

Software Design Document

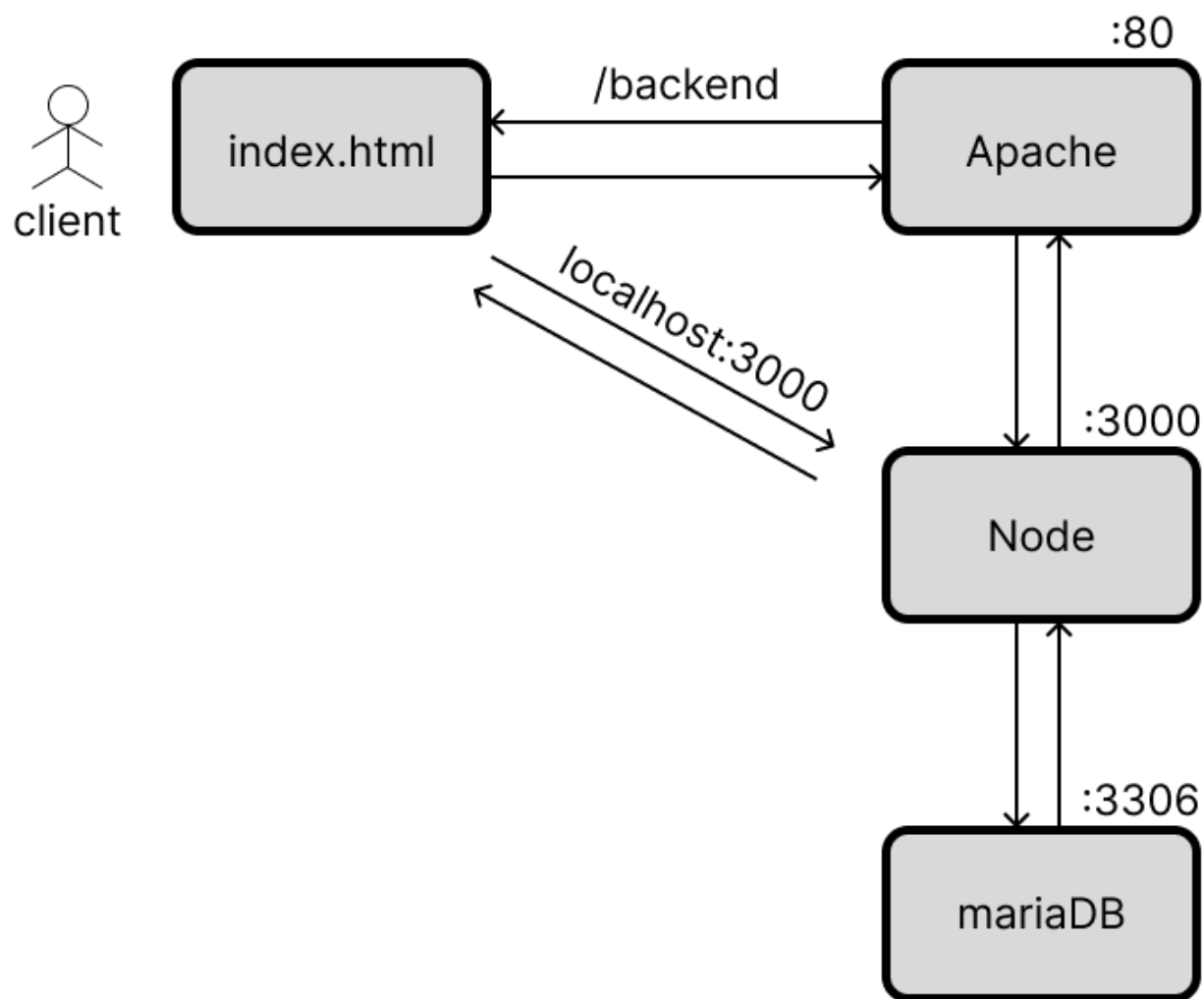
- Projekt: Alpine Apartments
- Verze: 1.1
- Autor: Radek Fajčík
- Datum: 19. 5 .2025

Tento dokument popisuje softwarový návrh webové aplikace Alpine Apartments. Aplikace umožňuje uživatelům rezervovat ubytování a restauraci, kontaktovat provozovatele a zároveň poskytuje administrační rozhraní pro správu dat.

Obsah

1.	Architektura aplikace	2
2.	Popis API	3
3.	ERD pro databázi	5
4.	Funkce backendu	6
5.	Ukázka UI.....	8

1. Architektura aplikace



Aplikace je rozdělena do tří hlavních vrstev:

- **Frontend**: HTML, CSS, JavaScript, běží ve webovém prohlížeči
- **Backend**: Node.js + Express.js, běží na serveru localhost:3000
- **Databáze**: MariaDB/MySQL, běží lokálně a používá databázi „mydb“

Vrstvy spolu komunikují pomocí REST API a data jsou přenášena ve formátu JSON.

2. Popis API

- Typ API: REST (Representational State Transfer)
- Formát dat: JSON (JavaScript Object Notation)
- Základní URL: <http://localhost:300/backend>

GET /ping

- Ověřuje, zda je backend dostupný
- Nepřistupuje k databázi

POST /backend/contact

- Ukládá zprávy z kontaktního formuláře do databáze
- Tabulka messages

POST /backend/bookNow

- Ukládá zprávy z rezervační formuláře pro ubytování do databáze
- Tabulky reservations, customers a rooms

POST /backend/restaurantReservation

- Ukládá zprávy z rezervační formuláře pro restauraci do databáze
- Tabulky reservations, customers a tables

GET /admin/messages

- Načítá zprávy z databáze do „admin panelu“
- Tabulka messages

GET /admin/accommodation

- Načítá informace o rezervaci ubytování z databáze do „admin panelu“
- Tabulky reservations, customers a rooms

GET /admin/restaurant

- Načítá informace o rezervaci stolu z databáze do „admin panelu“
- Tabulky reservations, customers a tables

GET /admin/accommodation/:id

- Načítá informace o rezervaci konkrétního pokoje z databáze do „admin panelu“
- Tabulky reservations, customers a rooms

GET /admin/restaurant/:id

- Načítá informace o rezervaci konkrétního stolu z databáze do „admin panelu“
- Tabulky reservations, customers a tables

PUT /admin/messages/:id/answered

- Mění zprávu v databázi jako označenou
- Tabulka messages

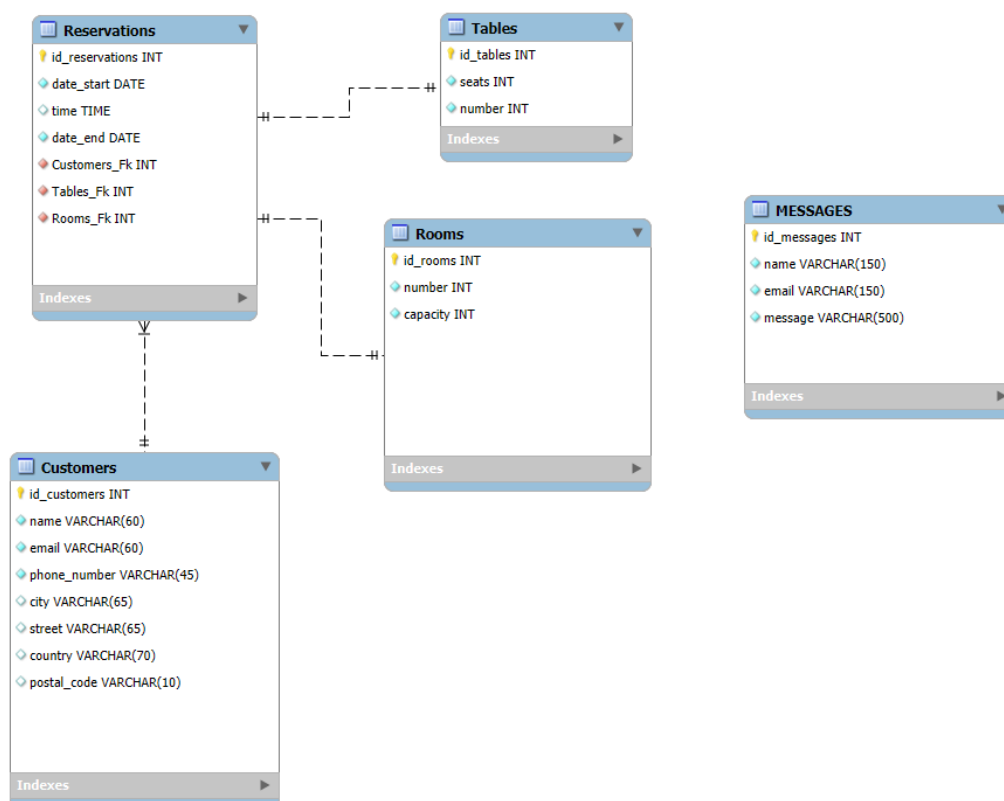
DELETE /admin/messages/:id

- Maže zprávu z databáze
- Tabulka messages

DELETE /admin/reservations/:id

- Maže rezervaci z databáze
- Tabulky rooms, tables a reservations

3. ERD pro databázi



Tento Entity-Relationship Diagram znázorňuje hlavní datové entity aplikace a jejich vztahy.

- **Customers** – uchovává informace o uživatelích, kteří provedli rezervaci. Obsahuje údaje jako jméno, e-mail, telefon a adresa.
- **Reservations** – centrální tabulka obsahující všechny rezervace. Každá rezervace může být buď pro pokoj, nebo stůl. Obsahuje cizí klíče na tabulky Customers, Rooms a Tables, a dále údaje o termínu (date_start, date_end, time).
- **Rooms** – uchovává informace o jednotlivých pokojích, včetně čísla pokoje a jeho kapacity.
- **Tables** – obsahuje údaje o restauraci – čísla stolů a počet míst u každého.
- **Messages** – ukládá zprávy z kontaktního formuláře. Každá zpráva obsahuje jméno, e-mail a textovou zprávu. (Nepřipojeno na jiné tabulky – samostatný formulář.)

Vztahy:

- Mezi Reservations a Customers je vazba N:1 – každý zákazník může mít více rezervací.
- Mezi Reservations a Rooms nebo Tables je vazba N:1 – každá rezervace odkazuje na konkrétní pokoj nebo stůl (v závislosti na typu rezervace).

Rezervace ubytování využívá pole date_start a date_end, zatímco rezervace stolu využívá date_start a time.

4. Funkce backendu

Backend aplikace Alpine Apartments je implementován pomocí Node.js a Express.js. Slouží jako prostředník mezi frontendem (uživatelské rozhraní) a databází (MySQL/MariaDB). Je zodpovědný za zpracování HTTP požadavků, přístup k datům a správu rezervací i zpráv.

1. Zpracování REST API požadavků

- Backend běží na adrese localhost:3000 a poskytuje endpointy přes /backend

2. Ukládání zpráv z kontaktního formuláře

- Data jsou ukládány do tabulky „Messages“
- Endpoint: POST /backend/contact
- Po úspěšném vložení vrací JSON odpověď o úspěchu

3. Rezervace ubytování

- Backend najde volný pokoj podle kapacity a termínu, vytvoří zákazníka v tabulce „Customers“ a poté uloží rezervaci do tabulky „Reservations“
- Endpoint: POST /backend/bookNow

4. Rezervace stolu

- Backend najde volný stůl podle počtu osob a termínu, vytvoří zákazníka v tabulce „Customers“ a poté uloží rezervaci do tabulky „Reservations“
- Endpoint: POST /backend/restaurantReservation

5. Získání dat pro personál

- Backend poskytuje pro administraci data z databáze těchto endpointů:
 - Endpoint: GET /backend/admin/messages (Získá všechny zprávy z kontaktního formuláře)
 - Endpoint: GET /backend/admin/restaurant (Získá všechny rezervace stolů)
 - Endpoint: GET /backend/admin/accommodation (Získá všechny rezervace ubytování)
 - Endpoint: GET /backend/admin/accommodation/:id (Získá rezervace konkrétního pokoje)
 - Endpoint: GET /backend/admin/restaurant/:id (Získá rezervace konkrétního stolu)
 - Endpoint: PUT /backend/admin/messages/:id/answered (Označí zprávu jako vyřízenou)
 - Endpoint: DELETE /backend/admin/messages/:id (Smaže zprávu z databáze)
 - Endpoint: DELETE /backend/admin/reservations/:id (Smaže rezervaci z databáze)

6. Testovací endpoint

- Testovací endpoint: GET /backend/ping
- Vrací zprávu "Backend is alive" pro ověření funkčnosti backendu

5. Ukázka UI

Na obrázku je ukázka návrhu pro index.html (Home), UI návrhy pro ostatní stránky jsou umístěny ve složce s názvem „UI“.

V návrzích jsou jednotlivé prvky označeny zeleně nebo červeně:

- Červeně jsou označeny prvky, s kterými je možné nějakým způsobem interagovat (tlačítka, formuláře...)
- Zeleně jsou označeny statické prvky

