**7 Szkolenie Flask**

**Zaawansowane Zagadnienia - Uzupełnienie**

**JWT Tokenizacja**

Materiał ten stanowi uzupełnienie dotychczas poznanej wiedzy. Identyfikacja przy użyciu tokena jest naprawdę często pojawiającym się zagadnieniem przy korzystaniu z API czy integracji lokalnych aplikacji z interfejsami tych wystawionych globalnie w sieci.

Jednym słowem - tokeny (których standard opisuje m.in JWT), służą do sprawdzenia tożsamości użytkownika próbującego skorzystać z, np. zasobów danej aplikacji. Tego rodzaju uwierzytelnienie jest niezbędne przy tworzeniu, np. własnego API. To dlatego, że nie chcemy, aby dowolna osoba mogła dodać nowy rekord do bazy, czy odczytać wrażliwe dane o użytkownikach, których dane są przechowywane.

**Źródła**

Z racji, że temat wymaga przeanalizowania wielu źródeł wiedzy i załapania kontekstu na kilku przykładach, wiedzę oprzemy o następujące źródła informacji:

* Zrozumieć zastosowanie JWT - w jakiego typu aplikacjach się sprawdzi, w jakich nie ([JWT tokens and security - working principles and use cases](https://www.vaadata.com/blog/jwt-tokens-and-security-working-principles-and-use-cases/)).
* Zapoznać się ze strukturą JWT: [JSON Web Token](https://en.wikipedia.org/wiki/JSON_Web_Token)
* Krok po kroku przeanalizować aplikację (z naciskiem na testy): [Token-Based Authentication With Flask – Real Python](https://realpython.com/token-based-authentication-with-flask/)
* Zwrócić szczególnie uwagę na to, z czego składa się payload i czym jest ext, iat, sub w kontekście wyżej stworzonej aplikacji

**Session-based authentication**

Alternatywą dla identyfikacji użytkownika przy użyciu tokena, jest tzw. session-based authentication. Uzupełnij swoją wiedzę o następujące źródło informacji:

<https://stackabuse.com/using-sqlalchemy-with-flask-and-postgresql/>

**PostgreSQL \*\*\*\*\***

Uzupełnij również swoją wiedzę o to, jak połączyć Flaskową aplikację z PostgreSQL. Taka implementacja relacyjnej bazy danych jest obecnie TOPem wśród rozwiązań komercyjnych, dlatego warto poświęcić jej nieco uwagi.

* <https://stackabuse.com/using-sqlalchemy-with-flask-and-postgresql/>

**Testy integracyjne**

Tworząc WEB-owe aplikacje, w celu kompletnego otestowania aplikacji - oprócz testów jednostkowych konieczne będzie stworzenie również testów integracyjnych. Zapoznaj się z materiałem opisującym, jak pisać testy tego poziomu:

[Integration test with Flask Api](https://jozimarback.medium.com/integration-test-with-flask-api-cf1b6fc4be4d)