



TRACKING MAPPING DATA

LEVEL UP PROJECT

Author: Uroš Petrović

Projekat – Pregled i Struktura

Projekat je podeljen u pet glavne celine:

- Baza i ORM
- Shipments stranica
- Upload stranica
- Shipment ID stranica
- Auth i Autorizacija

Svaka celina je opisana u nastavku.

1. Baza i ORM

Za bazu podataka koristim **MySQL** hostovan lokalno putem **Laragon-a** zbog njegove jednostavnosti i preglednosti.

Bazu sam povezo sa **IntelliJ Ultimate Database** alatom unutar IDE-a, što mi omogućava laku obradu podataka.

Za ORM koristim **Prisma** zbog njene odlične kompatibilnosti sa **TypeScript-om** i moje prethodne upoznatosti sa njenim funkcionalnostima. Prisma je omogućila efikasan i brz rad sa bazom.

2. Shipments stranica

Shipments stranica koristi **Client-Side Rendering** (CSR) iz nekoliko razloga:

- Parsiranje podataka na sledeću stranicu (Shipment ID stranica) za detaljan prikaz.
- Implementacija paginacije radi sortiranja podataka po stranicama.
- Omogućavanje pretrage po svim parametrima.

Zbog navedenih funkcionalnosti, **client-side rendering** je optimalan izbor zbog brzine izvršavanja i funkcionalnosti stranice.

3. Upload stranica

Upload proces je podeljen na **Client-Side** (file: `page.tsx`) i **Server-Side** (file: `route.ts`):

Client-Side

Na klijentskoj strani kreiram glavni izgled stranice i omogućavam učitavanje fajla.

Server-Side

Na serverskoj strani implementiram sledeće funkcionalnosti:

1. Obrada fajlova:

- Proveravam podatke specifične za različite provajdere (Hellmann, DHL, Logwin).
- Za **Hellmann**, fajl mora odgovarati sledećim kriterijumima:
 - Podaci se nalaze u sheet-u broj **2**.
 - Počinju iz **3. reda** i **2. kolone**.
- Naziv fajla se obrađuje tako da ostaje samo relevantan deo naziva (npr. „hellmann“) radi daljeg mapiranja.

2. Dinamički routing:

- Pozivam `carrierMappings.ts` iz **lib** foldera koji određuje odgovarajući mapping route na osnovu naziva fajla:

```
javascript
Copy code
const mapping = carrierMappings[carrierType];
```

3. Obrada podataka:

- **Obrada stringova:**
 - Uklanjam znakove za novi red (`\r\n`) i zamenjujem ih praznim mestom (`" "`).
 - Za null ili nepostojeće vrednosti vraća se `"Not Defined"`.
- **Obrada datuma:**
 - Proveravam validnost datuma. Ako je vrednost prazna ili nepravilnog tipa, vraća se `null`.
 - Ako je vrednost **„Self-Delivery“** (specifično za Hellmann), kolona za datum dobija vrednost **„1900-01-01 00:00:00“**.
 - Ako postoji datum bez vremena, podrazumevano se dodaje vreme **„00:00:00“**.
 - Konvertujem Excel datume iz numeričkog formata u **ISO-8601** format koji zahteva MySQL.

4. Mapiranje podataka:

- Prvo mapiram vrednost `carrier_type` iz `mapping.typesofcarriers` da bih zabeležio iz kog fajla dolazi red.
- Na kraju šaljem obrađene podatke u bazu koristeći Prismu.

4. Shipment ID stranica

Ova stranica prikazuje podatke za određeni **shipment ID**, koji baza automatski generiše putem **auto-increment** funkcionalnosti. Samim tim, ne postoji mogućnost duplikata po ID-u.

Funkcionalnosti

1. Formatiranje datuma:

- Ako je datum loš ili nepostojeći, vraća se `"Not Defined"`.
 - Ako je datum **„1900-01-01“**, prikazuje se vrednost **„Self-Delivery“**.
 - Ako su sati, minuti i sekunde jednaki nuli, prikazuje se samo datum.
 - Ako postoji kompletan datum, on se formatira u čitljiv oblik: **dani, meseci, godine, sati, minuti**.
-

5. Auth i Autorizacija

Autentifikacija

Autentifikacija je implementirana putem sledećih komponenti:

1. Endpoint za autentifikaciju:

- Lokacija: `src/app/api/auth/[...nextauth]/route.ts`
- Koristi **NextAuth.js** za upravljanje sesijama korisnika uz Prismu.

2. Login i registracija stranice:

- Lokacije:
 - `src/app/auth/login/page.tsx`: Stranica za prijavu korisnika.
 - `src/app/auth/register/page.tsx`: Stranica za registraciju novih korisnika.
- Ove stranice koriste komponente kao što su `AuthCard` i `AuthForm` (`src/components/AuthCard.tsx` i `src/components/AuthForm.tsx`) za prikazivanje obrazaca za unos podataka.

3. Middleware za zaštitu ruta:

- Lokacija: `src/middleware.ts`
- Middleware se koristi za zaštitu određenih ruta, osiguravajući da samo autentifikovani korisnici imaju pristup.

Autorizacija

Autorizacija u ovom projektu određuje da admin može da uploaduje fajlove a običan user nema pristup toj funkcionalnosti.

1. Zaštićene API rute:

- Lokacije:
 - `src/app/api/shipments/route.ts`
 - `src/app/api/shipments/upload/route.ts`
 - API rute koriste autentifikaciju za proveru prava pristupa pre nego što omoguće izvršavanje zahteva.
-