

# HOTEL ANIPIA

Hotel Anipia je webová aplikace, která je určená pro zákazníky, kteří chtějí vytvořit rezervaci pokoje pro svého domácího mazlíčka. Aplikace umožňuje:

- Odesílání kontaktního formuláře, který se ukládá do databáze H2 a současně odesílá e-mail přes Mailtrap.
- Registraci nového uživatele, přihlášení do jeho profilu a odhlášení.
- Vytvoření samotné rezervaci.

## Použité technologie a nástroje

Vrstva	Technologie / nástroje
Frontend	HTML, CSS, JavaScript, AJAX (fetch)
Backend	Java 17, Spring Boot, Spring Web, Spring Data JPA, Spring Mail
Databáze	H2
Testování e-mailů	Mailtrap (simulace SMTP)
Správa závislostí	Maven
Testování API	curl a nástroje typu Postman
Testování aplikace	API testování
Profilování / optimalizace	Chrome DevTools, Lighthouse

## Struktura projektu

```
ANIPIA/
├── data/
│   ├── anipia.mv.db
│   └── anipia.trace.db
├── Documentation/
│   ├── Diagrams/
│   ├── UI/
│   ├── Business story.pdf
│   └── Product page.png
├── Spring/anipia/
│   ├── src/
│   │   ├── main/java/com/anipia/
│   │   │   ├── controller/
│   │   │   ├── model/
│   │   │   ├── repository/
│   │   │   ├── service/
│   │   │   └── AnipiaApplication.java
│   │   └── resources/
│   │       ├── static/
│   │       │   ├── CSS/
│   │       │   ├── Photos/
│   │       │   └── *.html, *.js
│   │       ├── templates/
│   │       └── application.properties
└── README.md
```

---

## Architektura komponent a zobrazení

Aplikace obsahuje logicky odlišné zobrazení (stránky):

Úvodní stránka, About, Gallery, Prices, Login, My profile atd.

Navigace je řešena přes odkazy mezi stránkami a AJAX pro interaktivní odesílání dat.

---

## Správa stavu a práce s daty

- **Frontend** – stav uživatele (vyplněný formulář, přihlášený profil) se drží v paměti prohlížeče během aktuální relace.
- **Backend** – stav je uchováván v databázi H2. Každý kontakt nebo uživatel má unikátní ID.
- **Dataflow** – uživatel vyplní formulář, frontend provede validaci přes AJAX POST, backend přijme odpověď ve formátu JSON, service uloží do DB, odešle e-mail, frontend uživateli zobrazí potvrzení.

Data z API jsou přenášena jako JSON, odpovědi obsahují status a zprávu o výsledku akce.

---

## API endpointy

Endpoint	Metoda	Popis
/contact/submit	POST	Odeslání kontaktního formuláře
/contact/ping	GET	Kontrola běhu serveru
/contact/by-email	GET	Vyhledání zpráv podle e-mailu
/contact/deleteByEmail	DELETE	Smazání zpráv podle e-mailu
/export/pdf	GET	Export všech zpráv do PDF
/api/zakaznici/signup	POST	Registrace nového uživatele
/api/zakaznici/login	POST	Přihlášení uživatele

---

## Testování

**API testování** – curl příkazy pro kontrolu jednotlivých endpointů.

Příklad testovacího curl příkazu:

```
curl -X POST http://localhost:8080/contact/submit \  
-H "Content-Type: application/json" \  
-d '{"name":"Uzivatel","email":"uzivatel@test","message":"Test"}'
```