

MEDCATEGORIZER

APLIKACJA DO GROMADZENIA, KLASYFIKACJI I TAGOWANIA ZDJĘĆ LARYNGOSKOPOWYCH PACJENTÓW

Spis treści

1. Wstęp	1
2. Architektura aplikacji	1
3. Komponenty, których funkcjonalności nie uległy zmianie	3
4. Komponenty nowe lub zmienione	3
4.1. Panel administracyjny	3
4.2. Logowanie	4
4.3. Kartoteki pacjentów	4
4.3.1. Dodawanie nowej kartoteki pacjenta	4
4.3.2. Notatki nt. badanego pacjenta	4
4.3.3. Żądanie sklasyfikowania obiektu	4
5. Klasyfikator	4

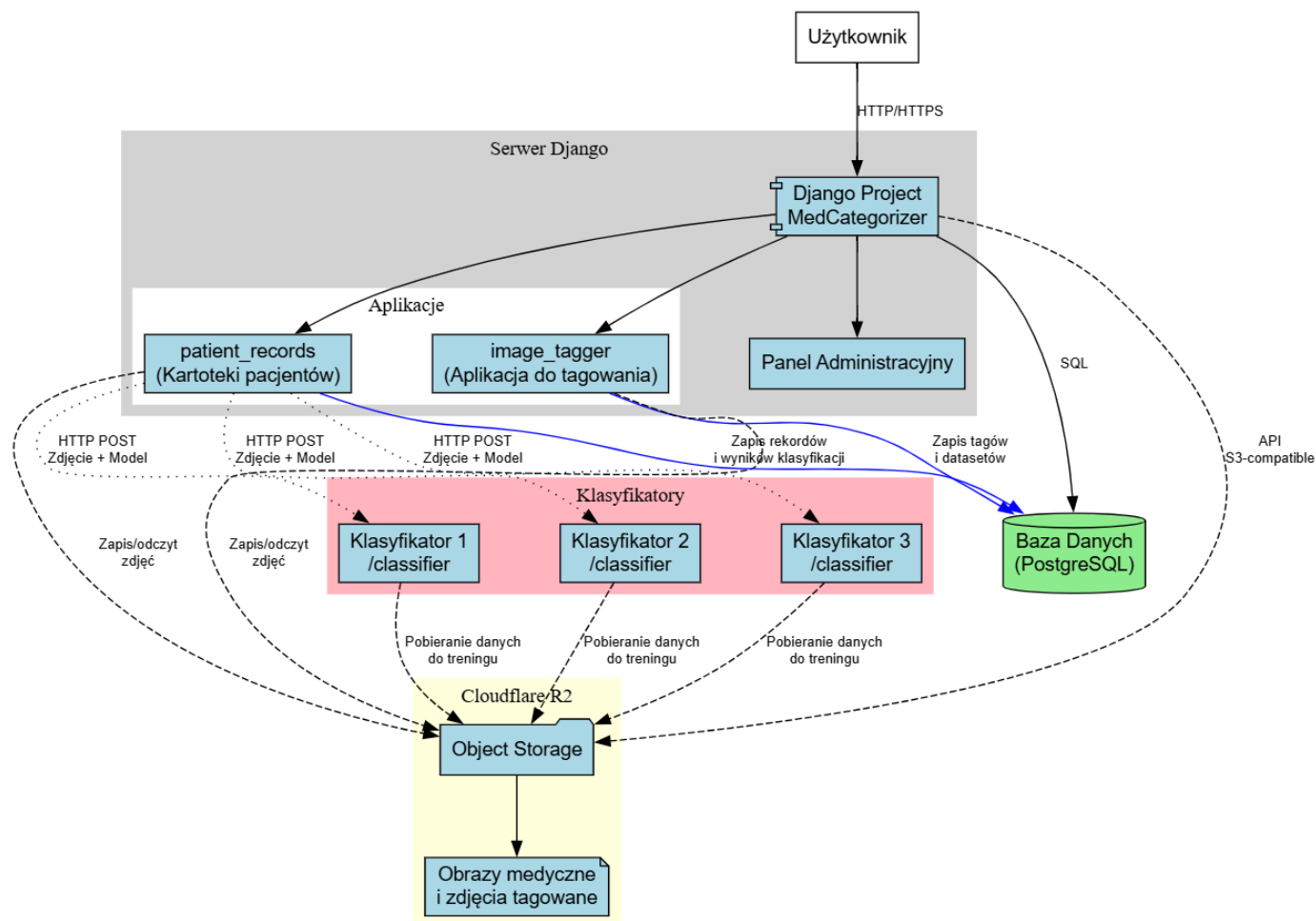
1. Wstęp

Aplikacja webowa MedCategorizer jest narzędziem bazującym na ramie uprzednio wydanej aplikacji MedTagger i stanowi jej rozszerzenie. MedCategorizer pozwala gromadzić zdjęcia laryngoskopowe pacjentów w ramach konta pacjenta a także je tagować (celem stworzenia bazy obrazów uczących klasyfikator) lub klasyfikować (na podstawie decyzji klasyfikatora).

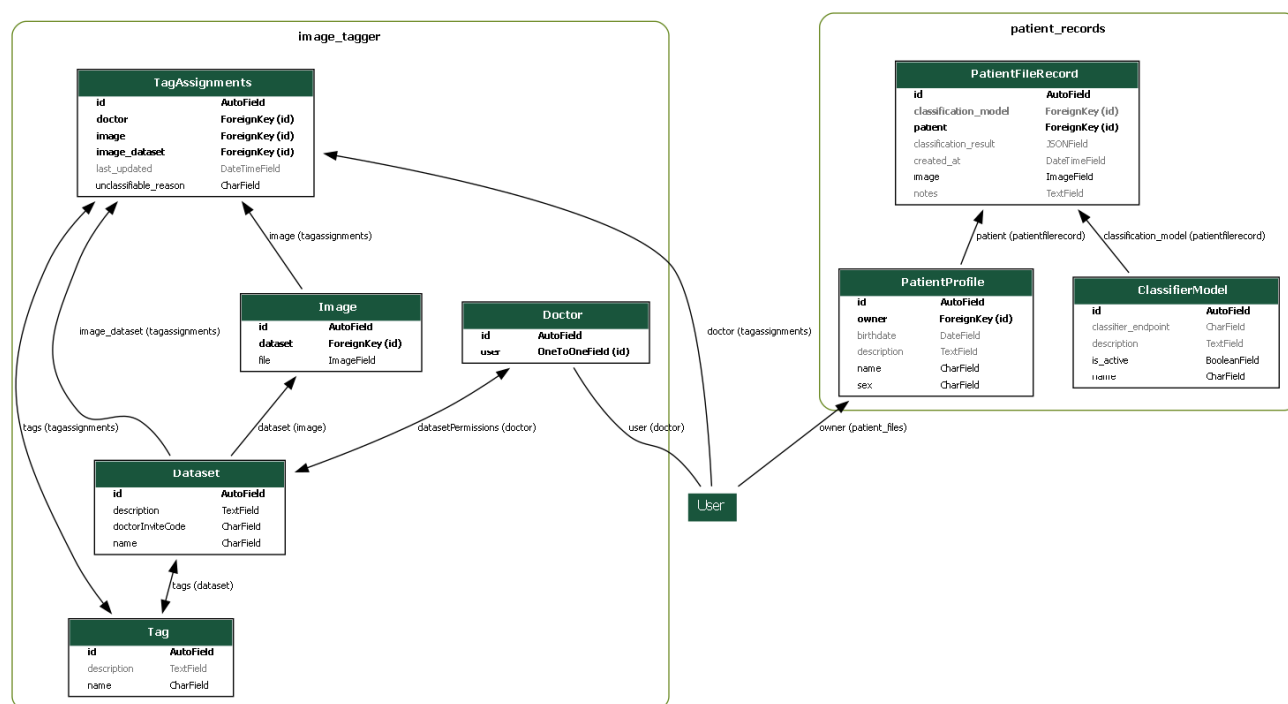
Szczegóły dotyczące komponentów MedTagger znajdują się w dokumencie „Aplikacja webowa do tagowania zdjęć” napisanym wcześniej - te komponenty nie uległy zmianie.

2. Architektura aplikacji

Podobnie jak w przypadku MedTagger, aplikacja webowa MedCategorizer została zbudowana przy użyciu frameworka Django, ze względu na jego prostotę oraz dużą ilość dostępnych bibliotek ułatwiających pracę. Aplikacja składa się z kilku części: panelu administracyjnego całej aplikacji, podaplikacji MedTagger oraz podaplikacji kartoteki pacjentów. Do przechowywania danych użytkowników, danych z kartoteki pacjentów, wyników klasyfikacji i oznaczeń nowych obrazów używana jest baza danych PostgreSQL. Do przechowywania samych zdjęć używana jest chmura Cloudflare R2. Kolejnym elementem aplikacji jest klasyfikator, który zrealizowano tak, aby mógł on być osobnym mikroserwisem. Klasyfikator i kartoteka pacjentów z niego korzystająca są względem siebie niczym czarne skrzynki (black box). Pozwala to na łatwą wymianę, wybór innego typu klasyfikatora lub zwiększenie wydajności tego komponentu poprzez dołączenie nowych instancji. Klasyfikator ma dostęp do chmury Cloudflare R2, z której pobiera zbiory uczące lub obrazy do analizy i sklasyfikowania. Komunikacja między podaplikacją kartoteki pacjentów a klasyfikatorem odbywa się za pomocą API REST. Klasyfikator przyjmuje adres w chmurze R2 obrazu do sklasyfikowania oraz nazwę modelu, który ma być użyty do klasyfikacji. Klasyfikator zwraca wynik klasyfikacji w formacie JSON, który wraz z danymi z kartoteki pacjenta zapisuje się do bazy danych PostgreSQL oraz wyświetla w otwartej karcie pacjenta.



Rysunek 1: Schemat architektury aplikacji



Rysunek 2: Schemat bazy danych

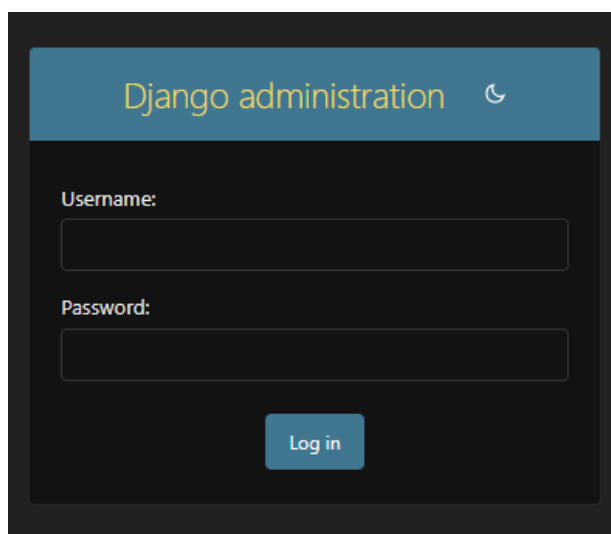
3. Komponenty, których funkcjonalności nie uległy zmianie

1. Sposób przysyłania zbioru obrazów do oznaczenia.
2. Moduł rejestracji.
3. Moduł oznaczania zdjęć.
4. Historia oznaczeń.

4. Komponenty nowe lub zmienione

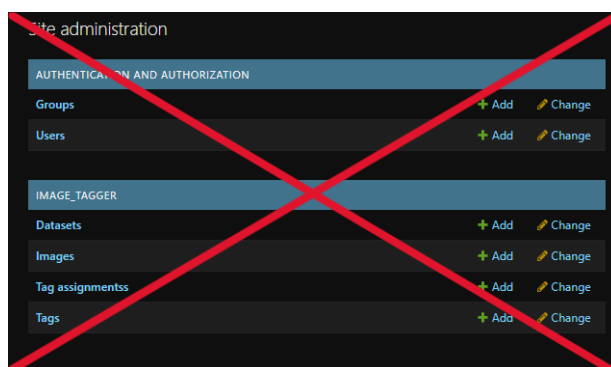
4.1. Panel administracyjny

Aplikacja posiada panel administracyjny, który pozwala na zarządzanie użytkownikami, zdjęciami, oznaczeniami oraz kartoteką pacjentów. Jest to wspólny panel administracyjny dla obydwóch podaplikacji. Panel ten jest dostępny pod adresem / admin i wymaga zalogowania się za pomocą konta administratora.



Rysunek 3: Logowanie do panelu administracyjnego

Po zalogowaniu widocznych jest [JB: ileśtam] sekcji:

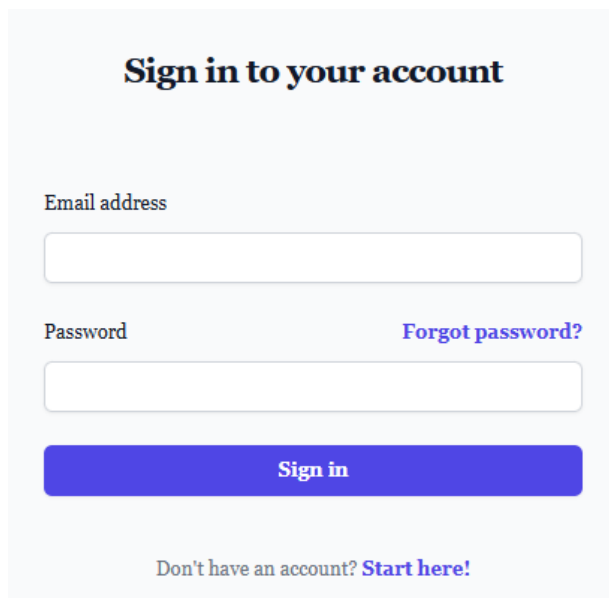


Rysunek 4: Sekcje panelu administracyjnego

[JB: ileśtam sekcji]

1. **Groups** - sekcja pozwalająca na zarządzanie grupami użytkowników (obecnie nie używana)
2. **Users** - sekcja pozwalająca na zarządzanie użytkownikami, ich hasłami oraz uprawnieniami
3. **Datasets** - sekcja pozwalająca na zarządzanie zbiorami zdjęć
4. **Images** - sekcja pozwalająca na zarządzanie pojedynczymi zdjęciami
5. **Tag assignments** - sekcja pozwalająca na zarządzanie i wgląd w oznaczenia zdjęć
6. **Tags** - sekcja pozwalająca na definiowanie dostępnych tagów

4.2. Logowanie



The image shows a login form titled "Sign in to your account". It contains two input fields: "Email address" and "Password". To the right of the password field is a link "Forgot password?". Below the fields is a blue "Sign in" button. At the bottom, there is a link "Don't have an account? Start here!".

Rysunek 5: Formularz logowania

Moduł logowania się do aplikacji jest jeden, spójny dla wszystkich podaplikacji. Pomyślne zalogowanie przyznaje dostęp do każdej podaplikacji.

4.3. Kartoteki pacjentów

4.3.1. Dodawanie nowej kartoteki pacjenta

[JB: oczekuję na zrealizowaną funkcjonalność]

4.3.2. Notatki nt. badanego pacjenta

[JB: oczekuję na zrealizowaną funkcjonalność]

4.3.3. Żądanie sklasyfikowania obiektu

Lekarz [JB: czy na pewno] dodając zdjęcie do kartoteki pacjenta, może zlecić jego sklasyfikowanie przy użyciu wybranego klasyfikatora. Podaplikacja kartoteki wysyła żądanie HTTP na stosowny endpoint klasyfikatora. Żądanie zawiera nazwę klasyfikatora oraz zdjęcie do sklasyfikowania (adres URL w chmurze R2). W odpowiedzi zwracany jest wynik klasyfikacji w formacie JSON, który obrabia się celem estetycznej prezentacji w kartotece oraz zapisuje się go do bazy danych. [JB: tu obrazek]

5. Klasyfikator

Mając na względzie wciąż niewielką ilość danych, w chwili obecnej klasyfikator może nie zwracać w pełni poprawnych klasyfikacji. Niemniej, podjęto decyzję o zaimplementowaniu ramy klasyfikatora, dzięki czemu podmiana modeli klasyfikatora jest możliwa a dalsze prace mogą być prowadzone bez przeszkód i przestojów z wykorzystaniem mock'owego klasyfikatora. Komunikacja z klasyfikatorem odbywa się za pomocą API REST. Klasyfikator przyjmuje adres w chmurze R2 obrazu do sklasyfikowania oraz nazwę modelu, który ma być użyty do klasyfikacji. Klasyfikator zwraca wynik klasyfikacji w formacie JSON i zapisuje go wraz z danymi z kartoteki pacjenta do bazy danych PostgreSQL.