

Zadanie 4

Ubytovací systém

Obsah

1. Výber technológií.....	3
Backend	3
1.1. Java 17	3
1.2. Spring Boot framework 3.1.4.....	3
1.3 JPA Repozitár 3.1.4	3
1.4 PotgreSQL 42.6.0	3
Frontend	3
1.5 Javascript	3
1.6 React.js	3
2. Implementácia	4
2.1 Registrácia	4
2.2 Prihlásenie	5
2.2 Prihlásenie	6
2.3 Šifrovanie a dešifrovanie faktúry/potvrdenia.....	8
2.4 Autorizácia dopytov.....	8
2.5 SSL a HTTPS.....	8
3. Ukážka funkčnosti aplikácie:.....	9
3.1 Backend	9
3.2 Frontend	14
3.2.1 Registrácia používateľa	14
3.2.2. Prihlásenie používateľa.....	14
3.2.3 Vyhľadanie ubytovania	15
3.2.4 Detail ubytovania.....	16
3.2.5 Rezervácia ubytovania	17
3.2.6 Zoznam rezervácií.....	18
3.2.7 Faktúra / Potvrdenie rezervácie.....	19
3.2.8 Odhlásenie používateľa	19
3.2.9 Docker.....	20
3.2.10 Spustenie aplikácie bez Dockeru	20
4 Popis zdrojového kódu.....	20
.....	21
Zdroje:.....	21
Ďalšie zdroje použité pri implementácii:	21

1. Výber technológií

Backend

1.1. Java 17

Java je viacúčelový, objektovo orientovaný programovací jazyk ktorý je rýchly, bezpečný a spoľahlivý (1). V našom projekte využívame jazyk Java pre implementáciu biznis logiky a manipuláciu s dátami. Rozhodli sme sa preň z dôvodu predchádzajúcich skúseností s týmto jazykom.

1.2. Spring Boot framework 3.1.4

Spring Boot je framework pre vývoj mikroslužieb a webových aplikácií v jazyku Java. Poskytuje jednoduchý spôsob na vytvorenie samostatnej, produkčne pripravenej aplikácie. V našom projekte sme sa rozhodli pre Spring Boot kvôli našim skúsenostiam s jeho používaním a rýchlosti vývoja (2).

1.3 JPA Repozitár 3.1.4

V repozitári, ktorý interaguje s databázou rozširujeme rozhranie JPA Repository, ktoré nám umožňuje používať množstvo preddefinovaných metód. V zadaní sme zatiaľ využili možnosť získania záznamu z databázy na základe primárneho kľúča.

1.4 PostgreSQL 42.6.0

PostgreSQL je open-source relačný databázový systém, ktorý sme zvolili pre náš projekt na ukladanie a manipuláciu s dátami. Táto voľba vychádza z jeho spoľahlivosti a výkonu. Využívame ho ako backendovú databázu pre náš JPA repozitár, čo nám umožňuje efektívne spravovať dátové operácie v našich mikroslužbách implementovaných pomocou Spring Boot frameworku (3).

Frontend

1.5 Javascript

JavaScript, je skriptovací programovací jazyk. Jazyk je používaný najmä pri tvorbe webových stránok(5). Rozhodli sme sa preň z dôvodu predchádzajúcich skúseností s týmto jazykom

1.6 React.js

React je open-source frontendová knižnica JavaScriptu na vytváranie používateľských rozhraní alebo komponentov používateľského rozhrania(6). React.js sme si vybrali kvôli jeho popularite, jednoduchosti a z dôvodu predchádzajúcich skúseností s vývojom webovej aplikácie pomocou tohto nástroja.

2. Implementácia

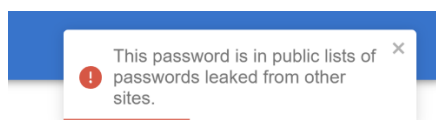
V aplikácii je implementovaná registrácia, prihlásenie aj prenos šifrovaných údajov a následné dešifrovanie. Po spustení kontajnera v backendovej časti aplikácie pomocou príkazu **docker-compose up --build** a následnom spustení frontendu je možné tieto funkcionality pretestovať, ukážky funkcionality sú aj na screenshotoch v ďalšej časti dokumentácie.

2.1 Registrácia

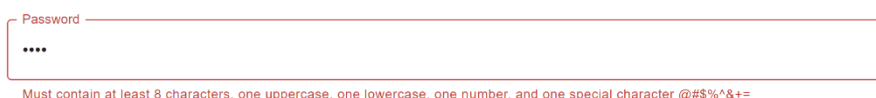
Kontrola zložitosti hesla

Kontrola zložitosti hesla prebieha vo frontendovej aj backendovej časti aplikácie pomocou regexov a kontroly dĺžky hesla. V našej aplikácii sú minimálne požiadavky na dĺžku hesla 8 znakov a heslo musí obsahovať minimálne 1 malé písmeno, 1 veľké písmeno, jednu číslicu a jeden špeciálny znak zo zoznamu: @#\$%^&+=. Pokiaľ heslo nespĺňa požiadavky, používateľ je na to upozornený a registrácia nie je úspešná.

Heslo je v frontendovej časti kontrolované pomocou yup validačnej schémy. Taktiež je heslo kontrolované či sa nenachádza v zozname ukradnutých hesiel pomocou vytvorenia dopytu na url adresu <https://api.pwnedpasswords.com>. V prípade, že sa naše heslo nachádza v zozname ukradnutých hesiel je používateľ upozornený.



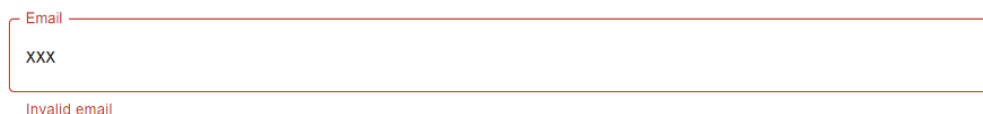
Obrázok 1 Upozornenie na uniknuté heslo



Obrázok 2 Validácia hesla v nesprávnom formáte

Kontrola platnosti emailu

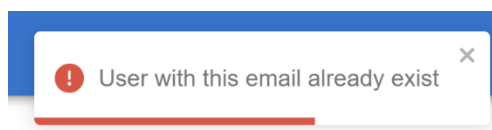
Pri pokuse o registráciu prebehne vo frontendovej aj backendovej časti aplikácie kontrola, či má zadaný email platný tvar. Kontrola sa vykonáva pomocou regexov a očakáva sa, že email je v tvare name@domainsite.domain. Pokiaľ nie je email v platnom tvare, používateľ je na to upozornený a registrácia nie je úspešná. Email je v frontendovej časti kontrolovaný pomocou yup validačnej schémy.



Obrázok 3 Validácia emailu v nesprávnom formáte

Kontrola existujúceho používateľa

V backendovej časti aplikácie prebehne pred registrovaním kontrola, či sa v databáze nenachádza používateľ s rovnakým emailom. Pokiaľ je už email použitý, registrácia je neúspešná.



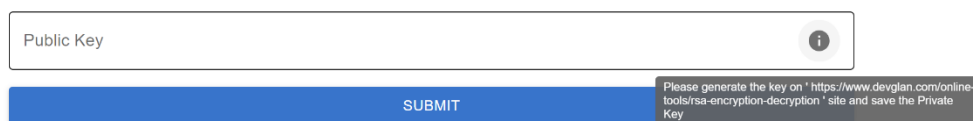
Obrázok 4 Upozornenie na existujúceho používateľa

Ukladanie hesla

Na zabezpečenie bezpečného uloženia hesla je použitý **BCryptPasswordEncoder**, pridaný do Spring Security. Bcrypt používa silný hashovací algoritmus spolu s pridaním „soli“ pre zvýšenie bezpečnosti hesiel. Tento algoritmus vytvára hash pomocou iteratívneho procesu založeného na Blowfish kryptografickom algoritme (4). Heslo je teda pred uložením do databázy bezpečne zahashované a je k nemu pridaný aj náhodne generovaný reťazec, teda „sol“.

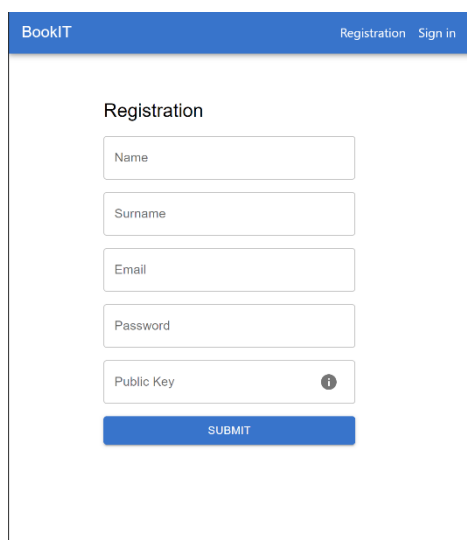
Nahratie verejného kľúča

Pri registrácii je používateľ povinný vyplniť aj textové pole s verejným kľúčom. Používateľ je vyzvaný, aby si RSA kľúčový pár vygeneroval, napríklad na stránke <https://www.devglan.com/online-tools/rsa-encryption-decryption>, odporúča sa veľkosť 2048 bitov. Verejný kľúč zadá pri registrácii a privátny kľúč je nutné uložiť si v jeho zariadení pre použitie pri dešifrovaní, na čo je taktiež upozornený pri registrácii. Verejný kľúč sa ukladá do databázy ako jeden z údajov novo registrovaného používateľa.

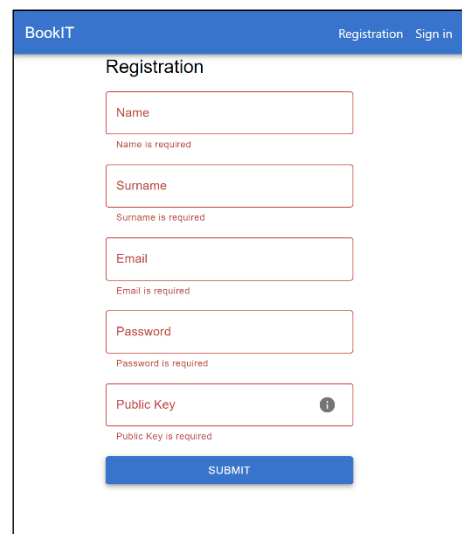


Obrázok 5 Nahratie verejného kľúča

Všetky polia v registračnom formulári sú validované na Frontendovej aj Backendovej strane. Je kontrolované, či sa hodnota v poli nachádza, dĺžka prípadne formát vstupu od používateľa. Používateľ je informovaný o úspešnej registrácii či prípadných problémoch, ktoré nastanú.



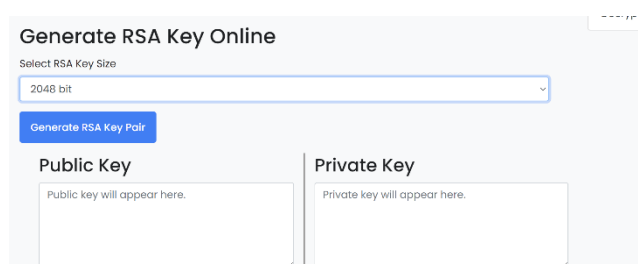
Obrázok 6 Registračný formulár



Obrázok 7 Nevalidný formulár

Public a private key musia byť vygenerované na stránke:

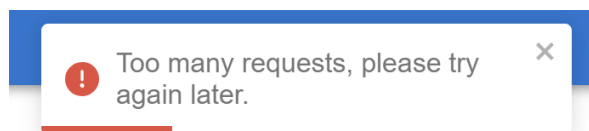
<https://www.devglan.com/online-tools/rsa-encryption-decryption>



2.2 Prihlásenie

Časový rozostup pri prihlasovaní

Na ochranu pred brute-force útokmi je vo frontendovej časti aplikácie implementovaná funkcia, ktorá ráta a kontroluje počet pokusov o odoslanie dopytu. Táto funkcia dovoľuje používateľovi len 3 pokusy o prihlásenie v priebehu 5 sekúnd. Pred odoslaním dopytu pomocou funkcie `fetch()`, je najskôr skontrolované či je posledný odoslaný dopyt v časovom intervale (5 sekúnd). V prípade, že sa snažíme vykonať dopyt, no posledný dopyt bol vykonaný skôr ako 5 sekúnd dozadu, je porovnaný počet dopytov v posledných 5 sekundách a maximálny povolený počet. Pokiaľ je pokusov viac, užívateľ je upozornený vyskakovacím oknom a prihlásenie neprebehne.



Obrázok 8 Upozornenie na priveľa dopytov

JWT

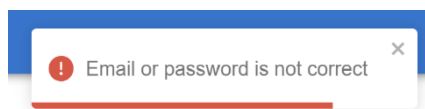
Po úspešnom prihlásení do aplikácie sa v backendovej časti vygeneruje JWT (JSON Web Token). Z bezpečnostných dôvodov je jeho platnosť nastavená na 30 minút, teda aj platnosť prihlásenia používateľa je iba 30 minút. V JWT je bezpečne uložené ID prihláseného používateľa, čo umožňuje bezpečnú autorizáciu pri ďalších dopytoch. Pri dopyte na endpoint, pre ktorý je nutná autorizácia s neplatným alebo vymysleným JWT je takáto aktivita v backendovej časti odchytená, používateľ je na to upozornený a jeho prístup nie je povolený. Vo frontendovej časti je tento token uložený do Cookies. Tento token sa posielá spolu s dopytmi na backendovú časť aplikácie v hlavičke dopytu. Po uplynutí 30 minút je z cookies odstránený a používateľ je presmerovaný na stránku prihlásenia.

Name	Value	D...	Path	Expires / Max-Age
auth	eyJhbGciOiJIUzI1NiJ9.eyJzdWiiOiJxliwiZXhwIjoxNzAwMzAxO...	lo...	/	2023-11-18T10:04:25.000Z

Obrázok 9 Uloženie tokenu v cookies

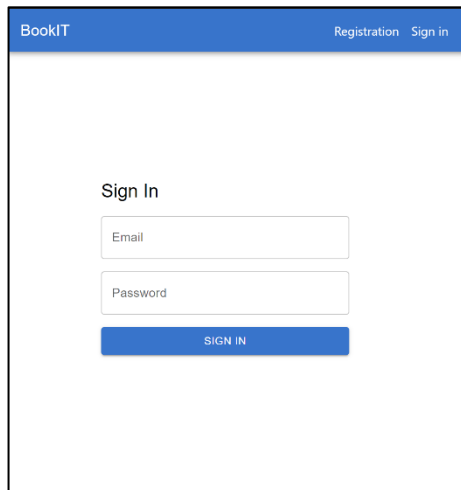
Validácie

Pri prihlásení je taktiež implementovaná validácia na vstupoch od používateľa. Taktiež je v backendovej časti implementovaná validácia na kontrolu správnosti emailu a hesla. V prípade, že jedno z týchto polí nie je správne, používateľ je upozornený.



Obrázok 10 Upozornenie na nesprávnosť mena/hesla

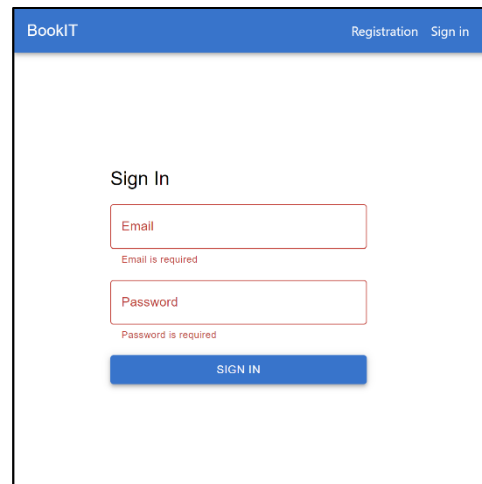
Po úspešnom prihlásení je používateľovi prispôsobený navigačný panel, v ktorom sa následne nachádza tlačidlo na odhlásenie, vyhľadanie ubytovania a zobrazenie rezervácií. V zozname rezervácií je možné získať zašifrovanú faktúru, ktorú je možné odšifrovať pomocou private key.



The screenshot shows the 'Sign In' form on the BookIT website. The header is blue with 'BookIT' on the left and 'Registration Sign in' on the right. The form is centered and contains the following elements:

- A heading 'Sign In'.
- An 'Email' input field.
- A 'Password' input field.
- A blue 'SIGN IN' button.

Obrázok 11 Prihlasovací formulár



The screenshot shows the 'Sign In' form with validation errors. The header is the same as in the previous image. The form elements are:

- A heading 'Sign In'.
- An 'Email' input field with a red border and the text 'Email is required' below it.
- A 'Password' input field with a red border and the text 'Password is required' below it.
- A blue 'SIGN IN' button.

Obrázok 12 Validácia prihlasovacieho formulára

2.3 Šifrovanie a dešifrovanie faktúry/potvrdenia

Uloženie a zobrazenie faktúry je zabezpečené šifrovaním.

PDF je vytvorené v backendovej časti aplikácie, následne je zašifrované pomocou AES. Kľúč využitý na šifrovanie faktúry je následne zašifrovaný pomocou RSA (public kľúčom uloženým v databáze) a tieto údaje sú zaslané na frontendovú časť.

Zašifrované pdf (pomocou AES) spolu so zašifrovaným kľúčom (pomocou RSA) je zaslané z backendovej časti aplikácie na frontendovú časť. Frontendová časť najskôr odšifruje kľúč na AES pomocou RSA odšifrovania privátnym kľúčom. Následne sa pdf odšifruje pomocou získaného odšifrovaného AES kľúču. Odšifrované pdf je následne uložené a je možné jeho nahliadnutie používateľovi.

RSA odšifrovanie bolo vykonané pomocou node-forge knižnice, AES odšifrovanie bolo vykonané pomocou knižnice crypto. Mód bol nastavený na ECB a padding na Pkcs7.

2.4 Autorizácia dopytov

Každý dopyt vykonaný po prihlásení používateľa je autorizovaný pomocou autorizačnej hlavičky zasielanej s dopytom. Táto autorizačná hlavička obsahuje JWT token, ktorý je uložený v Cookies aplikácie. V prípade modifikácie alebo odstránenia tohto tokenu, je používateľovi odmietnutý prístup a je následne presmerovaný na stránku s prihlásením. Tento token má exspiráciu 30 minút. Po tomto čase je používateľ presmerovaný na stránku s prihlásením.

2.5 SSL a HTTPS

V našej aplikácii sme implementovali podporu pre SSL (Secure Socket Layer), aby sme zabezpečili šifrovaný prenos dát medzi klientom a serverom. Vďaka požiadavkám na testovanie v lokálnom prostredí vývoja sme momentálne zakomentovali túto funkciu. Tento prístup umožňuje efektívne a bezproblémové testovanie aplikácie bez obmedzení spojených s vytváraním self-signed certifikátov alebo výskytom varovaní o neoveriteľných certifikátoch.

3. Ukážka funkčnosti aplikácie:

3.1 Backend

Registrácia:

POST localhost:8443/api/register

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded **raw** binary GraphQL JSON

```
1 {
2   "name": "Adam",
3   "surname": "Testovič",
4   "email": "adamd.ssss.com",
5   "password": "hesielkooA5%",
6   "publicKey": "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAs8eUNIdK3Yb1wM/Sc+YVoE7B9gIuN6
7   +nSPvQoFjWZa41KN6ReJyBLfQ4SudkeAzoShg5QMInGHuSomd0UyTTDPjAhwzTUR/VsKSB5g7x4eP2nJzVRoj350j/Pj/iQFKj+g7b0nyIIomXrolS12S1TapRISUSq
  +v7BoPS81ipHzWCgnhEuXiUpmjzv46fgSiwducLkj2/iZX8vxFXTXauJnfc3ZmRu2oXP8519dV2ed+tQEKwM/13eA8/h5X2IQ/LHHHaBkbFgY+IRU61+Q/5J5r16/
  oV0sSRG7Smw26o0BSmIwfguTMGX1K4A7emRbzZ7dV47TV0m+r53hhrcQIDAQAB"
```

Body Cookies (1) Headers (14) Test Results Status: 201 Created Time: 240 ms Size: 509 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "name": "Adam",
4   "surname": "Testovič",
5   "email": "adamd.ssss.com",
6   "jwt": null
7 }
```

Registrácia s nedostatočne zložitým heslom:

POST localhost:8443/api/register

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded **raw** binary GraphQL JSON

```
1 {
2   "name": "Adam",
3   "surname": "Testovič",
4   "email": "adam@mail.com",
5   "password": "hesielko",
6   "publicKey": "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAs8eUNIdK3Yb1wM/Sc+YVoE7B9gIuN6
7   +nSPvQoFjWZa41KN6ReJyBLfQ4SudkeAzoShg5QMInGHuSomd0UyTTDPjAhwzTUR/VsKSB5g7x4eP2nJzVRoj350j/Pj/iQFKj+g7b0nyIIomXrolS12S1TapRISUSq
  +v7BoPS81ipHzWCgnhEuXiUpmjzv46fgSiwducLkj2/iZX8vxFXTXauJnfc3ZmRu2oXP8519dV2ed+tQEKwM/13eA8/h5X2IQ/LHHHaBkbFgY+IRU61+Q/5J5r16/
  oV0sSRG7Smw26o0BSmIwfguTMGX1K4A7emRbzZ7dV47TV0m+r53hhrcQIDAQAB"
```

Body Cookies (1) Headers (13) Test Results Status: 400 Bad Request Time: 18 ms Size: 599 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "Password should contain at least: 1 uppercase, 1 lowercase, 1 digit and one of special characters: @$%&+=",
3   "status": 400,
4   "details": "com.example.upbz4.exception.RegistrationException"
5 }
```

Prihlásenie:

The screenshot shows a REST client interface for a POST request to `localhost:8443/api/login`. The request body is in JSON format, containing `email` and `password` fields. The response status is 200 OK, and the response body is a JSON object containing `id`, `name`, `surname`, `email`, and a long `jwt` token.

```
POST localhost:8443/api/login

{"email": "testovic@mail.com", "password": "securepassword4="}

{"id": 1, "name": "User", "surname": "Testovic", "email": "testovic@mail.com", "jwt": "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiIxIiwiaXNjaXhwaXoxNzAwMzg4NzIzZjQ.VzUzpnRPVPYHA54EZG9SpPfgFptsZu3Tf1MI_ZQa2Tw"}
```

Prihlásenie s nesprávnym emailom/heslom:

The screenshot shows a REST client interface for a POST request to `localhost:8443/api/login`. The request body is in JSON format, containing `email` and `password` fields. The response status is 400 Bad Request, and the response body is a JSON object containing `message`, `status`, and `details` fields.

```
POST localhost:8443/api/login

{"email": "adamd.s@sss.com", "password": "incorect"}

{"message": "Email or password is not correct", "status": 400, "details": "com.example.upbz4.exception.LoginException"}
```

Všetky endpointy aplikácie (okrem registrácie a prihlásenia) sú zabezpečené a vyžadujú platný JWT v autorizačnej hlavičke dopytu.

V opačnom prípade:

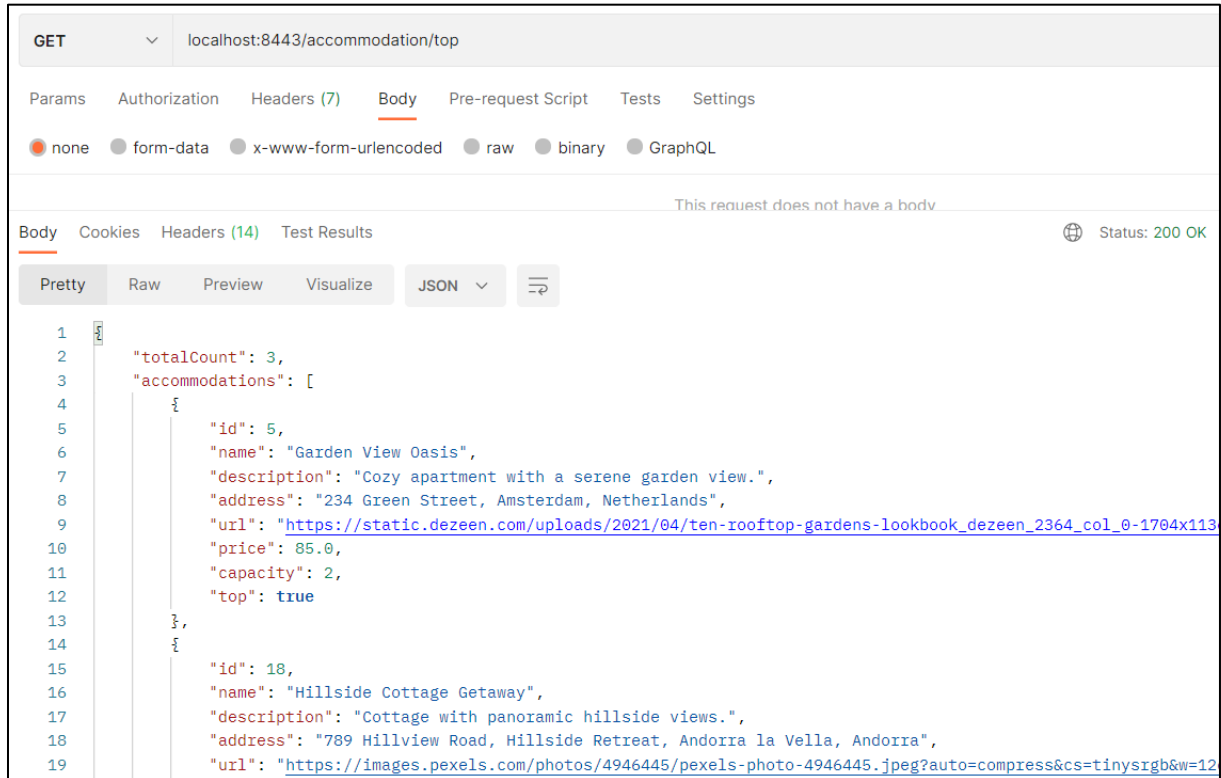
The screenshot shows a REST client interface for a POST request to `localhost:8443/reservation/new`. The request body is in JSON format, containing `accommodationId`, `checkInDate`, `checkOutDate`, and `guestsCount` fields. The response status is 401, and the response body is a JSON object containing `message`, `status`, and `details` fields.

```
POST localhost:8443/reservation/new

{"accommodationId": 1, "checkInDate": "2024-01-01", "checkOutDate": "2024-01-02", "guestsCount": 3}

{"message": "Invalid token", "status": 401, "details": "com.example.upbz5.exception.LoginException"}
```

Zobrazenie top ubytovaní (použitá na úvodnej strane)



GET localhost:8443/accommodation/top

Params Authorization Headers (7) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

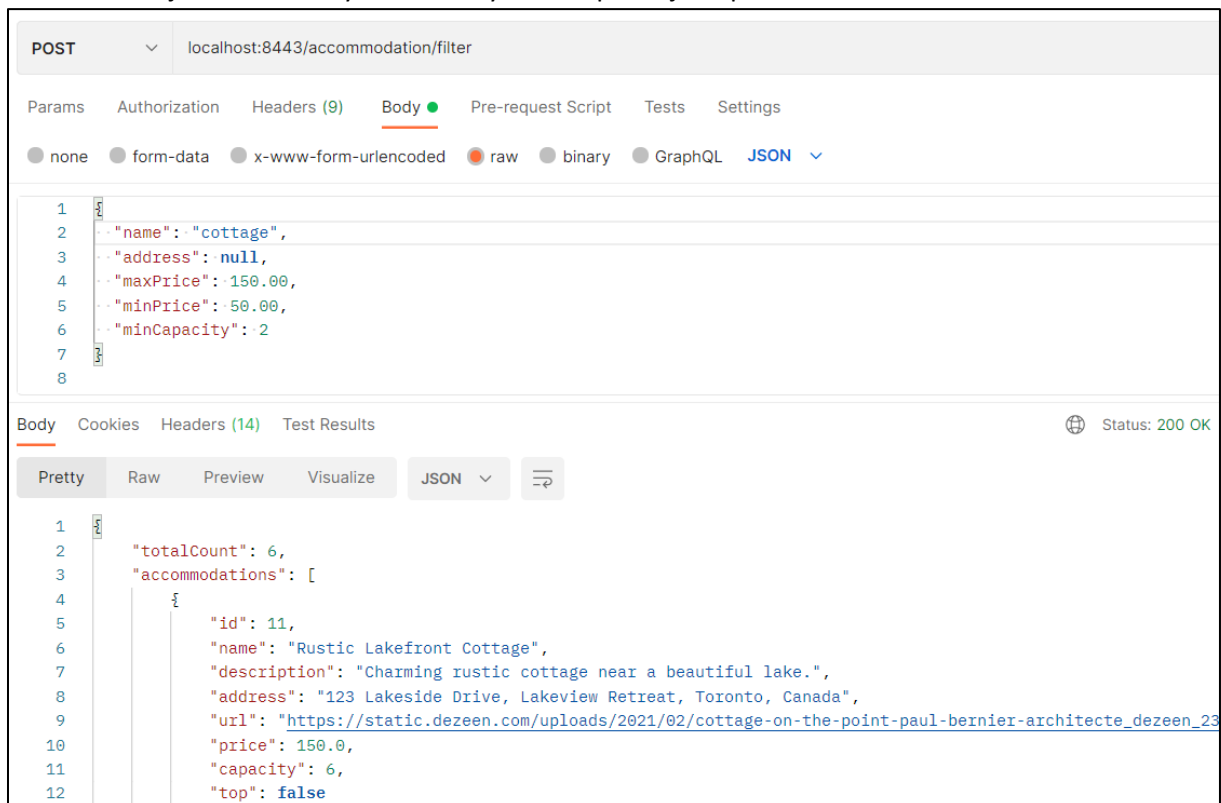
This request does not have a body

Body Cookies Headers (14) Test Results Status: 200 OK

```
1 {
2   "totalCount": 3,
3   "accommodations": [
4     {
5       "id": 5,
6       "name": "Garden View Oasis",
7       "description": "Cozy apartment with a serene garden view.",
8       "address": "234 Green Street, Amsterdam, Netherlands",
9       "url": "https://static.dezeen.com/uploads/2021/04/ten-rooftop-gardens-lookbook_dezeen_2364_col_0-1704x113",
10      "price": 85.0,
11      "capacity": 2,
12      "top": true
13    },
14    {
15      "id": 18,
16      "name": "Hillside Cottage Getaway",
17      "description": "Cottage with panoramic hillside views.",
18      "address": "789 Hillview Road, Hillside Retreat, Andorra la Vella, Andorra",
19      "url": "https://images.pexels.com/photos/4946445/pexels-photo-4946445.jpeg?auto=compress&cs=tinysrgb&w=1280&h=720&dpr=2"
20    }
21  ]
22 }
```

Vyhľadanie ubytovania pomocou filtra

Používateľovi je umožnené vyfiltrovať ubytovanie podľa jeho požiadaviek:



POST localhost:8443/accommodation/filter

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "name": "cottage",
3   "address": null,
4   "maxPrice": 150.00,
5   "minPrice": 50.00,
6   "minCapacity": 2
7 }
8
```

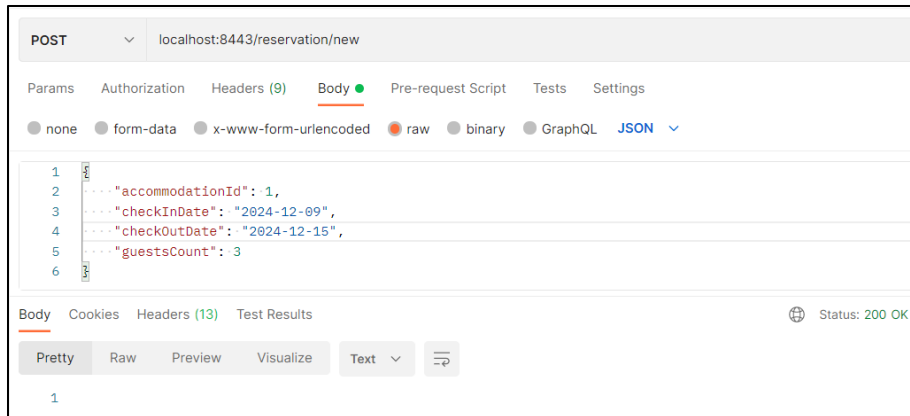
Body Cookies Headers (14) Test Results Status: 200 OK

```
1 {
2   "totalCount": 6,
3   "accommodations": [
4     {
5       "id": 11,
6       "name": "Rustic Lakefront Cottage",
7       "description": "Charming rustic cottage near a beautiful lake.",
8       "address": "123 Lakeside Drive, Lakeview Retreat, Toronto, Canada",
9       "url": "https://static.dezeen.com/uploads/2021/02/cottage-on-the-point-paul-bernier-architecte_dezeen_23",
10      "price": 150.0,
11      "capacity": 6,
12      "top": false
13    }
14  ]
15 }
```

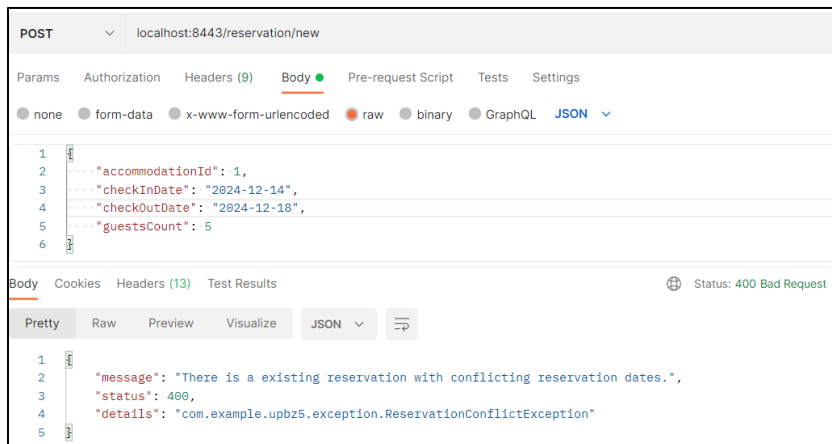
V tomto prípade jeho požiadavkám zodpovedalo 6 ubytovaní.

Rezervácia

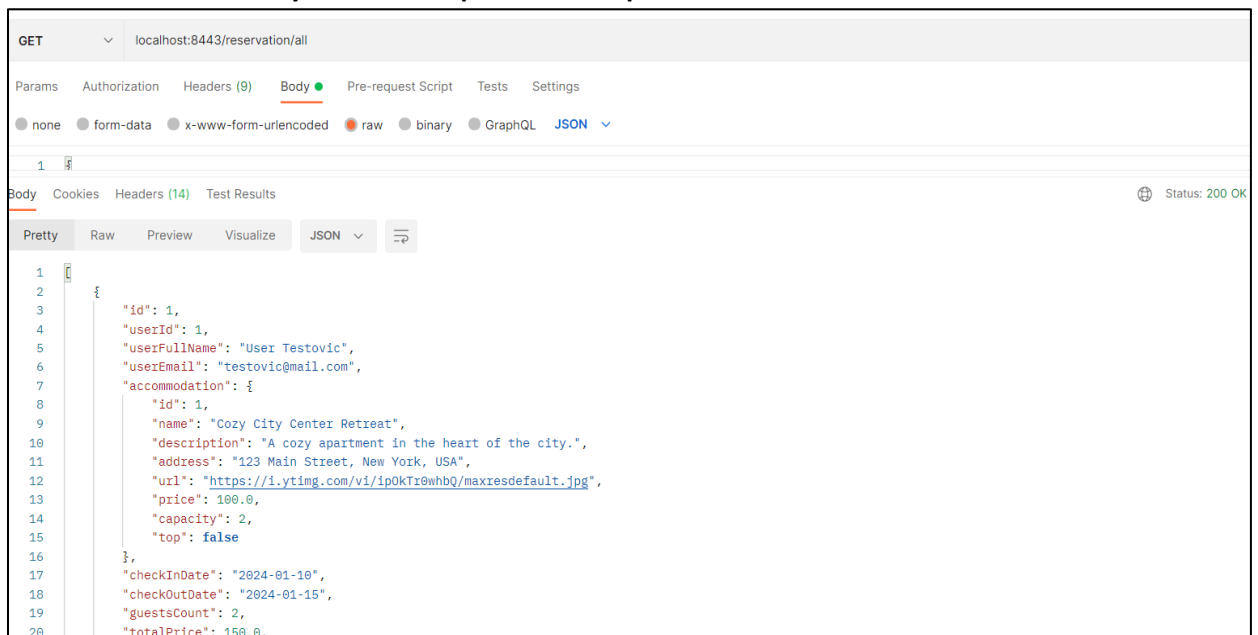
Úspešná rezervácia:



Snaha o rezerváciu, ktorá by sa prekrývala s inou, už vytvorenou rezerváciou daného ubytovania: (ošetrené aj na FE, v kalendári nie je možné „vyklikáť“ tieto dátumy)



Získanie zoznamu všetkých rezervácií prihláseného používateľa:



Získanie zakódovanej faktúry a zakódovaného AES kľúča:

GET

localhost:8443/reservation/detail/1

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

5

Body

Cookies

Headers (14)

Test Results

Status: 200 OK

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

json

"pdf": "mixVzc1d+U8sw7AHNLNxxNV/v/1hvfK1dKbnxozGkoabs+Quz1apXC5tGZW/peueOcBvXuyZK279ushyd93augYKQbZ0A3Nd+mf08jZIk19TAd1ohueD81PzEPdB/iBPaALd//uQJX4smbRCs4A36oq;

"encryptedAes": "P294Qef3F7ISMNap1E5NLsmcTMTH5oMS8mUzrp8vIndIS3KfUUtYGuW+dowLQunWzU7Z+hHS4u5hzaXZwQH1E9v90S9AKf8gYbPGz6wmmvFh9S18v8y7GVMacYu4As1yp8a9r1F3D1aR1Yf

Všetky

Pre zjednodušenie testovania je v databáze vytvorený testovací user, ktorého je možné využiť.

E-mail	Password	Private key
testovic@mail.com	securepassword4=	MIIeVQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQCc+2aULldq3r7BfD+ZAZo3TxuCFXMMkDGeUUTrhGhYtU+iuW97MYxZtFhzmAYVA0Kq1blvPSOFScOjd9l3trU5nhErBORmnVv30fhWxb6UyiqSQoAl5YkRzFXV6B08WsLrIMgbT9z36R4kd/s+2iC5CSCI7wMSZ1IEqQrnWdAZcW5eI+Z2lAsgV5nlsCkMZPcwhtSQq8Vacy37/bicfZ3pO+XGItAc+RZM5V/Sg3+GN4DIV+C7GArmaQfgT/3lIHAdUkpVySUDgaPvLIBFnzovvvoCjhmiedXHF80F2gJuP+es5lrC1qfBHAc8qxbmg/c81Lujz25Q7d2//br3/vAgMBAAECggEAda/3YisIRpPvj6BA8HR9ITcGC8A2EFK7EnoJB+iHbTUz4YzmOvWxGqStWNs7Lt5AhZafumv0/HjLCAu3VR1P2D1gUKugRZJX9HXVewnCypc3kZfG2Flbg5sqjUTheCz4tpu8kja29CE9AUxou+DE7d6ZY3F8eDnonvkZ40phjHzcAa4btZILUC7HoAAxtPREU0LIRAJ0l27YdQP30Q33V9vWZSEzBMehI1AwJ1yG17oZVvRU5dz88xAVnf5gq7QXeX8PJL61vn5WI7bCv9mewXvPyLaLidaRzlfNCVoPrtaixZlfdFMq5RNaPtZCaopclAyLMD7qELdOP902heAQKBgQC6/z5pRQj5+zGkQ0QwAR1ZjaiD0XPSH5a+BBGGNNAQfebUrfXeYfNkLf25kGPF2E8zE437AlvzsAfDCHiu1ZelSm3sq/WHB3LzW9KhMW9W+ymtnYb32zLAl0/hgeDXU9GV1V+KYdJocXX66NM9zA2/+fw8c606ocDSv9faE7wKBgQCzUky8WLVQE6URrmfqt855bwR3/m8/dxlqQ44LYhCvBO0Ek+P9IG6zRMmGHXsluCZ6BE6q63jpFww1PWfZyqfBHi3R/3HDATIScwmw/83Cd2cecb2WO7IBANjCUVAhkVPZSHnt5kTSXOaaW9RP0gu27CWlIJUdaK8gpV2gS1AQKBgQCPIKkyjGZbg8rBQcNFQvUlsM9E5eJ5qF8G3bdNuMqwdM1jaCtSZ/ScAPLuBduUHH8OvAn6f6hE02j12qfG5rFwlgN1JsFtMuRnJJCZTe88kk5ZADtyphZtZEffQUUq5Uw/UdTA4uqyUTUBTPkbluCWtUz5Ci7J9gy0H9jV03y/QKBgDHPfjxPtAYLfVbaXhjt+GAZK7edcQ/DJsOgwmt9ET9c8e4sms+Jv56CxCxCMjQo219XpPartimjQlvjshA8UEZ46qeD+v9dQvnMtD9Lg4QGB0oozaBXXAUuS5GbrX0ZcNc86nXJIT9Q32rdp2PdUOZ848Pdh9cPSe3Z/uoEBAoGAcOAPdvT9qxkbDpwZnTgR1tFoqzR1HYCaJj6nvn1J8wPe+/2tjVc3ZgO05rmNcN+elxcvwOCb4c8HY1gqOUyyzoiWm5vn0WfTbOpcD+/4B0UZ7mUQjlaPGrV5ddFJBwQqigubGbbxSzXJKSidiI9qglaeRLBNXkZYNT2Y4/gMA=

V prípade potreby je privátny kľúč priložený v textovom súbore.

3.2 Frontend

3.2.1 Registrácia používateľa

BookIT

RegistrationSign in

Registration

Name

Surname

Email

Password

Public Key

SUBMIT

3.2.2. Prihlásenie používateľa

BookIT

RegistrationSign in

Sign In

Email

testovic@mail.com

Password

SIGN IN

3.2.3 Vyhľadanie ubytovania

Úspešne prihlásený užívateľ je automaticky presmerovaný na stránku s vyhľadaním ubytovania. Na stránke je mu ešte pred vyhľadaním poskytnutý náhľad top ubytovaní.


BookIT

ReservationsSearchSIGN OUT

Q Search accommodation

Capacity

SEARCH




Garden View Oasis

234 Green Street, Amsterdam, Netherlands
Cozy apartment with a serene garden view.

MORE

★

85 €




Hillside Cottage Getaway

789 Hillview Road, Hillside Retreat, Andorra la Vella, Andorra
Cottage with panoramic hillside views.

MORE

★

140 €



Mansion by the Lake

567 Lakeside Road, Lake Como, Italy
Grand mansion with lakeside location and expansive grounds.

MORE

★

450 €

Ak chce vyhľadať ubytovanie je potrebné do poľa „search accommodation“ zadať názov. Prípadne je možné doplniť taktiež kapacitu, ktorú by malo ubytovanie spĺňať. Následne po stlačení tlačidla „search“ sú mu zobrazené ubytovania, ktoré zodpovedajú jeho požiadavkám.


BookIT

ReservationsSearchSIGN OUT

Q cottage

3

SEARCH




Rustic Lakefront Cottage

123 Lakeside Drive, Lakeview Retreat, Toronto, Canada
Charming rustic cottage near a beautiful lake.

MORE

150 €




Cozy Countryside Cottage

456 Meadow Lane, Countryside Haven, Bibury, UK
Cozy cottage nestled in the countryside.

MORE

130 €



Quaint Cottage by the River

234 Riverside Lane, Riverside Bliss, Montalino, Italy
Quaint cottage overlooking a river.

MORE


110 €

3.2.4 Detail ubytovania

Na prehľade ubytovania klikne na tlačidlo „More“ pri jednotlivom ubytovaní. Je presmerovaný na stránku s detailom ubytovania, kde si môže zarezervovať ubytovanie.

BookIT

ReservationsSearchSIGN OUT



Rustic Lakefront Cottage

Charming rustic cottage near a beautiful lake.

Address: 123 Main Street, City, Country

Price: 150 € per night

From

To

Number of Guests

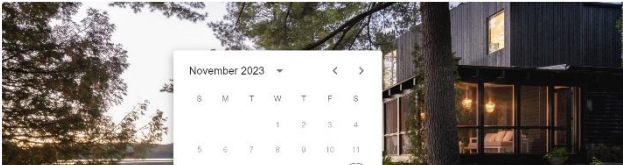
Price

0

€

BOOK NOW

Vyberie dátum od, dátum do a počet hostí.



Rustic Lakefront Cottage

Charming rustic cottage near a beautiful lake.

Address: 123 Main Street, City, Country

Price: 150 € per night

From

11/29/2023

To

MM/DD/YYYY


Number of Guests

Price

0

€

BOOK NOW



Rustic Lakefront Cottage

Charming rustic cottage near a beautiful lake.

Address: 123 Main Street, City, Country

Price: 150 € per night

From

11/29/2023

To

11/30/2023

Number of Guests

1 guest

2 guests

3 guests

4 guests

5 guests


6 guests

Price

0

€

BOOK NOW



Rustic Lakefront Cottage

Charming rustic cottage near a beautiful lake.

Address: 123 Main Street, City, Country

Price: 150 € per night

From

11/29/2023

To

11/30/2023

3 guests

Price

150

€

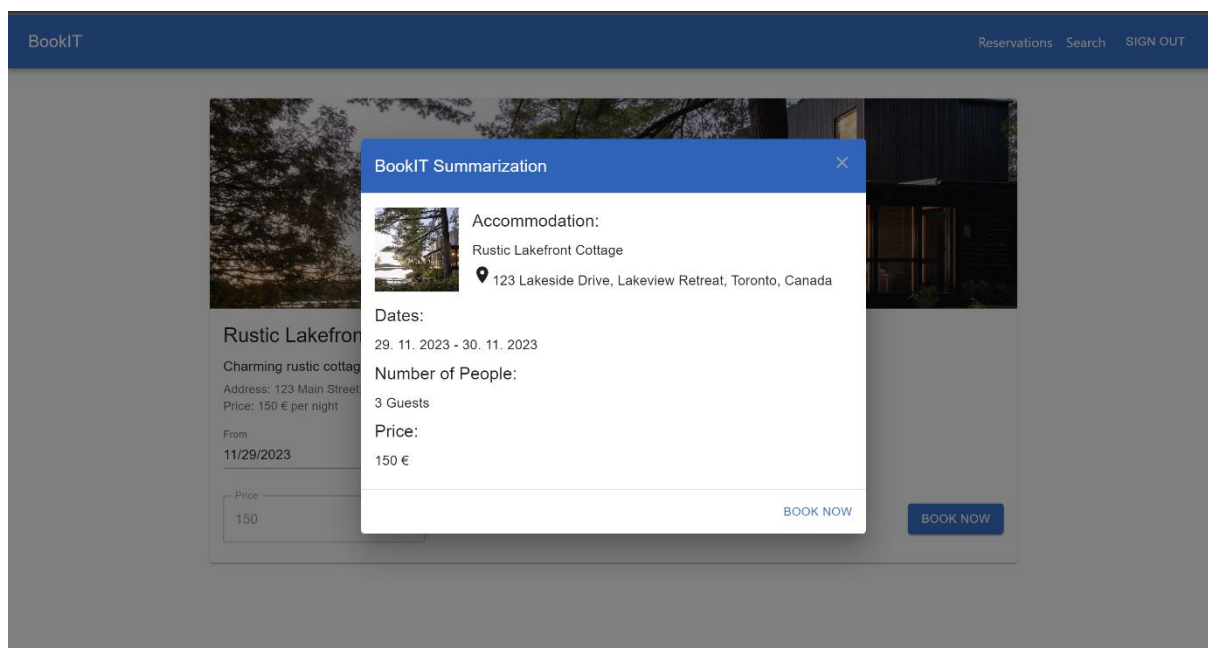
BOOK NOW

Následne je mu automaticky vyrátaná cena podľa počtu nocí a je mu umožnené kliknúť na tlačidlo „Book now“. V prípade, že by chcel zarezervovať dátum, ktorý už je rezervovaný alebo do minulosti, prípadne dátum od by bol neskôr ako dátum do, nie je mu to umožnené. Manuálne vpísanie ľubovoľného dátumu taktiež nie je umožnené.

November 2023

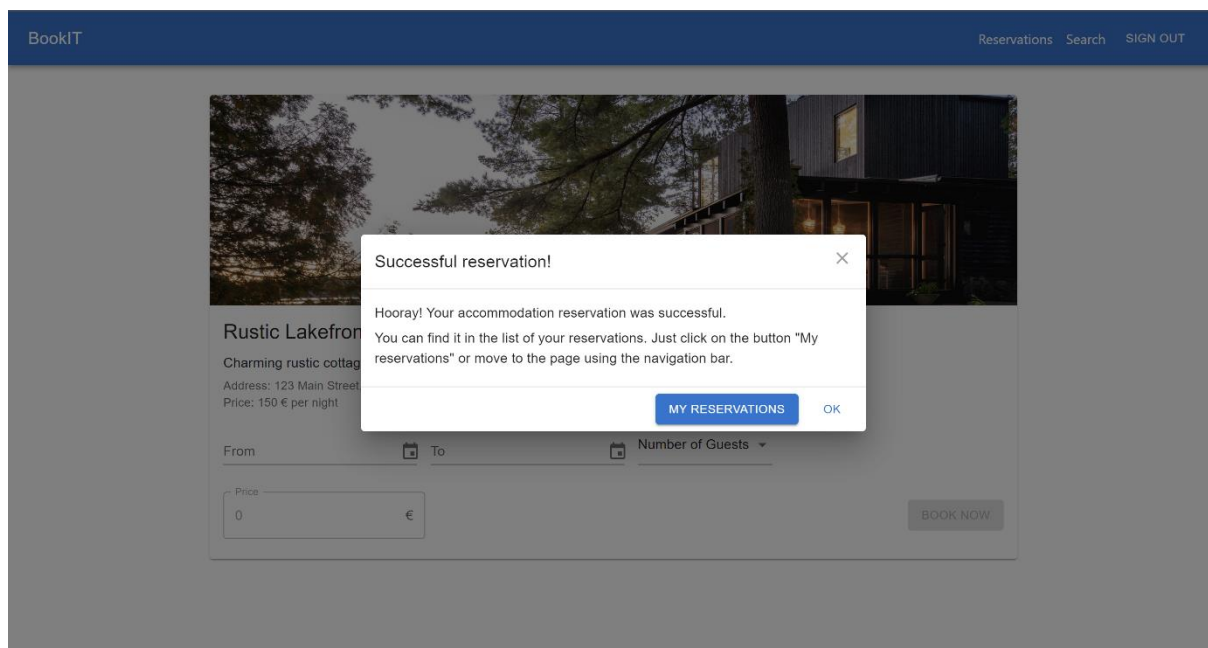
S	M	T	W	T	F	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

3.2.5 Rezervácia ubytovania







Po kliknutí na tlačidlo „book now“ je mu poskytnutý náhľad sumarizácie rezervácie. Ak je so sumarizáciou spokojný, môže kliknúť na tlačidlo „BOOK NOW“ a jeho rezervácia je odoslaná. Ak nesúhlasí, môže okno jednoducho zavrieť pomocou „X“ v pravom hornom rohu a svoju objednávku rezervácie upraviť.

V prípade že objednávku rezervácie potvrdí. Je tým patrične oboznámený.







3.2.6 Zoznam rezervácií

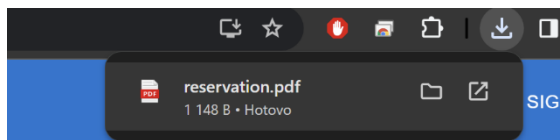
Na zoznam svojich rezervácií sa používateľ dostane pomocou navigačného panelu. Alebo v prípade vykonania rezervácie po kliknutí na tlačidlo „My reservations“ na modálnom okne potvrdenia.

BookIT						
Reservations Search SIGN OUT						
My Reservations						
Accommodation	Address	Guests	Check In	Check Out	Total Price	
Cozy City Center Retreat	123 Main Street, New York, USA	2	2024-01-10	2024-01-15	150 €	
Cozy City Center Retreat	123 Main Street, New York, USA	1	2024-02-05	2024-02-10	120 €	
Cozy City Center Retreat	123 Main Street, New York, USA	3	2024-03-20	2024-03-25	180 €	
Rustic Lakefront Cottage	123 Lakeside Drive, Lakeview Retreat, Toronto, Canada	3	2023-11-29	2023-11-30	150 €	

Na zozname rezervácií používateľ vidí svoje vykonané rezervácie jednotlivých ubytovaní. V prípade, že chce zobraziť potvrdenie a faktúru pre jednotlivé rezervácie, klikne na ikonu súboru v poslednom stĺpci tabuľky.

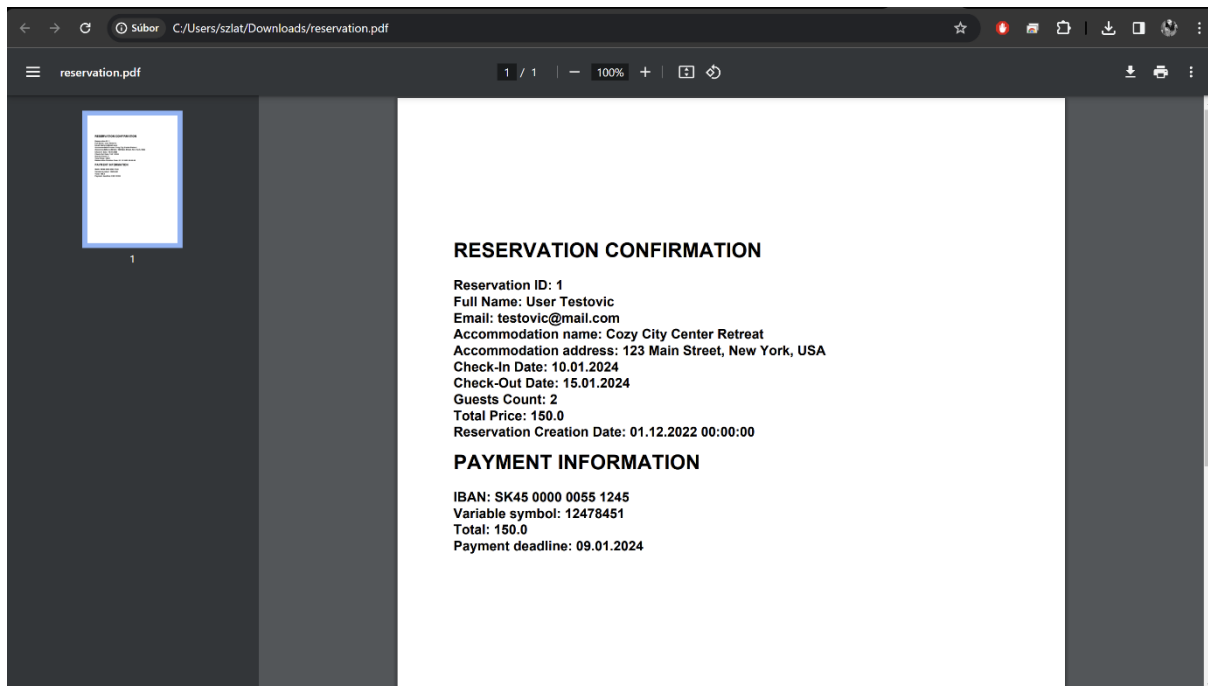
BookIT						
Reservations Search SIGN OUT						
My Reservations						
Accommodation	Address	Guests	Check In	Check Out	Total Price	
Cozy City Center Retreat	123 Lakeside Drive, Lakeview Retreat, Toronto, Canada	3	2023-11-29	2023-11-30	150 €	
Cozy City Center Retreat					120 €	
Cozy City Center Retreat					180 €	
Rustic Lakefront Cottage					150 €	

Následne je mu otvorené modálne okno, do ktorého zadá svoj privátny kľúč. Po zadaní správneho privátneho kľúču, je faktúra úspešne uložená.



3.2.7 Faktúra / Potvrdenie rezervácie

Uloženú faktúru nájde v stiahnutých súboroch. Je vo formáte pdf. Jej náhľad vyzerá takto:



3.2.8 Odhlásenie používateľa

Používateľovi je taktiež umožnené odhlásenie pomocou tlačidla „Sign Out“ na navigačnom paneli. Ak klikne na toto tlačidlo, je automaticky presmerovaný na stránku s prihlásením a jeho token je odstránený z cookies.

3.2.9 Docker

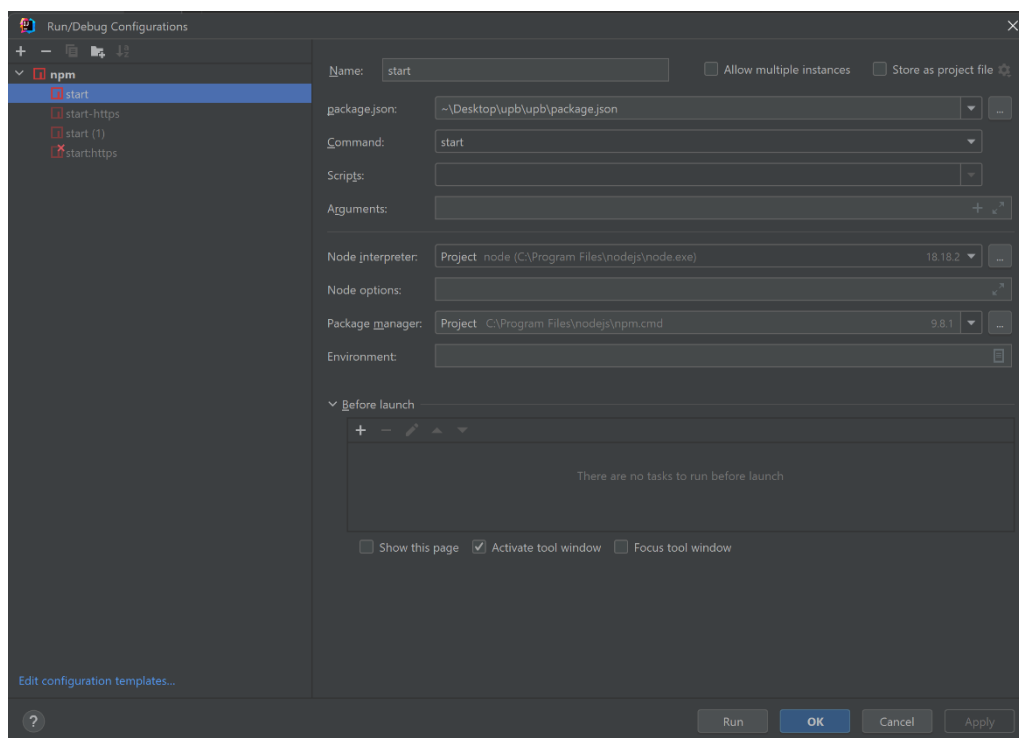
Obe časti webovej aplikácie sú zdockerizované. Obsahujú DockerFile a docker-compose.yaml. Teda je možné spustiť aplikácie pomocou dockeru.

Frontendová časť aplikácie sa nachádza na porte 3000 (<http://localhost:3000/>). Backendová časť aplikácie sa nachádza na porte 8443 (<http://localhost:8443>)

Na správne fungovanie dockeru je potrebné mať nainštalovanú a spustenú aplikáciu Docker Desktop (<https://www.docker.com/products/docker-desktop/>). Obe časti aplikácie je potrebné spustiť pomocou **docker-compose up --build** v termináli v root adresári jednotlivého projektu. (Vytvorenie a spustenie kontajnerov môže chvíľu trvať). V databáze spustenej pomocou dockeru sa nachádzajú dáta obsahujúce jednotlivé ubytovania, testovací používateľ a jeho rezervácie.

3.2.10 Spustenie aplikácie bez Dockeru

Frontendovú časť aplikácie je možné spustiť aj bez dockeru. Je potrebné mať nainštalovaný Node.js verziu 18.18.2 (<https://nodejs.org/en>). Je potrebné v termináli v root adresári projektu spustiť „npm install“. Po nainštalovaní knižníc je potrebné spustiť príkaz „npm start“, následne by mala byť spustená frontendová časť aplikácie. Aplikáciu ja následne možné stopnúť pomocou stlačenia kláves CTRL+C v tom istom termináli a potvrdení. Prípadne je možné nastaviť konfiguráciu vo vývojovom prostredí nasledovne .



4 Popis zdrojového kódu

Zdrojový kód sa nachádza v dvoch súboroch, je rozdelený na frontendovú časť a backendovú časť. Tieto časti je potrebné otvoriť zvlášť a taktiež zvlášť spúšťať. Obe časti sú docker-izovaná, BE obsahuje taktiež docker-izovanú databázu. Frontendová časť sa nachádza v adresári upb. Backendová časť sa nachádza v adresári UPB-z4-BE.

Zdroje:

- (1) https://www.w3schools.com/java/java_intro.asp
- (2) <https://www.ibm.com/topics/java-spring-boot>
- (3) <https://aws.amazon.com/rds/postgresql/what-is-postgresql/>
- (4) <https://www.baeldung.com/spring-security-registration-password-encoding-bcrypt>
- (5) <https://sk.wikipedia.org/wiki/JavaScript>
- (6) [https://sk.wikipedia.org/wiki/React_\(webov%C3%BD_framework\)](https://sk.wikipedia.org/wiki/React_(webov%C3%BD_framework))

Ďalšie zdroje použité pri implementácii:

- <https://www.baeldung.com/java-rsa>
- <https://www.devglan.com/java8/rsa-encryption-decryption-java>
- <https://docs.spring.io/spring-security/site/docs/current/api/org.springframework.security.crypto.bcrypt/BCryptPasswordEncoder.html>
- <https://www.baeldung.com/java-email-validation-regex>
- <https://www.geeksforgeeks.org/how-to-validate-a-password-using-regular-expressions-in-java/>
- <https://formik.org/docs/guides/validation>
- <https://mui.com/material-ui/getting-started/>
- <https://www.tabnine.com/code/javascript/functions/node-forge/decode64>
- <https://snyk.io/advisor/npm-package/node-forge/functions/node-forge.pki.privateKeyFromPem>
- <https://fkhadra.github.io/react-toastify/introduction>
-