Implementácia Backendu

Backendová časť systému bola implementovaná ako samostatná webová služba, ktorá zabezpečuje autentifikáciu používateľov, správu dokumentov, komunikáciu s AI modulmi a prácu s databázou. Backend vystupuje ako centrálny integračný prvok celého systému a sprostredkúva komunikáciu medzi aplikačnou vrstvou, dátovým úložiskom a externým AI modulom.

1 Použité technológie

Backend systému bol implementovaný v programovacom jazyku Python s využitím frameworku **FastAPI** [1]. Tento framework sme zvolili najmä pre jeho vysoký výkon, natívnu podporu asynchrónneho spracovania požiadaviek a automatické generovanie OpenAPI dokumentácie.

Na prácu s databázou sme použili ORM nástroj **SQLAlchemy** [2], ktorý umožňuje objektovo-orientovaný prístup k relačnej databáze a zjednodušuje správu dátových modelov. Ako databázový systém je použitá **PostgreSQL** [3], ktorá poskytuje spoľahlivé ukladanie dát a podporu pre komplexnejšie dátové štruktúry.

Autentifikácia používateľov je realizovaná kombináciou **JSON Web Tokens (JWT)** [4] a **OAuth 2.0** [5]. Tento prístup umožňuje podporu klasickej registrácie používateľov, ako aj prihlásenie prostredníctvom externého poskytovateľa identity (Google OAuth).

Komunikácia s AI modulmi prebieha prostredníctvom REST API s využitím knižnice **httpx** [6], ktorá umožňuje asynchrónne odosielanie požiadaviek a spracovanie odpovedí. Backend je pripravený na beh v kontajnerizovanom prostredí **Docker** [7], čo zjednodušuje nasadenie a integráciu jednotlivých komponentov systému.

2 Architektúra backendu

Backend je navrhnutý ako modulárna aplikácia, v ktorej sú jednotlivé časti systému oddelené podľa ich zodpovednosti. Základ aplikácie tvorí hlavný vstupný bod, ktorý inicializuje FastAPI aplikáciu, konfiguruje middleware a registruje jednotlivé API endpointy.

Architektúra backendu zahŕňa samostatné moduly pre:

- správu používateľov a autentifikáciu,
- definíciu dátových modelov a databázových entít,
- validáciu vstupných a výstupných dát pomocou schém,
- bezpečnostné mechanizmy (hashovanie hesiel, overovanie tokenov),
- pomocné nástroje a utility funkcie.

Backend využíva dependency injection mechanizmus frameworku FastAPI, ktorý umožňuje efektívne spravovať databázové spojenia a autentifikačný kontext používateľa. Prístup k databáze je realizovaný prostredníctvom databázovej session, ktorá je automaticky sprístupnená jednotlivým endpointom.

Dôležitou súčasťou architektúry je oddelenie backendu od AI modulov. AI spracovanie dokumentov prebieha v samostatnej službe, ku ktorej backend pristupuje prostredníctvom REST API. Tento prístup umožňuje nezávislý vývoj a škálovanie AI časti systému bez zásahu do backendovej logiky.

3 Autentifikácia a autorizácia

Autentifikácia a autorizácia používateľov sú v systéme implementované ako kľúčové bezpečnostné mechanizmy, ktoré zabezpečujú kontrolovaný prístup k funkcionalitám backendu. Backend podporuje kombináciu klasickej autentifikácie pomocou používateľského mena a hesla a autentifikáciu prostredníctvom externého poskytovateľa identity založenú na štandarde OAuth 2.0.

Pri klasickej registrácii používateľa je heslo pred uložením do databázy hashované pomocou kryptografickej hashovacej funkcie. Po úspešnom prihlásení je používateľovi vygenerovaný JSON Web Token (JWT), ktorý obsahuje identifikátor používateľa a ďalšie nevyhnutné informácie. Tento token je následne využívaný pri autorizácii jednotlivých API požiadaviek.

Overovanie JWT tokenu prebieha pomocou dependency injection mechanizmu frameworku FastAPI. Chránené endpointy vyžadujú platný token, ktorý je kontrolovaný z hľadiska platnosti, integrity a časovej expirácie. Na základe dekódovaného tokenu je identifikovaný aktuálne prihlásený používateľ.

Súčasťou autentifikačného procesu je aj verifikácia e-mailovej adresy používateľa. Po registrácii je používateľovi zaslaný overovací e-mail obsahujúci jednorazový token. Až po úspešnom overení e-mailovej adresy je používateľovi povolený prístup k chráneným funkcionalitám systému.

Autentifikácia prostredníctvom OAuth 2.0 je realizovaná s využitím externého poskytovateľa identity **Google**. OAuth konfigurácia je spravovaná v prostredí **Google Cloud Platform** [8], kde je registrovaná aplikácia systému a definované oprávnenia na prístup k základným informáciám o používateľovi.

Autorizácia je založená na kontrole identity používateľa a jeho oprávnení pri jednotlivých operáciách. Backend overuje, či má používateľ právo vykonávať požadované operácie.

3.1 Integrácia AI modulov

Integrácia AI modulov je v systéme realizovaná prostredníctvom samostatnej služby, ktorá je oddelená od backendovej aplikácie. Backend zabezpečuje komunikáciu medzi aplikačnou vrstvou a AI modulmi.

Komunikácia s AI modulmi prebieha prostredníctvom REST API. Backend odosiela vstupné dáta, ako sú obrázky dokumentov alebo parametre spracovania, a očakáva odpoveď obsahujúcu výsledky analýzy. Na realizáciu tejto komunikácie je využitá asynchrónna HTTP knižnica, ktorá umožňuje efektívne spracovanie požiadaviek bez blokovania hlavného aplikačného vlákna.

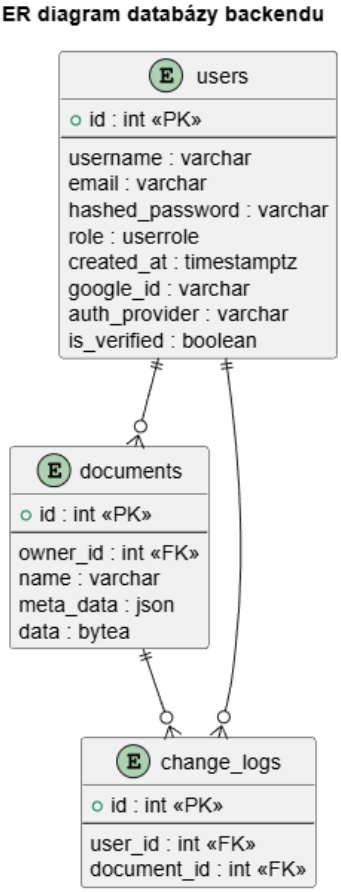
3.2 Práca s databázou a dátový model

Na ukladanie perzistentných dát systému je použitá relačná databáza PostgreSQL. Prístup k databáze je realizovaný prostredníctvom ORM nástroja SQLAlchemy, ktorý umožňuje mapovanie databázových tabuliek na objektové reprezentácie v aplikačnej logike.

Dátový model systému je navrhnutý tak, aby podporoval správu používateľov, historických dokumentov a výsledkov ich spracovania. Základnými entitami sú používateľ, dokument a záznamy o zmenách vykonaných v systéme. Dokumenty sú mapované na konkrétnych používateľov, čím je zabezpečené vlastníctvo a kontrola prístupu k dátam.

Okrem samotných dokumentov databáza uchováva aj metadáta generované AI modulmi, ako sú informácie o štruktúre dokumentu alebo výsledkoch predspracovania. Súčasťou databázy sú aj záznamy o aktivitách používateľov, ktoré umožňujú sledovanie zmien a audit operácií vykonaných v systéme.

Vzťahy medzi jednotlivými entitami a ich hlavné atribúty sú znázornené v entitno-relačnom diagrame databázy, ktorý ilustruje štruktúru dátového modelu a väzby medzi jeho jednotlivými časťami.



3.3 API rozhranie

Backend systému poskytuje REST API rozhranie, prostredníctvom ktorého aplikačná vrstva komunikuje so systémom. API je navrhnuté v súlade so štýlom REST a využíva štandardné HTTP metódy na realizáciu jednotlivých operácií. Zahŕňa endpointy na správu používateľských účtov, autentifikáciu, nahrávanie a správu dokumentov, ako aj spúšťanie spracovania dokumentov prostredníctvom AI modulu. Prístup k vybraným endpointom je chránený autentifikačnými mechanizmami a vyžaduje platný JWT token.

FastAPI framework umožňuje automatické generovanie OpenAPI dokumentácie, ktorá poskytuje prehľad dostupných endpointov, ich vstupných parametrov a návratových hodnôt. Táto dokumentácia uľahčuje integráciu frontendu a zjednodušuje testovanie backendových služieb.

API rozhranie je navrhnuté tak, aby podporovalo rozšíriteľnosť systému a umožňovalo jednoduché doplňanie nových funkcionalít bez nutnosti zásahu do existujúcich častí aplikácie.

Použité zdroje

[1] Sebastián Ramírez, *FastAPI*, 2024, `https://fastapi.tiangolo.com/`, Dostupné z: `https://fastapi.tiangolo.com/` [cit. 2026-01-15]

[2] Mike Bayer, *SQLAlchemy*, 2024, `https://www.sqlalchemy.org/`, Dostupné z: `https://www.sqlalchemy.org/` [cit. 2026-01-15]

[3] PostgreSQL Global Development Group, *PostgreSQL Documentation*, 2024, `https://www.postgresql.org/docs/`, Dostupné z: `https://www.postgresql.org/docs/` [cit. 2026-01-15]

[4] M. Jones, J. Bradley, N. Sakimura, *JSON Web Token (JWT)*, RFC 7519, 2015, `https://datatracker.ietf.org/doc/html/rfc7519`

[5] D. Hardt, *The OAuth 2.0 Authorization Framework*, RFC 6749, 2012, `https://datatracker.ietf.org/doc/html/rfc6749`

[6] Encode OSS, *httpx: A next generation HTTP client for Python*, 2024, `https://www.python-httpx.org/`, Dostupné z: `https://www.python-httpx.org/` [cit. 2026-01-15]

[7] Docker Inc., *Docker Documentation*, 2024, `https://docs.docker.com/`, Dostupné z: `https://docs.docker.com/` [cit. 2026-01-15]

[8] Google LLC, *Google Identity Platform – OAuth 2.0*, 2024, `https://cloud.google.com/identity/docs/oauth2`, Dostupné z: `https://cloud.google.com/identity/docs/oauth2` [cit. 2026-01-15]