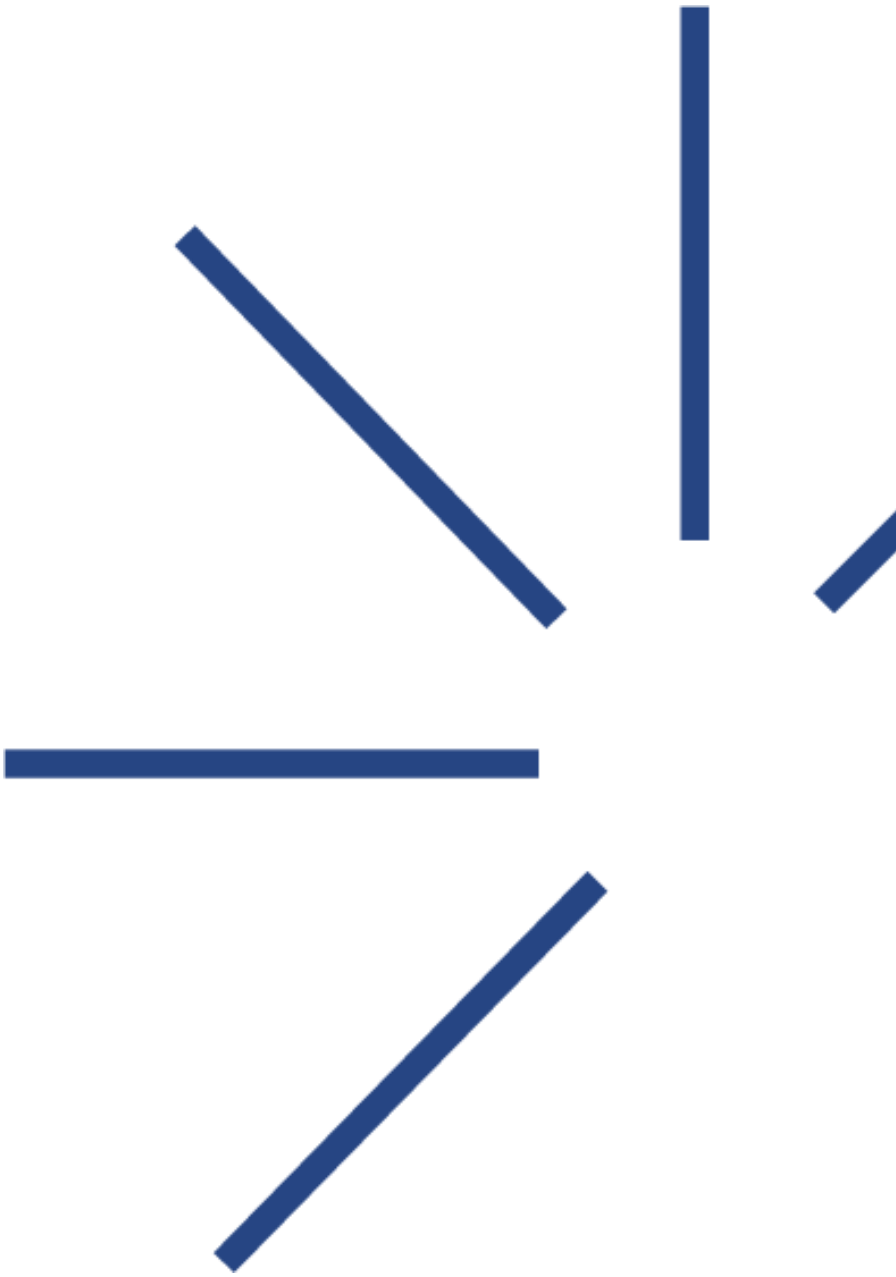
D3 – Metódy pre kontinuálne sťahovanie



Projekt financovaný Európskou úniou Next GenerationEU prostredníctvom Plánu obnovy a odolnosti SR v rámci projektu č. 09I05-03-V02-00049.

Financované Európskou úniou NextGenerationEU
PLÁN OBNOVY
ÚRAD PODPREDSEDU VLÁDY SLOVENSKEJ REPUBLIKY PRE PLÁN OBNOVY A ZNALOSTNÚ EKONOMIKU
VAIA VÝSKUMNÁ A INOVAČNÁ AKTIVITA
VÝSKUMNÁ AGENTÚRA

ÚVOD

Projekt „**Automatizácia analýzy právnych textov na základe strojového učenia**“ (ďalej len „**ALTAML**“) predstavuje kľúčovú iniciatívu zameranú na integráciu inovatívnych prístupov v oblasti spracovania údajov a následnej analýzy, so zameraní na oblasť práva, ktorá zahŕňa aj právo informačných a komunikačných technológií. Cieľom tohto projektu je vyvinúť a overiť účinné metódy automatizovanej analýzy právnych textov pomocou techník strojového učenia. To zahŕňa najmä vývoj nástrojov, ktoré uľahčujú spracovanie a analýzu veľkých objemov údajov vo forme rôznych právnych dokumentov, ako aj extrakciu relevantných informácií (atribútov) z takýchto dokumentov vrátane identifikácie kľúčových pojmov, odkazov na právne predpisy a ďalších atribútov.

Cieľom projektu ALTAML je teda prispieť k efektívnejšiemu prístupu k právnym informáciám a k zrýchleniu právnych procesov, čím sa zabezpečí vyššia miera právnej istoty a zlepší sa dostupnosť právnych textov, výsledkov ich analýzy a relevantných právnych informácií pre odbornú aj širokú verejnosť.

V rámci pracovného balíka KPB1 boli vyvinuté D3 – Metódy pre kontinuálne sťahovanie. Tento dokument predstavuje popis vysoko odolného systému pre kontinuálne sťahovanie súdnych rozhodnutí, implementovanom v jazyku Python s využitím architektúry mikroslužieb a komunikácie prostredníctvom RabbitMQ. Texty rozhodnutí a metadáta sú prevedené do formátu JSON a uložené v databáze MongoDB. Pre rýchle vyhľadávanie sú vybrané atribúty uložené aj v relačnej databáze MySQL. Systém je navrhnutý tak, aby bol odolný voči zlyhaniam externých aj interných komponentov a je možné ho nasadiť pomocou technológií Docker a Docker Compose.

1. ALTAML sťahovač rozhodnutí

Sťahovač rozhodnutí je projekt vytvorený v jazyku Python s cieľom sťahovať naposledy zverejnené súdne rozhodnutia, konvertovať ich do tvaru JSON a uložiť ich do databáz MongoDB a MySQL.

Nasledujúce kapitoly popisujú hlavné funkcionality, systémové požiadavky, konfiguráciu a nasadenie.

Vlastnosti

Základná funkcionalita zahŕňa niekoľko kľúčových procesov:

- **Sťahovanie dát**: Automatické získanie posledných súdnych rozhodnutí z ministerstva spravodlivosti SR.
- **Transformácia dát**: Spracovanie a transformácia stiahnutých metadát a PDF súborov na jednotný JSON formát vhodný pre ďalšie použitie.
- **Uloženie dát**: Uloženie transformovaných dát do MongoDB a MySQL databáz, aby sa optimalizovalo vyhľadávanie a analýza týchto dát.
- **Paralelné spracovanie** – Celý proces je založený na mikroslužbách, ktoré komunikujú cez RabbitMQ message broker. Mikroslužby je jednouché replikovať na viaceré servery a umožniť tak rozloženie záťaže.

Požiadavky

Pred nasadením je potrebné, aby boli k dispozícii systémové komponenty, ktoré zabezpečia stabilné a izolované prostredie. Projekt je plne kontajnerizovaný pomocou Docker a Docker compose a je závislý na databázach MongoDB a MySQL na prezistentné uchovanie.

Vyžadované komponenty:

- Docker and Docker compose
- MongoDB (na uloženie transformovaných dát a metadát)
- MySQL (na uloženie transformovaných metadát)

Nastavenie MongoDB a MySQL na serveri

Inicializácia a konfigurácia databázy sa vykonáva pomocou Docker compose .

Na spustenie oboch databáz na servreri vykonajte na serveri nasledovné kroky:

- Spustite docker-compose-databases.yaml
  - Tento postup predpokladá použitie používateľských premenných prostredia
  - Ak mate premenné prostredia v defaultnom súbore .env spustite príkaz: docker compose up -f docker-compose-databases.yaml -d

- Ak mate premenné prostredia v inom súbore spustite príkaz: docker compose --env-file variables.env -f docker-compose-databases.yaml up -d

2. DataDownloader

DataDownloader komponent riadi extrakciu a uloženie pôvodných metadát do MySQL. Vytvára základ pre vrstvu získavania dát, pričom sa stará o konzistenciu a sledovateľnosť spracovania stiahnutých rozhodnutí.

V MySQL vytvorte tabuľku raw_decision

```
create table raw_decision (
  guid varchar(75),
  datum_vydania date,
  spisova_znacka varchar(20),
  our_id varchar(150),
  ecli varchar(45),
  file_name_original varchar(150),
  file_name_txt varchar(150),
  downloaded_json datetime,
  saved_json_with_fulltext datetime,
  primary key (datum_vydania, guid)
);
```

RabbitMQ

RabbitMQ slúži ako základná komunikačná platforma medzi mikroslužbami, pričom zabezpečuje spoľahlivé doručovanie správ a koordináciu práce. Na inštaláciu a nastavenie RabbitMQ vykonajte nasledovné kroky:

- Na inštaláciu RabbitMQ spustite docker-compose-rabbitmq.yaml
  - Aby ste ho mohli spustiť potrebujete správny .ez súbor ktorý obsahuje plugin pre oneskorené správy
  - Tento súbor pre konkrétnu verziu si môžete stiahnuť tu: https://github.com/rabbitmq/rabbitmq-delayed-message-exchange/releases
  - Zabezpečte, aby ste mali .ez súbor v rovnakom adresári ako dockerfilerabbitmq
    - dockerfilerabbitmq automaticky zapne všetky potrebné pluginy
  - zabezpečte v súbore dockerfilerabbitmq správne meno .ez súboru
- Na vytvorenie všetkých radov, Exchangov a prepojení spustite súbor docker-compose-config.yaml

Nastavenie premenných prostredia

Premenné prostredia definujú systémové parametre, nastavenia pripojenia a konfigurácie pre jednotlivé servery. Ak chcete nakonfigurovať prostredie, postupujte podľa týchto krokov:

- Vytvorte súbor variables.env v rámci koreňového adresára projektu (napr. data-collector/variables.env)
    - Skopírujte doňho obsah súboru variables_example.env a upravte podľa vašich potrieb
    - Ak mate iný názov súboru, potrebujete meniť env súbor pre každý súbor docker-compose
- Premenné prostredia budú iné na každom serveri
- Každý server musí mať inú hodnotu premennej SERVER_NUMBER ktorá označuej každý server
- Spojenie na rabbitmq, mongo a mysql bude pravdepodobne rovnaké na každom serveri, pretože ak všetky služby bežia na jednom mieste, osotatné s ana nich napoja.
- Cesty na uloženie súborov sa môžu líšiť v závislosti o diskových nastavení serverov, ale pravdepodobne cesty k súbprom ostanú rovnaké ak pre nich používame cestrálen úložisko
    - Aj keď nepoužívame centrálne úložisko, môže si každý server uchovávať súbory na rovnakej ceste
- Môžete si nastaviť súbor src\no_consumer_services\last_date.txt, v ktorom je uvedený datum od ktorého sa budú nabudúce ťahať nové rozhodnutia z Ministerstva spravodlivosti

3. Spustenie servera pomocou docker-a

Beh systému na serveri nevyžaduje manuálnu inštaláciu závislostí. O všetko sa postarajú Docker kontajnery.

- Nie je potrebné nič inštalovať, všetky závislosti budú nastavené vo vnútri docker kontajnerov
- Stačí použiť súbor s premennými prostredia, ktorý bol spomínaný vyššie
- Ak nepoužívate defaultný názov súboru .env, je potrebné zmeniť env súbor v každom docker-compose súbore.

Ako spustiť docker-compose súbory

Docker Compose zjednodušuje vykonávanie a orchestráciu služieb.

- docker compose up -d sa použije na spustenie defaultného docker compose súboru
- docker compose -f docker-compose-file up -d sa použije na spustenie špecifického docker-compose súboru

Služby konzumerov

Služby konzumerov vytvárajú základnú kostru postupnosti automatického spracovania dát. Starajú sa o úlohy sťahovania, transformácie, uloženia a monitorovania.

- Na spustenie všetkých služieb dátového toku spustite súbor docker-compose.yaml f
  - Na spustenie iba jednej služby spustite: docker compose up -d - build meno, použite meno z docker-compose súboru
  - Ak služba používa volume, dodajte cestu z premennej prostredia
- Na spustenie monitorovacej služby na monitorovanie zlyhaní spracovania dátového toku spustite súbor docker-compose-email.yaml
  - Kód na odosielanie emailov sa nachádza v email_sender_service.py
  - Na nastavenie e-mailovej adresy, na ktorú majú odchádzať e-maily zmeňte premennú receiver
- Na spustenie služby, ktorá bude konzumovať stream v Rabbit MQ a vkladať rozhodnutia do MongoDB kolekcie spustite docker-compose-stream.yaml
- Na spustenie služby na sťahovanie nových alebo aktualizovaných rozhodnutí spustite súbor docker-compose-list.yaml
  - Ak spúšťate službu na linuxe, dopíšte 0 2 * * Sat .../Start-decision-list.sh do crontab -e súboru, uveďte správnu cestu k súboru Start-decision-list.sh

Administrátor

Administrátorské služby sú určené na správu, zotavenie po zlyhaní a opravu dát.

- Na vytvorenie export zlyhaných rozhodnutí spustite súbor docker-compose-export.yaml. Na nastavenie toho, z ktorej služby sa má export spraviť zmeňte hodnotu premennej prostredia v docker-compose súbore
- Na opätovné vloženie zlyhaných rozhodnutí z MySQL naspäť do RabbitMQ radov spustite docker-compose-reset.yaml, Na nastavenie toho, z ktorej služby sa má export spraviť zmeňte hodnotu premennej prostredia v docker-compose súbore
- Druhý stream konzument používa administrator na vloženie starej verzie rozhodnutia zo streamu. Na spsustenie použite docker-compose-stream-admin-yaml
  - V docker-compose súbore sú dve premenné prostredia ktoré špacifikujú filtrovaciu podmienku a hodnotu offsetu

4. Zdrojové súbory

Zdrojové súbory projektu sú spravované v internom úložisku Gitlab na UPJŠ. Sä dpstupné na adrese: https://gitlab.science.upjs.sk/altaml/decisions-downloader