. Okrem toho budú mať možnosť filtrovať izby podľa ich stavu, či už sú zrekonštruované alebo nie. Výber bude zahŕňať aj možnosť vybrať si izbu na konkrétnom poschodí a v určitom bloku, čím získajú väčšiu kontrolu nad umiestnením svojho ubytovania v rámci internátu.[5]

Tieto nové funkcionality boli navrhnuté tak, aby vytvorili intuitívne a priateľské prostredie pre používateľov. Cieľom bolo nie len optimalizovať samotný proces rezervácií, ale tiež poskytnúť študentom rozsiahlejšie a transparentnejšie informácie, čo im pomôže pri kvalitnejšom výbere ubytovania.

3 Výsledky práce a diskusia

V tejto kapitole sme podrobne rozobrali proces vytvárania nášho ubytovacieho systému. Základom tejto tvorby bol návrh z 2. kapitoly a samotná implementácia webovej aplikácie, ktorá študentom umožní intuitívne a efektívne vybrať si ubytovanie na základe ich potrieb a preferencií.

3.1 Príprava vývojového prostredia

Príprava vývojového prostredia pre našu webovú aplikáciu začala inštaláciou najnovšej verzie Visual Studio Code (VS Code). Stiahli sme si inštallačný súbor z oficiálnej stránky VS Code a postupne nasledovali inštallačné kroky, ktoré boli vhodné pre operačný systém Windows 11, ktorý sme využívali. Po úspešnej inštalácii sme prešli k vytvoreniu dvoch nových projektov v rámci VS Code, a to projekt frontend a backend. Intenzívne sme využívali nástroj správy balíkov Node Package Manager (npm). *Npm je prostredie na správu balíkov a závislostí v JavaScriptových projektoch.* [6]

Jeho implementácia v našom vývojovom prostredí výrazne zjednodušila proces správy knižníc a nástrojov potrebných pre našu webovú aplikáciu. Po inštalácii npm sme inicializovali náš projekt pomocou príkazu `npm init`, čo nám umožnilo definovať a spravovať balíky a ich závislosti prostredníctvom súboru `package.json`. Následne sme využívali npm na inštaláciu a aktualizáciu potrebných modulov, ako napríklad `React.js`, `Node.js`, `Express.js` a ďalších technológií, ktoré sme integrovali do našej aplikácie. S týmito krokmi sme úspešne prichystali vývojové prostredie pre našu webovú aplikáciu a inicializovali projekt vo Visual Studio Code, pripravený na ďalšie fázy.

V rámci prípravy nášho vývojového prostredia sme tiež integrovali nástroj s názvom Prettier, aby sme dosiahli prehľadný kód v našom projekte. *Prettier je automatizovaný formátovací nástroj, ktorý umožňuje jednoduchú a efektívnu úpravu formátu zdrojového kódu podľa stanovených štandardov.* [7]

3.2 Databáza

Dôležitým prvkom v našej webovej aplikácii bola aj databáza, kde sme uchovali všetky potrebné údaje, čím sme vytvorili prispôsobivú aplikáciu namiesto statickej.

Rozhodli sme sa pre PostgreSQL kvôli jeho využiteľnosti a modernosti. PostgreSQL je bezplatný a open-source relačný databázový manažment systém (RDBMS), ktorý kladie

dôraz na rozšíriteľnosť a dodržiavanie SQL štandardov. PostgreSQL je podporovaný na všetkých hlavných operačných systémoch, vrátane Linuxu, FreeBSD, OpenBSD, macOS a Windows. Má mnoho funkcií zameraných na pomoc vývojárom pri tvorbe aplikácií, administrátorom pri ochrane integrity dát a vytváraní odolných prostredí a pomoci pri správe dát, bez ohľadu na veľkosť dátovej sady. Medzi ďalšie dôležité funkcie patrí podpora štruktúrovaných a neštruktúrovaných dát, podpora rôznych dátových typov. Okrem toho, PostgreSQL má širokú komunitu, čo znamená, že je neustále vyvíjaný a aktualizovaný. Táto komunita vývojárov prispieva k bezpečnosti a stabilite systému, a zároveň umožňuje prístup k veľkej dokumentácii a rôznym online zdrojom pre riešenie problémov. [8]

Na správu databázy sme si nainštalovali PG Admin, grafické prostredie. PG Admin nám poskytol intuitívny spôsob na prácu s našou databázou. Po úspešnej inštalácii PG Admin a PostgreSQL sme si vytvorili databázu v PG Admin. Na základe vopred pripraveného návrhu sme pomocou príkazu uvedeného nižšie vytvorili potrebné tabuľky a stĺpce s dátovými typmi. [9]

```
CREATE TABLE internat (  
    id_internat INTEGER PRIMARY KEY,  
    nazov VARCHAR(255),  
    adresa VARCHAR(255),  
    suradnice DOUBLE PRECISION,  
    suradnice2 DOUBLE PRECISION,  
    fotky BYTEA,  
    popis TEXT  
);
```

Obrázok 7: Príklad príkazu pre vytvorenie tabuľky v SQL (zdroj: vlastné spracovanie)

Našu databázu sme potom prepájali so serverovou časťou resp. backendom pomocou informácií ako názov hostiteľa, používateľské meno, heslo a port. Po prepojení databázy sme do tabuliek vkladali údaje za pomoci príkazu, ktorý je uvedený na obrázku nižšie.[10]

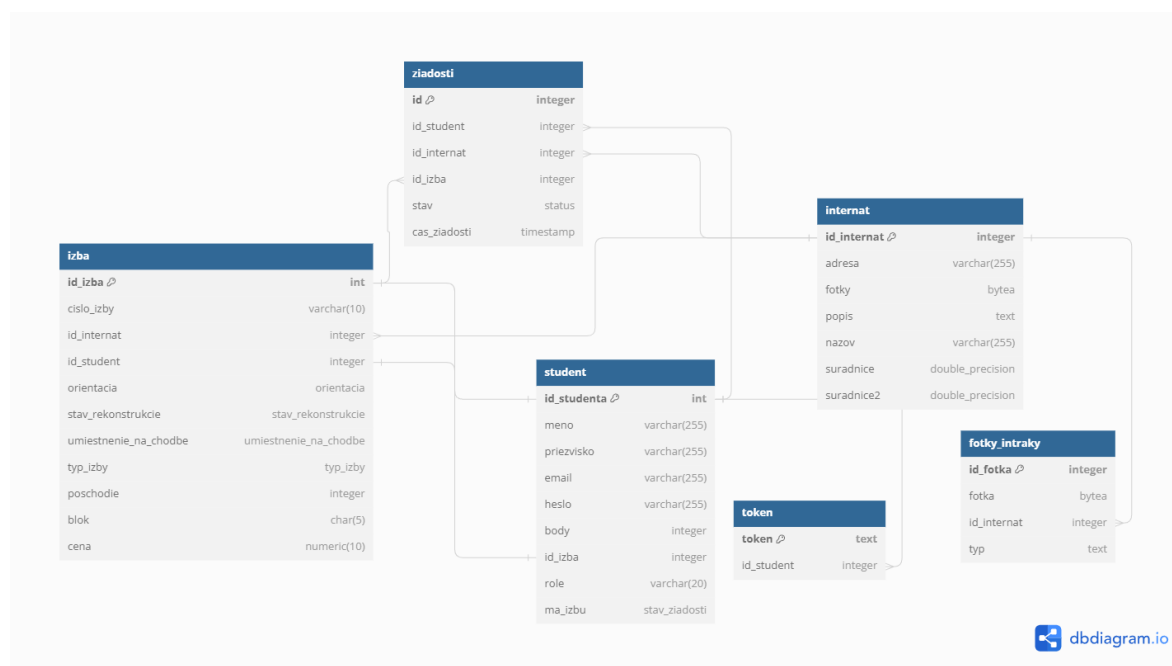
```
INSERT INTO internat (id_internat, nazov, adresa, suradnice, suradnice2, popis)  
VALUES  
('Názov internátu', 'Adresa internátu', 48.123456, 17.987654, , 'Popis internátu');
```

Obrázok 8: Príklad príkazu pre vloženie do tabuľky v SQL (zdroj: vlastné spracovanie)

V procese implementácie sme sa rozhodli vložiť obrázky do databázy pomocou Base64 image encodera. Tento kódovací mechanizmus umožňuje konverziu obrázkových súborov do textového reťazca vo formáte Base64, čo umožňuje ukladanie obrázkov priamo ako

textové údaje v databáze. Po získaní Base64 reprezentácie obrázku sme túto hodnotu vložili do príslušného stĺpca v databáze pomocou SQL príkazu INSERT INTO, ktorý je spomenutý vyššie.

Po celom procese vytvárania a následného dopĺňovania databázy počas implementovania aplikácie sme dospeli k záverečnému entito-relačnému modelu databázy, ktorý je uvedený na obrázku nižšie. Na grafické zobrazenie databázy sme použili stránku dbdiagram.io, vďaka ktorej sme zdarma mohli zobrazit' tabuľky, stĺpce, dátové typy a vzťahy medzi tabuľkami našej databázy. [11]



Obrázok 9: ER diagram (zdroj: vlastné spracovanie v dbdiagram.io)

V tabuľke „student“ sme uchovali informácie o študentoch. Každý študent má v AiS uvedené svoje ID, ktoré sme použili aj my. Stĺpec *id_student* je zároveň primárny kľúč, pretože nemôže existovať študent s rovnakým ID. Pre stĺpce *id_student*, *body* a *id_izba* sme zvolili dátový typ integer, čiže celé číslo. *Id_izba* je cudzí kľúč z tabuľky „izba“, o ktorej si povieme v ďalšom odseku. V stĺpcoch *meno*, *priezvisko*, *email*, *heslo* a *role* sme použili varchar. Okrem role, kde sme tento typ limitovali na 20 bytov resp. znakov, sme všade použili limit 255 bytov. V stĺpci *role* sa bude uchovávať hodnota „student“ alebo „admin“. Stĺpec *ma_izbu* má nami vytvorený typ *stav_ziadosti*. *Stav_ziadosti* je enum, ktorý uchováva hodnoty „nepožiadané“, „požiadané“, „schválené“ a „neschválené“. *Id_student* sa používal aj v menšej tabuľke „token“, v ktorej sa ukladajú JWT tokeny, o ktorých si povieme v ďalšej kapitole.

Ďalšia tabuľka s názvom „izba“ uchováva informácie o daných izbách a obsahovala stĺpce ako primárny kľúč stĺpec *id_izba*, ktorý je spolu s *id_student*, *id_internat* a *poschodie* typ integer. V tejto tabuľke sa nachádza viacero cudzích kľúčov, a to *id_student*, čo sme si spomenuli v predchádzajúcom odseku a taktiež *id_internat* z tabuľky „internat“, o ktorom si povieme v nasledujúcom odseku. V tejto tabuľke sa nachádza aj stĺpec *cislo_izby* typu varchar s obmedzením na 10 znakov. Hoc by sa mohlo zdať, že stĺpec *cislo_izby* by mal byť integer, väčšina izieb obsahuje aj písmená označenia ako A alebo B, preto by integer nebol vhodný. V tejto tabuľke sa nachádza viacero stĺpcov s vlastnými dátovými typmi, ktoré v sebe ukrývajú povolené hodnoty, ktoré sa pre daný stĺpec môžu vložiť. Stĺpec *orientacia* má typ *orientacia* s povolenými hodnotami „sever“, „juh“, „východ“, „západ“. *Stav_rekonstrukcie* je typu *stav_rekonstrukcie* a sú v ňom enumy „Rekonštruovaná“ alebo „Nerekonštruovaná“. *Umiestnenie_na_chodbe* je typu *umiestnenie_na_chodbe* s povolenými hodnotami „Začiatok“, „Koniec“, „Pri výťahu“, „Pri schodoch“, „Pri kuchynke“, „Pri spoločných WC“. V stĺpci *typ_izby*, ktorý má dátový typ *typ_izby* sú povolené hodnoty „Jednolôžková“, „Dvojľôžková“, „Trojlôžková“. Ako ďalší stĺpec máme *blok*, ktorý je dátového typu char s 5 povolenými znakmi. Posledným stĺpcom tejto tabuľky je *cena*, ktorá je typu numeric a môže obsahovať dokopy 10 číslíc vrátane desatinnej čiarky.

Tabuľka „internat“ obsahuje údaje o rôznych internátoch. Každý internát je identifikovaný unikátnym číslom *id_internat*, čo je zároveň primárny kľúč. Ďalej tabuľka obsahuje stĺpec *adresa* a *nazov*, kde oba majú dátový typ varchar s počtom povolených znakov 255. Stĺpec *popis* je dátového typu text a slúži na podrobný popis o internátoch. Stĺpec *fotky* je typu *bytea*. Typ *bytea* uchováva obrázky vo formáte binárnych dát. *Suradnice* a *suradnice2* sú typu *double precision* a uchovávajú geografické súradnice internátov. Táto tabuľka nemá žiadne cudzie kľúče z iných tabuliek.

V tabuľke „fotky_intrakty“ uchovávame fotografie jednotlivých internátov. Každá fotografia má unikátne id, ktoré reprezentuje stĺpec *id_fotka* a tento stĺpec je zároveň primárny kľúč typu integer. Tabuľka obsahuje aj stĺpec *id_internat* z tabuľky „internat“, ktorý je cudzím kľúčom a vďaka tomu priradzujeme fotky ku konkrétnemu internátu a je taktiež typu integer. Stĺpec *fotka* je typu *bytea*, o ktorom sme si už povedali v odseku vyššie. Posledným stĺpcom v tejto tabuľke je stĺpec *typ*, ktorý je typu text a obsahuje informáciu, čoho sa týka daná fotografia. Môže obsahovať názov bloku, miestnosti alebo či je to fotka exteriéru.

Poslednou tabuľkou je tabuľka „ziadosti“, ktorá uchováva všetky žiadosti o izby. Primárnym kľúčom je stĺpec *id*, ktorý je typu integer a je nastavený tak, že sa navyšuje automaticky s pribúdajúcimi žiadosťami o izby. Cudzie kľúče v tejto tabuľke predstavujú stĺpce *id_student* z tabuľky „student“ a predstavuje to študenta, ktorý si podal žiadosť. Stĺpec *id_internat* z tabuľky „internat“ a dáva informáciu o tom, o aký internát si študent žiada a stĺpec *id_izba* z tabuľky *izba* hovorí, o akú izbu si požiadal. Všetky tieto stĺpce sú typu integer. Ďalším stĺpcom v tejto tabuľke je stĺpec *stav* a jeho dátovým typom je status, ktorý sme vytvorili my ako enum a vložili sme do neho povolené hodnoty „nevybavené“, „schválené“ a „zamietnuté“. Posledným stĺpcom je *cas_ziadosti* typu timestamp, kde je uložený čas a dátum poslednej aktualizácie konkrétnej žiadosti.

V tejto časti sme veľa hovorili o tabuľkách, stĺpcoch a dátových typoch, ktoré sa nachádzajú v našej databáze. Spomenuli sme aj vytváranie vlastných dátových typov a povolených hodnôt. Dátové typy sme vytvárali pomocou príkazu uvedeného na obrázku 7:

```
CREATE TYPE status AS ENUM ('nevybavené', 'schválené', 'zamietnuté');
```

Obrázok 10: Vytvorenie vlastného dátového typu v PostgreSQL (zdroj: vlastné spracovanie)

Následne sme tento dátový typ použili pri vytváraní tabuľky a pri vkladaní do tabuľky sme v tomto stĺpci použili jednu z predvolených hodnôt.

3.3 Klientska časť

Klientska časť je neoddeliteľnou súčasťou našej práce, pretože umožňuje interakciu s používateľmi a poskytuje im dôležitú spätnú väzbu o ich správaní. Okrem toho, dizajn klientskej časti hrá rovnako dôležitú úlohu pri formovaní celkového dojmu používateľov. Webové aplikácie využívajú kombináciu rôznych technológií, z ktorých hlavné zahŕňajú HTML, CSS a JavaScript. Tie sme aj použili pri implementácii klientskej časti našej aplikácie. [12]

HTML, čo je skratka pre HyperText Markup Language, nám poslužil na štruktúrovanie obsahu webových stránok a zabezpečil ich správne zobrazenie v prehliadači. [13]

Spolu s CSS, ktoré definuje vzhľad a formátovanie stránky, tvorili základ našej webovej aplikácie. [14]

JavaScript, kľúčový skriptovací jazyk, pridal interaktivitu a dynamiku do našej webovej aplikácie, umožňujúc manipuláciu s obsahom a reakciu na používateľské

interakcie. Javascriptový framework React.js s jeho komponentovým prístupom a JSX syntaxou zjednodušil tvorbu komponentov a poskytol nám flexibilitu pre vývojárov pri vytváraní znovu použiteľných komponentov. Komponent v Reacte je základnou stavebnou jednotkou užívateľského rozhrania, obsahujúcou HTML, CSS a JavaScript kód a umožňujúcou tvorbu modulárneho a opakovateľne použiteľného kódu. [15]

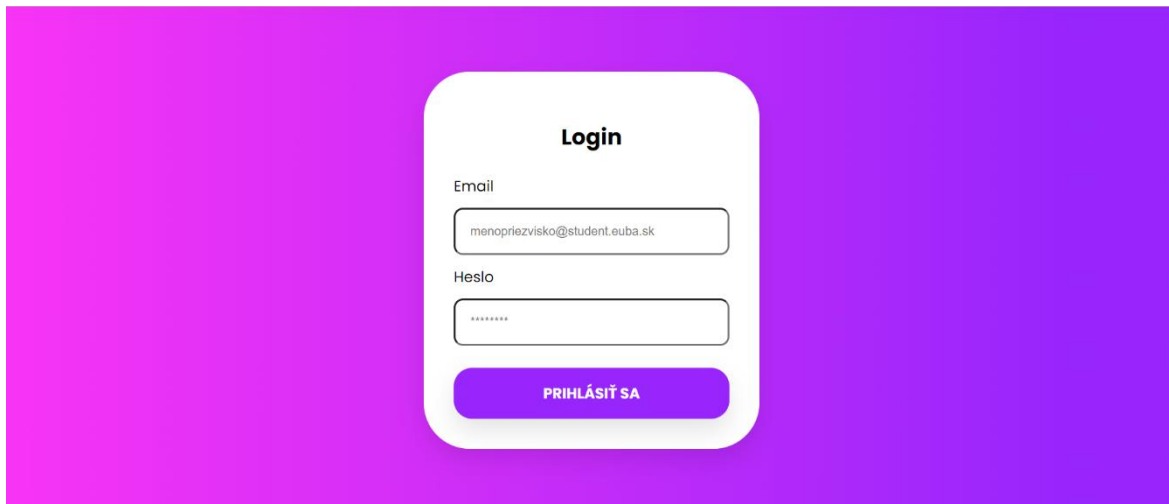
Všetky tieto technológie zohrali významnú časť pri implementácii klientskej časti resp. frontendu. V nasledujúcich kapitolách si popíšeme jednotlivé podstránky a komponenty na nich, ktoré sme používali v našej webovej aplikácii.

3.3.1 *Stránka pre prihlasovanie*

Medzi prvými bola vytvorená podstránka pre prihlasovanie, na ktorej sa nachádza Login komponent, ktorý nám zabezpečil prihlasovanie na základe údajov z databázy, ktoré sú uložené v tabuľke „student“. Tento komponent bol zodpovedný za získanie používateľských údajov, ako je email a heslo, a ich odoslanie na server na overenie a prihlásenie používateľa. Komponent bol písaný v React.js ako zvyšné komponenty a na ukladanie stavov sme použili Redux, čo je knižnica pre správu stavov v aplikácii. Táto knižnica nám umožnila správu stavových údajov pre celú aplikáciu. Pomocou Reduxu sme mohli udržiavať a aktualizovať stavy, ako napríklad či je používateľ prihlásený alebo odhlásený. [16]

Na začiatku sme importovali potrebné funkcie a knižnice, ako napríklad useState pre správu stavu komponentu, useNavigate pre navigáciu v rámci aplikácie a useDispatch pre dispečovanie akcií Redux. V komponente Login sa nachádza jedna funkcia handleSubmit a je spúšťaná po kliknutí na tlačidlo „Prihlásiť sa“. Najprv sa vykonávajú kontroly platnosti údajov, a teda sa overuje, či bol zadán aj email a heslo a či sú tieto údaje správne. Po odoslaní požiadavky na server typu POST, ktorý obsahuje email a heslo v JSON formáte, sa čaká na odpoveď. Ak je odpoveď úspešná, používateľ je prihlásený a presmerovaný na komponent Home v opačnom prípade sa zobrazí chybová hláška. V aplikácii sme používali JWT token na autentifikáciu a udržanie stavu prihlásenia užívateľa aj po obnovení stránky, čím sme umožnili používateľovi zostať prihláseným bez nutnosti opakovane zadávania prihlasovacích údajov. K tomu využívame local storage na ukladanie JWT tokenu, čo nám umožnilo udržiavať prihlasovací stav medzi reláciami aplikácie. Vizuálny vzhľad komponentu sme definovali pomocou štýlov v CSS, ktoré nastavujú zarovnanie,

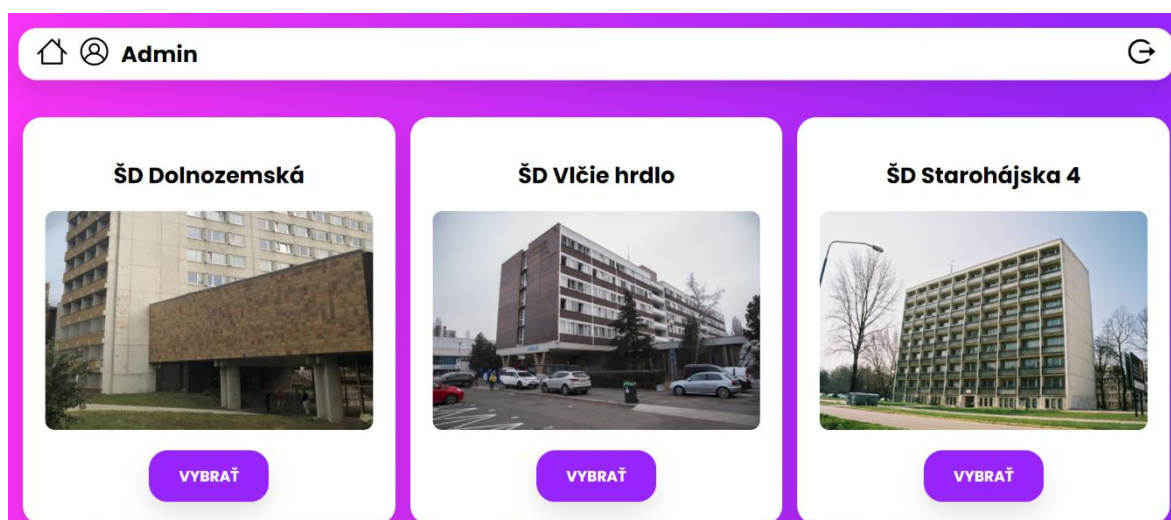
formátovanie vstupných polí a tlačidla, ako aj zobrazením spätných väzieb a stavov počas procesu prihlásenia.



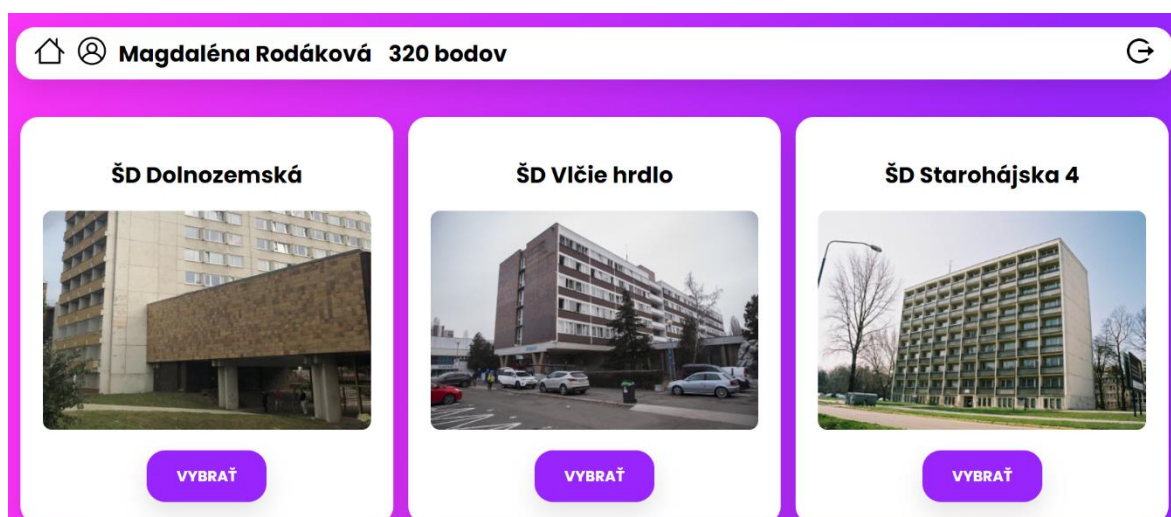
Obrázok 11: Stránka pre prihlasovanie (zdroj: vlastné spracovanie)

3.3.2 Domovská stránka

Hlavnou stránkou, ktorú používateľ ihneď po prihlásení uvidí je práve domovská stránka, na ktorej sa nachádzajú komponenty Home, User a Accommodation. V komponente Home sme importovali React a niekoľko potrebných funkcií, ako napríklad `useState` a `useEffect`, ktoré boli nevyhnutné pre prácu so stavmi a efektami. Taktiež sme importovali spomínané komponenty Accommodation a User. Samotný komponent Home obsahuje stavovú premennú `accommodations`, ktorá uchováva zoznam internátov načítaných zo servera respektíve databázy. Pomocou funkcie `internat()` sme načítali údaje o internátoch z API servera a uložili sme ich do stavu. Táto funkcia je volaná pri načítaní komponentu pomocou `useEffect()`. Po úspešnom načítaní dát sú tieto informácie zobrazené v komponente. Štýly pre komponent Home a jeho podkomponenty sú definované v súbore `App.css`. Napríklad, pre Home je definovaná trieda `home-container`, ktorá určuje rozloženie a pozadie domovskej stránky. Podkomponenty ako Accommodation majú špecifické triedy, ako `accommodation-box` pre kontajner, ktorý obsahuje zoznam internátov. Ďalšie štýly určujú vzhľad tlačidiel, zobrazenie informácií a ďalšie prvky na domovskej stránke. Tieto štýly prispievajú k celkovému vizuálnemu vzhľadu aplikácie a zabezpečujú konzistentný a atraktívny dizajn.



Obrázok 12: Domovská stránka pre admina (zdroj: vlastné spracovanie)



Obrázok 13: Domovská stránka pre študenta (zdroj: vlastné spracovanie)

User komponent sa zobrazuje na každej stránke okrem stránky prihlasovania, a sú na ňom zobrazené všetky dôležité informácie o používateľovi, dôležité prepojenia, ktoré zabezpečujú ikony. Po úspešnom prihlásení sa používateľovi priradí JWT token, ktorý obsahuje informácie o jeho identite a oprávneniach. Tento token je uložený v localStorage pre budúce použitie a identifikáciu používateľa aj po obnovení stránky. Ako pri predošlých komponentov, aj v tomto sme importovali niekoľko potrebných funkcií a knižníc z Reactu, ako napríklad useNavigate, useLocation pre navigáciu v aplikácii a useSelector, useDispatch pre prácu so stavom v Reduxe. Taktiež sme importovali akciu logout z reduceru UserReducer, ktorá zabezpečila odhlásenie používateľa. Komponent obsahuje viacero funkcií, ktoré sa spúšťajú po kliknutí na rôzne ikony. Napríklad funkcia handleLogout zabezpečuje odhlásenie používateľa a presmerovanie späť na prihlasovaciu stránku po kliknutí na tlačidlo „logout.svg“. Funkcie handleUserClick, handleUserStatusClick a

handleUserHelpClick presmerujú používateľa na rôzne podstránky v závislosti od toho, na ktorú časť používateľ klikne. V komponente je tiež implementovaná ikona, ktorá zabezpečuje návrat späť v histórii prehliadača pomocou funkcie handleBackClick, ktorá sa aktivuje kliknutím na toto tlačidlo. Samotné zobrazenie používateľských údajov, ako je meno, priezvisko a počet bodov, je podmienené ich dostupnosťou na serverovej strane. Pokiaľ sú údaje k dispozícii, zobrazia sa, inak sa zobrazí správa „Loading“. Namiesto toho, ak sú niektoré údaje v databáze null, nezobrazia sa vôbec. To znamená, že ak má admin v databáze napríklad meno "Admin", ale priezvisko a počet bodov sú null, zobrazí sa iba meno "Admin".

Používateľ má tiež možnosť kliknúť na ikonu domčeka, ktorá ho vráti na komponent Home. Ikona zobrazujúca používateľa presmeruje prihláseného používateľa s rolou študent rovno na Status komponent, ak je však prihlásený admin, zobrazí sa mu menu s ponukou na presmerovanie na komponenty Status alebo Add. O týchto komponentoch si povieme v nasledujúcich kapitolách. Všetky ikony sme stiahli vo formáte svg zo stránky www.icons8.com a uložili ich v našom vývojom prostredí do priečinka „images“. [17]



Obrázok 14: Používateľská lišta pre admina (zdroj: vlastné spracovanie)



Obrázok 15: Používateľská lišta pre študenta (zdroj: vlastné spracovanie)



Obrázok 16: Ukážka menu pre admina (zdroj: vlastné spracovanie)

Komponent Accommodation slúži na zobrazenie všetkých internátov, ktoré sú v databáze dostupné. Aj v tomto komponente importujeme React a potrebné funkcie z knižnice react-router-dom pre prácu s navigáciou v aplikácii. Taktiež importujeme súbor App.css na definovanie vizuálneho vzhľadu komponentu. V komponente Accommodation sme definovali jednu funkciu, ktorá prijíma ako parameter data, čo sú údaje o internáte. V jednom boxe, kde sa zobrazuje internát, sa z databázy zobrazuje názov internátu a jeho úvodná fotka. Fotky sme s povolením ŠP EUBA stiahli zo stránky www.sp.euba.sk. Fotka je načítaná v base64 formáte a zobrazuje sa pomocou elementu. Okrem toho je tu

tlačidlo "Vybrať", ktoré slúži ako odkaz na stránku s detailnými informáciami o internáte. Tlačidlo "Vybrať" je zapuzdrené v značke `<Link>` z modulu "react-router-dom", ktorý vytvára odkaz. Tento odkaz obsahuje URL adresu, ktorá zahŕňa identifikátor konkrétneho internátu. Týmto spôsob umožňuje používateľovi po kliknutí na tlačidlo prejsť na stránku s detailmi o konkrétnom internáte. Komponent je štylovaný pomocou rôznych formátových vlastností v App.css, kde sú nastavené rôzne vlastnosti, ako napríklad rozmiestnenie, veľkosť, okraje a pozadie jednotlivých elementov.



Obrázok 17: Ukážka zobrazenia internátu na domovskej stránke (zdroj: vlastné spracovanie)

3.3.3 Stránka s informáciami o internátoch

Na podstránke s informáciami o internáte sme pracovali s komponentom Info, kde sme využili funkcie poskytované Reactom, ako sú `useState`, `useEffect` a `useParams` z knižnice "react-router-dom". Taktiež sme importovali knižnicu "leaflet", ktorú budeme potrebovať na zobrazenie mapy pre jednotlivý internát. Komponent Info zobrazuje detaily konkrétneho internátu na základe jeho identifikátora (id), ktorý získavame pomocou `useParams`. V komponente sa načítavajú informácie o internáte a jeho fotografie API pomocou asynchrónnych funkcií `fetchData` a `fetchPhotos`. Po úspešnom načítaní sa tieto údaje zobrazia na stránke. Okrem toho sme implementovali funkciu pre navigáciu na ďalšiu stránku po kliknutí na tlačidlo "Vybrať izbu", ktoré presmeruje používateľa na výber samotnej izby. Komponent tiež obsahuje mapu zobrazenú pomocou knižnice "leaflet", ktorá zobrazuje polohu internátu na základe jeho súradníc, ktoré sa získali z databázy. *Leaflet je open-source JavaScript knižnica na tvorbu interaktívnych mapových aplikácií pre webové stránky.* [18]

Magdaléna Rodáková

320 bodov

Info

VYBRAŤ IZBU

V internáte je k dispozícii ubytovanie pre študentov za rôzne mesačné poplatky v závislosti od typu bloku. Bloky A, B a C ponúkajú ubytovanie za 120,00 Eur mesačne na osobu. Blok C, dokončený v roku 2015, má mesačný poplatok 105,00 Eur na osobu. Bloky D, E a F majú mesačný poplatok 105,00 Eur za osobu v prípade bloku D a 100,00 Eur za osobu v prípade blokov E a F. Internát pozostáva z celkovo piatich blokov, označených písmenami A, B, C, D, E a F. Každý blok je vybavený bunkovým systémom, kde každá bunka zahŕňa dve 2-lôžkové izby, kúpeľňu s WC a kuchynku vybavenú chladničkou a indukčnou jednoplatničkou. V každej izbe je dostupná dvojportová prípojka do siete s možnosťou inštalácie vlastného routera. Okrem ubytovacích priestorov internát ponúka aj rôzne spoločné priestory vrátane vrátnice s možnosťou tlače a kopírovania, dvoch výťahov, pracovne, troch študovní, barového bufetu a ďalších zariadení. V areáli internátu nájdete aj telocvičňu s možnosťou prenájmu, saunu, multifunkčné ihrisko a malý park s ohniskom. Internát je dobre napojený na verejnú dopravu. Z Hlavnej stanice je možné sa dostať na internát autobusovými linkami č. 41 s výstupom na zastávke Hroboňova alebo Prokopa Veľkého. Zastávka Pod stanicou umožňuje prístup k linkám č. 83/84/63. Študenti majú tiež možnosť dopraviť sa z internátu do školy pomocou autobusových liniek č. 184 s výstupmi pri Ekonomickej Univerzite alebo č. 84, a Bulikovej. Okrem toho, je možné dopraviť sa do centra mesta pomocou liniek č. 84/83 s výstupom na zastávke Hodžovo námestie alebo Zochova. Na nákupy môžu študenti využiť potraviny ZAP pri zastávke Prokopa Veľkého, Tesco Lamač alebo Kaufland.

Fotky

Obrázok 18: Ukážka zobrazenia informácií o internáte (zdroj: vlastné spracovanie)

Magdaléna Rodáková

320 bodov

Blok A a B

Obrázok 19: Ukážka zobrazenia fotiek internátu (zdroj: vlastné spracovanie)

Mapa

VYBRAŤ IZBU

Obrázok 20: Ukážka zobrazenia polohy internátu na mape (zdroj: vlastné spracovanie)

3.3.4 Stránka pre výber izby

Na stránke, kde si študenti budú môcť zvoliť izbu, sme použili komponent `Select`, vďaka ktorému si používateľ vyselektuje požadovanú izbu a taktiež komponent `Room`, ktorý predstavuje jednotlivé izby. V komponente `Select` sme využili stavové premenné a efekty z `Reactu`, ako napríklad `useState` a `useEffect`, na správu filtrov a na načítanie zoznamu izieb z internátov. Po načítaní internátov sa tieto údaje používajú na vytvorenie filtra. Pri zmene hodnôt filtrov sa volá funkcia `handleFilterChange`, ktorá aktualizuje stav filtrov. Komponent `Select` obsahuje filter s rôznymi možnosťami, ako napríklad výber internátu podľa názvu, orientácie, bloku, poschodia, typu izby, stavu rekonštrukcie a umiestnenia na chodbe. Po zvolení hodnôt vo filtri sa zobrazia izby podľa zvolených filtrov. Okrem toho sme implementovali funkciu `handleResetFilters`, ktorá vymaže všetky zvolené filtre a nastaví ich na počiatočný stav. Celkovým cieľom komponenty `Select` je umožniť používateľovi filtrovať a vyberať izby na základe rôznych kritérií a zobrazit' výsledky v komponente `Room`, o ktorom si povieme v nasledujúcej kapitole.

V komponente `Room` sme znova využili stavové premenné pomocou funkcie `useState` na uchovávanie dát o izbách, vybranej izbe a stavu požiadavky používateľa. Taktiež sme použili `useEffect` na načítanie údajov o izbách. Komponent tiež využíva `Redux` `useSelector` na získanie údajov o prihlásenom používateľovi. V komponente sa tiež vykonáva filtrovanie dostupných izieb na základe nastavených filtrov. Výsledné izby sú zobrazené vo forme boxov, ktoré obsahujú informácie o izbe a tlačidlo na výber izby. Ak sa nenájde žiadna izba s kritériami, ktoré si zvolil používateľ, vypíše sa hláška „Kritériám nevyhovujú žiadne izby.“. Funkcia `handleSelectRoom` zabezpečuje, že ak používateľ klikne na tlačidlo „Zvoliť“ pre danú izbu, vykoná sa príslušná akcia. V tomto prípade akcia zahŕňa overenie, či používateľ už nepožiadaval alebo nemá schválenú žiadosť o izbu, a ak nie, odošle sa požiadavka na server, aby sa aktualizovala tabuľka „žiadosti“ v databáze s novou žiadosťou o túto izbu. Taktiež zobrazuje upozornenia v prípade úspešne vybranej izby alebo chyby, ak si napríklad používateľ už nejakú izbu zvolil. Avšak ak je používateľ prihlásený ako admin, nemôže si izbu zvoliť.

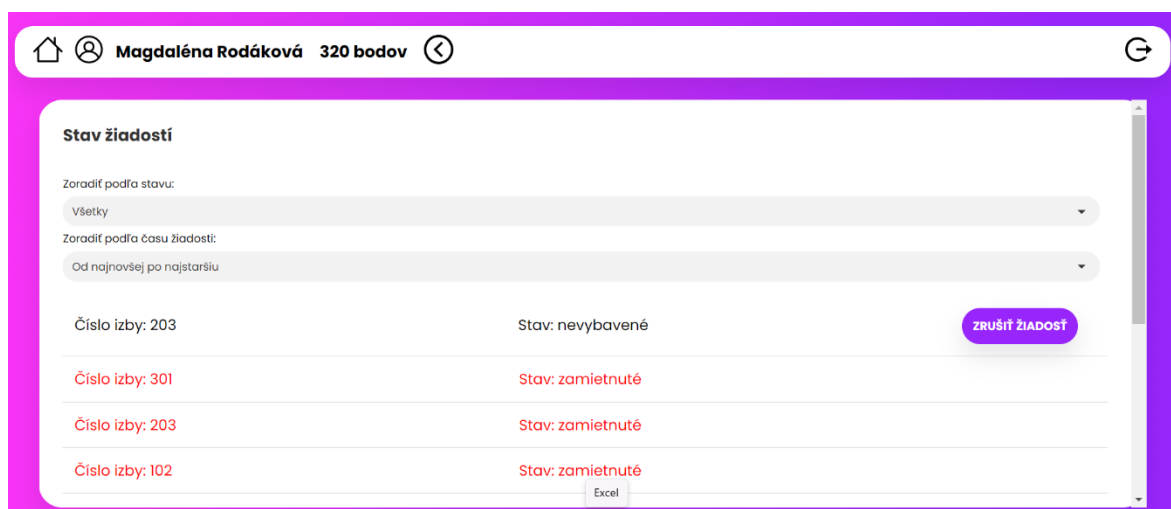
Internát	Číslo izby	Poschodie	Orientácia	Typ izby	Cena	Stav
ŠD Vičie hrdlo	205	4.poschodie	Sever	Dvojľôžková	85€	Nezrekonštruovaná
ŠD Starohájka 8	202	2.poschodie	Juh	Dvojľôžková	72€	Nezrekonštruovaná
ŠD Ekonóm	401	2.poschodie	Sever	Trojľôžková	100€	Nezrekonštruovaná
ŠD Hroboňova 4	102	3.poschodie	Juh	Dvojľôžková	70€	Nezrekonštruovaná

Obrázok 21: Ukážka podstránky pre výber izby (zdroj: vlastné spracovanie)

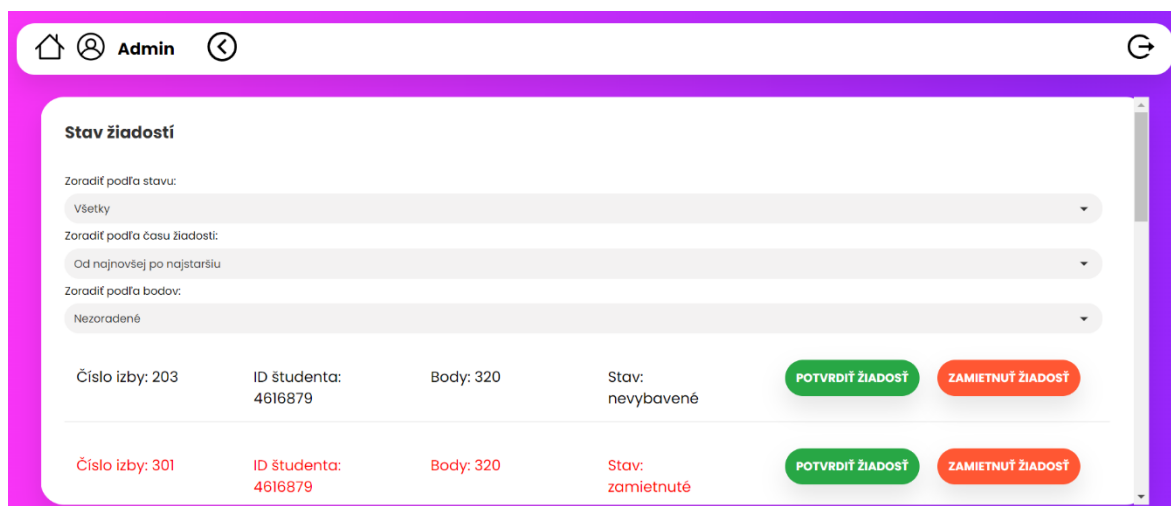
3.3.5 Stránka pre správu a sledovanie žiadostí

Na túto stránku sa používateľ vie dostať cez ikonu používateľa z User komponentu, o ktorom sme si povedali v predošlej kapitole. V tejto stránke sme použili komponent Status. V komponente Status sme implementovali funkcionality na zobrazenie stavu žiadostí o izby. Používateľ, ktorý je prihlásený ako študent vidí vo svojej žiadosti číslo izby a stav žiadosti. Admin vidí nie len číslo izby a stav žiadosti, ale aj id daného študenta a jeho počet bodov. Použili sme stavové premenné pomocou funkcie `useState` na uchovávanie dát o žiadostiach a filtroch. Taktiež sme využili `Redux useSelector` na získanie údajov o prihlásenom používateľovi, aby sa vedeli správne zobrazovať žiadosti. Následne sme implementovali funkcie pre zrušenie, schválenie a zamietnutie žiadostí. Zrušenie žiadosti je umožnené len používateľovi s rolou študent, a taktiež len ak je žiadosť ešte v stave „nevybavená“. Po schválení respektíve zamietnutí izby tlačidlo „Zrušiť žiadosť“ zmizne. Tlačidlá schválenie a zrušenie sú zobrazené len používateľovi s rolou admin. Aj po schválení alebo zamietnutí vie admin stále narábať s danou žiadosťou, aby sa ošetril prípad, že sa žiadosť schváli respektíve zamietne omylom. Komponent obsahuje tiež možnosť filtrovania žiadostí podľa ich stavu pre lepší prehľad nie len pre admina, ktorý spravuje žiadosti, ale aj pre študenta. Po zvolení filtra sa zobrazia len žiadosti, ktoré zodpovedajú zvolenému stavu. V tomto komponente je aj filter pre zoradenie žiadostí podľa času, kde predvolené sú zoradené od najnovšej po najstaršiu žiadosť. Je možné ich zoradiť od najstaršej po najnovšiu. Používateľ môže zoradiť žiadosti podľa počtu bodov od najväčšieho počtu bodov po najmenší alebo opačne. Celkovým cieľom komponentu Status je poskytnúť používateľovi prehľad o stave

jeho žiadostí a umožniť mu vykonávať príslušné akcie na základe stavu žiadostí.



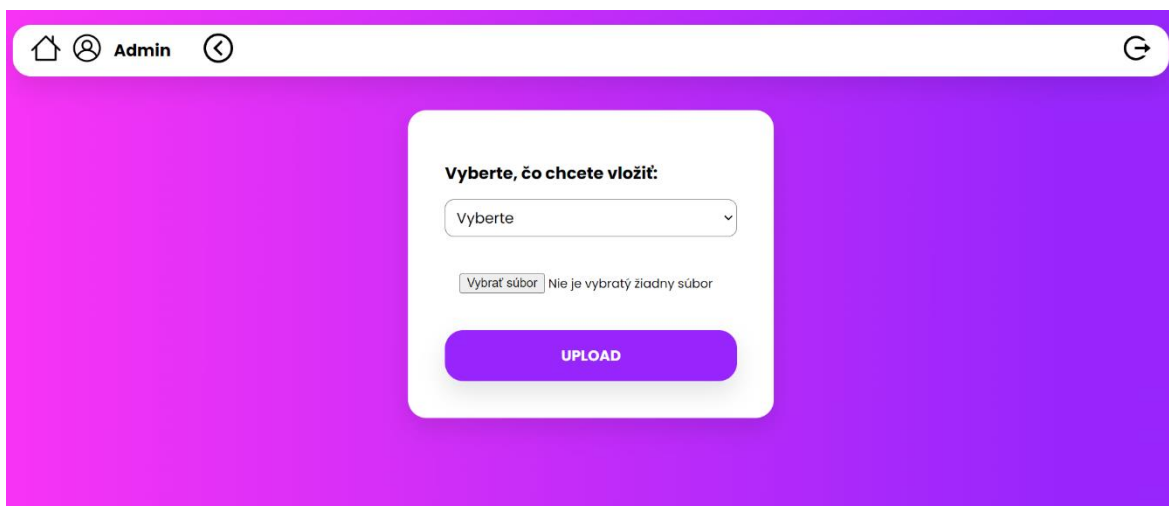
Obrázok 22: Stránka pre sledovanie žiadostí pre študenta (zdroj: vlastné spracovanie)



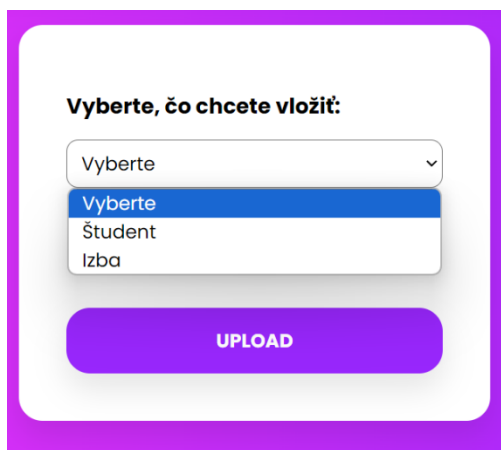
Obrázok 23: Stránka pre sledovanie žiadostí pre admina (zdroj: vlastné spracovanie)

3.3.6 Stránka pre import csv súboru

Táto stránka slúži len pre admina. Využívali sme tu komponent Add. Na túto stránku sa používateľ dostane len ak má rolu admin. Slúži na to, aby admin mohol do databázy jednoducho cez csv súbor importovať potrebné údaje. Na tejto stránke sa adminovi zobrazí text „Vyberte, čo chcete vložiť:“ spolu s rolovacím menu s možnosťami importovať študentov alebo izby. Cez tlačidlo „Vybrať súbor“ admin zvolí vhodný csv súbor a cez tlačidlo „Upload“ tento súbor pošle na server pomocou POST požiadavky, ktorý túto požiadavku spracuje.



Obrázok 24: Stránka pre import csv súboru (zdroj: vlastné spracovanie)



Obrázok 25: Ukážka menu pre typ csv súboru (zdroj: vlastné spracovanie)

3.4 Serverová časť

Serverová časť je taktiež nevyhnutnou súčasťou našej modernej webovej aplikácie. Je zodpovedná za spracovanie požiadaviek a interakciu s databázou.

Na tento účel sme sa rozhodli využiť technológiu Node.js, pretože je známy pre svoju rýchlosť a efektívnosť. Node.js umožňuje vykonávať JavaScript na strane servera, čo zjednodušuje vývoj a umožňuje používanie jedného jazyka na oboch stranách aplikácie. Node.js je ideálny pre vývoj real-time aplikácií, ako sú chaty, online hry alebo kolaboratívne nástroje, vďaka svojej schopnosti efektívne spravovať súbežné udalosti. Node.js je široko využívaný v spoločnostiach, ako sú Netflix, LinkedIn, PayPal a ďalšie, čo potvrdzuje jeho spoľahlivosť. [19]

Pre jednoduché a efektívne vytváranie webových aplikácií a API sme zvolili populárny a minimalistický framework pre Node.js Express.js. Tento framework poskytuje množstvo nástrojov na rýchle a ľahké vytváranie serverových aplikácií v JavaScripte. Express.js je ľahko integrovaný s rôznymi databázami, vrátane MongoDB, MySQL či PostgreSQL. Vďaka tomu môžeme tvoriť komplexné webové aplikácie s trvalým ukladaním informácií. [20]

Ďalším dôležitým aspektom serverovej časti je navrhovanie rozhraní pre komunikáciu medzi systémami, kde sme sa rozhodli implementovať REST API, čo je štandardný spôsob navrhovania rozhraní. Používa HTTP metódy, ako sú GET, POST, PUT a DELETE pre manipuláciu so zdrojmi. Tento prístup umožňuje efektívnu komunikáciu medzi klientom a serverom. [21]

3.4.1 *Koncové body*

V adresári backend sme vytvorili súbor s názvom server.js, kde sme definovali rôzne endpointy resp. koncové body, ktoré slúžia na komunikáciu medzi backendom a frontendom našej webovej aplikácie. Každý endpoint má špecifickú úlohu a zabezpečuje určitú funkcionality aplikácie. Nižšie sú spomenuté všetky endpointy a ich funkcie v našej aplikácii:

1. POST /login: Tento koncový bod slúži na overenie prihlasovacích údajov používateľa. Po prijatí prihlasovacích údajov od používateľa sa vykoná dotaz z databázy, kde sa načítajú úplne všetky stĺpce z tabuľky student na overenie správnosti údajov a ak sú údaje správne, používateľ bude úspešne prihlásený. Ak sa údaje, ktoré zadal používateľ na frontende, nebudú zhodovať s tými z databázy, vypíše sa hláška o neúspešných prihlasovacích údajoch. V tomto endpointe získame úplne všetky údaje, ktoré budeme potrebovať nie len pri prihlásení, ale počas celej aplikácie, či už kvôli roli používateľa, alebo mena, priezviska a počtu bodov. Po úspešnom overení prihlasovacích údajov je vygenerovaný JWT token, ktorý je vrátený v odpovedi ako súčasť úspešného prihlásenia. JWT tokeny sa používajú hlavne na overovanie identity používateľov a autorizáciu prístupu.[22]
2. GET /home: Tento jednoduchý endpoint slúži na získanie informácií o internátoch z databázy. Po prijatí požiadavky sa vykoná dotaz z databázy z tabuľky internat a vráti sa zoznam internátov s ich id, názvom, popisom a úvodnou fotkou.

3. GET /info/:id: Endpoint slúži na získanie všetkých informácií o konkrétnom internáte na základe jeho jedinečného id. Po prijatí id sa vykoná dotaz do databázy z tabuľky internat a vrátia sa všetky informácie o danom internáte. Ak sa nenájde internát podľa zadaného dotazu z databázy, vypíše sa hláška, že internát nebol nájdený.
4. GET /izba: V tomto endpointe sme získavali zoznam všetkých izieb a ich vlastností. Tradične sa vykoná dotaz z databázy. V tomto dotaze pristupujeme k dvom tabuľkám, kde sme získali všetky informácie o izbách z tabuľky izba vrátane názvu internátu, ku ktorému patria z tabuľky internat. Tieto dve tabuľky sme spojili vďaka JOIN na základe zhody id internátov. Následne sa vráti zoznam izieb s ich atribútmi. Izby, ktoré sú momentálne priradené k žiadosti so stavom "nevybavené" alebo "schválené", sa nezobrazia v zozname, čím sa ošetrí prípad, že si niekto iný bude môcť zvoliť izbu, ktorú si už zvolil iný študent, hoc izbu nemá ešte schválenú. Akonáhle bude žiadosť zamietnutá, izba sa opäť zobrazí. V prípade chyby pri vykonávaní dotazu sa vráti odpoveď so statusom 500 a správou o chybe tak ako to je zvykom v každom našom endpointe.
5. POST /select/:id_izba: Tento endpoint slúži na výber izby používateľom. Najprv sa získa rola z tabuľky student. Ak je táto rola „admin“, vypíše sa hláška, že Admin si nemôže vybrať izbu. Ďalej sa zisťuje, či si študent už zvolil izbu alebo ju už má schválenú. To sa zisťuje dotazom, kde sa vyberie stĺpec maizbu z tabuľky student. Ak sa v tomto stĺpci nachádza hodnota „Schválené“ alebo „Požiadané“, vypíše sa hláška, že si študent už izbu vybral. Ak sa tam nachádza „Nepožiadané“ alebo „Zamietnuté“, povolí sa študentovi výber izby. Po výbere izby sa zapíše nová žiadosť do tabuľky ziadosti, kde sa nastaví aj stav „nevybavené“. Taktiež sa vykoná update pre stĺpec maizbu v tabuľke student, kde sa v tomto stĺpci prepíše predchádzajúca hodnota na „Požiadané“. Na konci sa vypíše, že izba bola úspešne vybratá. Ak sa nezíska stav z maizbu, vypíše sa hláška, že sa nepodarilo získať stav izby pre vybraného študenta.
6. GET /requests/:id_student: Endpoint získava zoznam všetkých žiadostí o ubytovanie pre konkrétného študenta na základe jeho id. Predtým sa však zistí, či je daný používateľ admin alebo študent podľa role z tabuľky student. Ak je rola prihláseného používateľa admin, vidí úplne všetky žiadosti, ak je rola študent, vidí len svoje žiadosti o danú izbu.

7. GET /fotky/:id_internat: Tento endpoint slúži na získanie fotiek konkrétneho internátu na základe jeho id. Po prijatí požiadavky sa vykoná dotaz do databázy a vrátia sa fotky príslušného internátu. Ak sa vyskytne chyba pri získavaní fotiek, vypíše sa chybová hláška.
8. PUT /approve/:id: Tento endpoint definuje spracovanie PUT požiadavky na schválenie konkrétnej žiadosti. Server najskôr získa id žiadosti a id študenta. Potom sa overí, či daná žiadosť v tabuľke existuje, ak nie, vypíše sa chybová hláška, že sa žiadosť nenašla. Ak ale žiadosť existuje, tak sa aktualizuje stĺpec id_izba v tabuľke student, kde sa zapíše pre daného študenta id_izby, ktorú si zvolil. Ďalej sa aktualizuje tabuľka izba, kde sa do stĺpca id_student zapíše id študenta. Stav žiadostí v tabuľke žiadosti sa prepíše na „schválené“ a taktiež sa „Schválené“ zapíše do stĺpca maizbu v tabuľke student, čím sa mu znemožní opätovné vybratie si izby.
9. PUT /reject/:id : V tomto endpointe je implementované zamietnutie žiadosti. Znova sa najprv overí, či existuje daná žiadosť. Ak neexistuje, vypíše sa chybová hláška a ak existuje, aktualizujú sa stĺpce v databáze. Stav v tabuľke žiadosti sa prepíše na „zamietnuté“ a v tabuľke student v stĺpci maizbu sa zmení predošlá hodnota na „Zamietnuté“ a umožní sa tak znova výber izby. Pre prípad, že by bola najprv žiadosť schválená a následne zamietnutá, je preventívne potrebné po zamietnutí izby nastaviť stĺpec id_izba v tabuľke student na NULL, a to isté je potrebné pre tabuľku izba pre stĺpec id_student.
10. DELETE /request/:id: V endpointe odstraňujeme konkrétnu žiadosť o ubytovanie na základe jej id. Po prijatí požiadavky sa vykoná odstránenie záznamu z databázy. Ak sa nenájde daná žiadosť, vypíše sa, že žiadosť nebola nájdená. Ak sa žiadosť nájde, aktualizuje sa stav stĺpca maizbu pre daného študenta, ktorému bola žiadosť priradená, na hodnotu „Nepožiadané“, čím sa mu povolí opätovne si vybrať izbu.
11. POST /upload-students : V tomto endpointe nahrávame informácií o študentoch zo súboru vo formáte CSV do databázy. Po prijatí požiadavky sa súbor spracuje a jeho obsah sa ukladá do tabuľky student v databáze. Každý riadok súboru reprezentuje jedného študenta s atribútmi ako id študenta, meno, priezvisko, email, heslo, body a rola. Ak nastane chyba pri nahrávaní csv súboru či dát do databázy, vypíšu sa príslušné errorové hlášky.
12. POST /upload-room : Podobný princíp ako v predošlom endpointe je aj v tomto koncovom bode. Nahrávame v ňom údaje o izbách do tabuľky izba. Ak je csv súbor

v poriadku, obsah súboru sa úspešne spracuje a údaje sa ukladajú do databázy do tabuľky izba. Ak nie, vypíšu sa rovnaké hlášky ako v predchádzajúcom endpointe.

13. POST /me : Tento endpoint umožňuje získať informácie o aktuálne prihlásenom používateľovi na základe poskytnutého JWT tokenu. Po prijatí požiadavky obsahujúcej token sa overí jeho platnosť a pomocou neho sa získa id používateľa. Následne sa vyhľadajú všetky informácie o používateľovi v databáze z tabuľky student a vrátia sa ako odpoveď. Ak nastane chyba pri spracovaní požiadavky na endpointe, čo môže zahŕňať napríklad chybu pri overovaní platnosti JWT tokenu alebo pri vyhľadávaní informácií o používateľovi v databáze, vypíšu sa errorové hlášky.

3.5 Zálohovanie

Na zálohovanie našej aplikácie sme sa rozhodli použiť GitHub Desktop aplikáciu. GitHub je platforma na hosting softvérových projektov, ktorá slúži na správu kódov a kolaboráciu vývojárov. Umožňuje vývojárom ukladať, spravovať a zdieľať svoje zdrojové kódy a projekty. Tlačidlá "Commit" a "Push" sú kľúčové funkcie v GitHub Desktop aplikácii, ktoré slúžia na prácu s lokálnymi a vzdialenými repozitármi na platforme GitHub. Pri commitovaní sú zmeny zapísané do histórie repozitára s popisom, ktorý približuje, čo sa v danom commite zmenilo. Táto funkcia umožňuje vývojárom sledovať vývoj projektu a späť sa vrátiť k predchádzajúcim verziám kódu v prípade potreby. Push zas odosiela zmeny z lokálneho repozitára na serverový repozitár na GitHub. Táto funkcia umožňuje vývojárom zdieľať svoj kód s ostatnými členmi tímu alebo verejnosťou. Po odoslaní sú zmeny dostupné pre ostatných členov tímu. Po stiahnutí a nainštalovaní tejto aplikácie sme vytvorili nový lokálny repozitár. Do tohto repozitára sme pridali všetky dôležité súbory a priečinky, ktoré sme chceli zálohovať. Po pridaní sme vytvorili "Commit" s popisom zmien pre lepšie identifikovanie v budúcnosti a odoslali ich na server GitHubu pomocou možnosti "Push". Je veľmi dôležité pravidelne aktualizovať kód, aby zodpovedal aktuálnemu stavu dát a aby sme napríklad pri zlyhaní počítača neprišli o všetok kód. Pomocou GitHub Desktop aplikácie sme vedeli jednoducho sledovať zmeny v repozitári a pravidelne ich aktualizovať. [23]

3.6 Výzvy

Počas implementácie práce sme sa stretli s mnohými výzvami, ktoré nás donútili prispôbiť a zdokonaľiť našu webovú aplikáciu. Keďže vývoj webovej aplikácie je veľmi

komplexný proces, museli sme si stanoviť prioritizáciu najdôležitejších funkcií a zabezpečiť tak včasnú implementáciu. Napriek našim úsiliam sme sa však stretli s niekoľkými nedostatkami a potrebami, ktoré bolo nevyhnutné riešiť.

Jedným z hlavných problémov, s ktorým sme sa stretli, bolo navrhnutie a implementácia ubytovacieho systému, ktorý by splnil všetky požiadavky a návrhy našich používateľov. Jednou z vecí, ktoré v aplikácii chýbajú a bolo by ich vhodné doplniť je stabilizácia izby, ktorá by bola doplnená spolu s notifikačným systémom, ktorý by študentov upozornil na dôležité dátumy a akcie spojené s ubytovacím procesom. To by znamenalo, že ak študent bude mať schválenú žiadosť o ubytovanie, teda ešte nie o samotnú izbu, v aplikácii by sa mu po prihlásení zobrazila notifikácia, kde by si odklikol, či si chce izbu, ktorú si vybral v predošlom akademickom roku, stabilizovať aj na ďalší. Ak áno, izba sa mu automaticky znova prideli a nebude si musieť vybrať novú. Avšak ak sa rozhodne, že si izbu nechce ponechať, bol by zaradený do procesu výberu izby na základe počtu bodov.

Okrem toho sme si uvedomili, že neexistuje efektívny proces hromadného pridávania študentov alebo izieb do aplikácie, čo by mohlo spomaľovať celý proces. Rozhodli sme sa to zmeniť, inak by administrátor musel pridávať tieto údaje cez príkazy v databáze a pri veľkom počte vkladáných údajov by to nebolo efektívne.[24]

Veľmi dôležitou vecou bolo aj upraviť aplikáciu pre rôzne rozlíšenia, teda zabezpečiť jej responzivnosť. Museli sme sa uistiť, že naša aplikácia bude schopná adekvátne reagovať na zmeny vo veľkosti obrazovky a zobrazíť sa správne na rôznych zariadeniach, vrátane počítačov, tabletov a mobilných telefónov. Týmto spôsobom sme mohli zabezpečiť optimálny používateľský zážitok bez ohľadu na to, aké zariadenie používateľ používa na prístup k našej aplikácii.[25]

Napriek výzvam, ktoré nás postretli pri vývoji, sme ich dokázali včas riešiť a implementovať. Tieto výzvy nás donútili hľadať kreatívne riešenia, aby sme dosiahli najhlavnejšie ciele.

3.7 Ďalšie možnosti rozšírenia

Ubytovací systém je oveľa komplexnejšia téma a existuje mnoho rozšírení a funkcionalít, ktoré by boli vhodné.

1. Implementáciou chatu v aplikácii by sme umožnili študentom komunikovať medzi sebou ale aj medzi parlamentom. Vďaka tomu by sa zabezpečila rýchla výmena informácií, riešenie problém a nedostatkov.
2. Plánovač rôznych aktivít a udalostí, ktoré sa budú týkať nie len internátov a ubytovania by študenti určite ocenili. Aplikácia by sa tým stala komplexnejšia vo svojej oblasti a poskytovala by široké spektrum informácií na jednom mieste.
3. Notifikačný systém by študentov upozornil na dôležité dátumy týkajúce sa podania žiadosti, výberu izby alebo informovania o pridelení respektíve nepridelení ubytovania.
4. Definovanie kôl pre výber izby, aby sa zachovala čo najväčšia hierarchia pri výbere. V databáze v tabuľke student by sa pridal stĺpec „kolo“, kde by každý študent mal definované poradie, o ktorej sa mu sprístupní výber izby. Začiatok kôl a ich počet si definuje parlament podľa počtu pridelovaných miest. Potom sa v aplikácii implementuje logika, ktorá zistí aktuálny čas a porovná ho s definovanými časovými intervalmi pre kola. Na základe toho sa zistí, ktoré kolo práve prebieha.

Tieto návrhy na ďalšie možnosti rozšírenia predstavujú len niektoré z mnohých, ako by sa naša webová aplikácia mohla v budúcnosti ďalej. S neustálym vývojom sa otvára mnoho príležitostí na ďalšie inovácie a vylepšenia.

Záver

V tejto práci sme sa venovali detailnému popisu súčasného stavu ubytovacích systémov univerzít a implementácií nového komplexnejšieho systému. Po dôkladnej analýze aktuálnych technológií sme sa rozhodli použiť pre vývoj React.js pre frontend, Express.js pre backend a PostgreSQL pre správu databázy.

V našej práci sme detailne analyzovali ubytovací systém troch najväčších univerzít v Bratislave, vďaka čomu sme ich viac pochopili a vedeli ich lepšie porovnať. Identifikovali sme silné, ale aj slabé stránky každého z nich. Následne sme študentom distribuovali dotazník ohľadom nedostatkov a návrhov zlepšenia tohto systému, aby sme ich pri implementácii vedeli čo najviac zohľadniť. Vďaka tomu sme vytvorili návrh nového riešenia priamo na základe potrieb študentov. Tohto návrhu sme sa potom držali pri samotnej implementácii webovej aplikácie.

Výsledkom tejto práce je webová aplikácia, kde si študenti môžu prezerať internáty, ktoré ponúka ich univerzita. Môžu si prečítať o každom z nich dostatočné množstvo informácií, prezerať si fotky či vidieť priamo na mape, kde sa internát nachádza. Vďaka tomu si vedia vybrať internát, ktorý bude spĺňať ich požiadavky. Pri výbere konkrétnej izby si môžu vďaka veľkému množstvu filtrov čo najlepšie vybrať izbu podľa ich kritérií. Pre študentov je v aplikácii dostupné aj sledovanie všetkých podaných žiadostí študenta. Pre admina je v aplikácii vytvorená možnosť zamietnuť a schváliť žiadosti študentov na základe bodov, ktoré vidia priamo v aplikácii pri danej žiadosti. Môžu si vyfiltrovať žiadosti podľa počtu bodov vzostupne alebo zostupne. Taktiež môžu filtrovať aj študenti a admin podľa času žiadosti alebo jej stavu. Administrátori tiež majú v aplikácii možnosť importovať naraz všetky potrebné údaje pomocou csv súboru. Do aplikácie tak majú prístup len tí študenti, ktorých administrátori pre dané fakulty importovali na základe podaných žiadostí v AiS. To zvyšuje bezpečnosť aplikácie a taktiež to, že do aplikácie nemajú prístup aj tí študenti, ktorí si žiadosť nepodali.

Napriek snahe o optimalizáciu má naša webová aplikácia stále priestor na zlepšenie. Existuje množstvo potenciálnych vylepšení, ktoré by ešte viac zvýšili komplexnosť a efektivitu celého procesu. S implementáciou týchto návrhov by sme dosiahli ešte vyššiu automatizáciu a používateľský komfort.

Zoznam použitej literatúry

1. **DAŇO, Ferdinand.** *Zásady pridelovania ubytovania v študentských domovoch Ekonomickej univerzity v Bratislave.* [online] [cit. 15.10.2023]. Dostupné na internete: https://euba.sk/www_write/files/SK/sluzby/ubytovanie/2023/b-zasady_pridelovania_ubytovania_2023.pdf
2. **ŠTEVČEK, Marek.** *Vnútorne predpisy Univerzity Komenského v Bratislave* [online] [cit. 15.10.2023]. Dostupné na internete: https://uniba.sk/fileadmin/ruk/legislativa/2023/Vp_2023_26.pdf
3. **STANKO, Štefan.** *Kritériá pre pridelovanie ubytovania študentom denného štúdia na STU* [online] [cit. 17.10.2023]. Dostupné na internete: https://www.stuba.sk/new/docs/stu/pracoviska/uz_sdaj/Kriteria_STU.pdf
4. **ALKHALDI, Daniyah.** In: *Developing and Implementing Web-based Online University Facilities Reservation System* [online]. 2018. [cit. 22.11.2023]. Dostupné na internete: http://www.ripublication.com/ijaer18/ijaerv13n9_27.pdf
5. **GERMAN, Josephine D., YAP, Dave Cullen, BINOYA, Guiller O.** Design and development of an integrated room reservation system for Higher Education Institutions. In: *2021 IEEE 8th International Conference on Industrial Engineering and Applications (ICIEA)*. 2021 [cit. 23.11.2023]
6. **Node.js.** *Node.js - an introduction to the NPM package manager.* [online] [cit. 08.02.2024]. Dostupné na internete: <https://nodejs.org/en/learn/getting-started/an-introduction-to-the-npm-package-manager>
7. **Prettier.** *Why prettier?* [online] [cit. 09.02.2024]. Dostupné na internete: <https://prettier.io/docs/en/why-prettier>
8. **PostgreSQL.** *About.* [online] [cit. 12.02.2024]. Dostupné na internete: <https://www.postgresql.org/about/>
9. **JUBA, Salahaldin, VOLKOV, Andrey, VANNAHME, Achim.** *Learning postgresql: Create, develop, and manage relational databases in real-world applications using PostgreSQL.* Birmingham, UK: Packt Publishing, 2015. [cit. 12.02.2024].
10. **PAGE, Dave.** *Documentation* [online] [cit. 14.02.2024]. Dostupné na internete: <https://www.pgadmin.org/docs/>
11. **Release Notes RSS.** *Introduction: Dbdiagram docs.* [online] [cit. 23.03.2024]. Dostupné na internete: <https://dbdiagram.io/docs/>

12. **M., Vipul A., SONPATKI, Prathamesh.** *ReactJS by example-- building modern web applications with react: Get up and running with reactjs by developing five cutting-edge and responsive projects.* Birmingham: Packt Publishing, 2016. [cit. 20.02.2024]
13. **MOZDEVNET.** *HTML: Hypertext markup language: MDN.* In: *MDN Web Docs* [online] [cit. 03.03.2024]. Dostupné na internete: <https://developer.mozilla.org/en-US/docs/Web/HTML>
14. **MOZDEVNET.** *CSS Basics - learn web development: MDN.* In: *MDN Web Docs* [online] [cit. 03.03.2024]. Dostupné na internete: https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics
15. **Javascript.** *JavaScript tutorial* [online] [cit. 08.03.2024]. Dostupné na internete: <https://www.w3schools.com/js/>
16. **Redux.** *Getting started with Redux.* [online] [cit. 18.04.2024]. Dostupné na internete: <https://redux.js.org/introduction/getting-started>
17. **Icons.** *Home icons, logos, symbols – free download PNG, SVG.* [online] [cit. 08.04.2024]. Dostupné na internete: <https://icons8.com/icons/set/home>
18. **Leaflet.** *Tutorials - leaflet - a JavaScript library for interactive maps.* [online] [cit. 12.04.2024]. Dostupné na internete: <https://leafletjs.com/examples.html>
19. **Node.js.** *Node.js - about node.js.* [online] [cit. 11.04.2024]. Dostupné na internete: <https://nodejs.org/en/about>
20. **Express routing.** *Routing.* [online] [cit. 14.04.2024]. Dostupné na internete: <https://expressjs.com/ko/guide/routing.html>
21. **REST API Tutorial.** *What is rest?* [online] [cit. 08.04.2024]. Dostupné na internete: <https://restfulapi.net/>
22. **AUTH0.COM.** *JSON web tokens introduction.* In: *JSON Web Token Introduction* [online] [cit. 22.04.2024]. Dostupné na internete: <https://jwt.io/introduction>
23. **GitHub.** *Build software better, together.* [online] [cit. 08.03.2024]. Dostupné na internete: <https://github.com/about>
24. **TOMAN, S.** 2021. *"Web Application for Data Import from XLSX into a Relational Database."* [Bachelor's Thesis]. Masarykova univerzita, Brno, Česká Republika. [cit. 18.04.2024]
25. **Responsive Web Design Media Queries.** *Responsive web design - media queries.* [online] [cit. 26.04.2024]. Dostupné na internete: https://www.w3schools.com/css/css_rwd_mediaqueries.asp