

# MEDCATEGORIZER

## APLIKACJA DO GROMADZENIA, KLASYFIKACJI I TAGOWANIA ZDJĘĆ LARYNGOSKOPOWYCH PACJENTÓW

### Spis treści

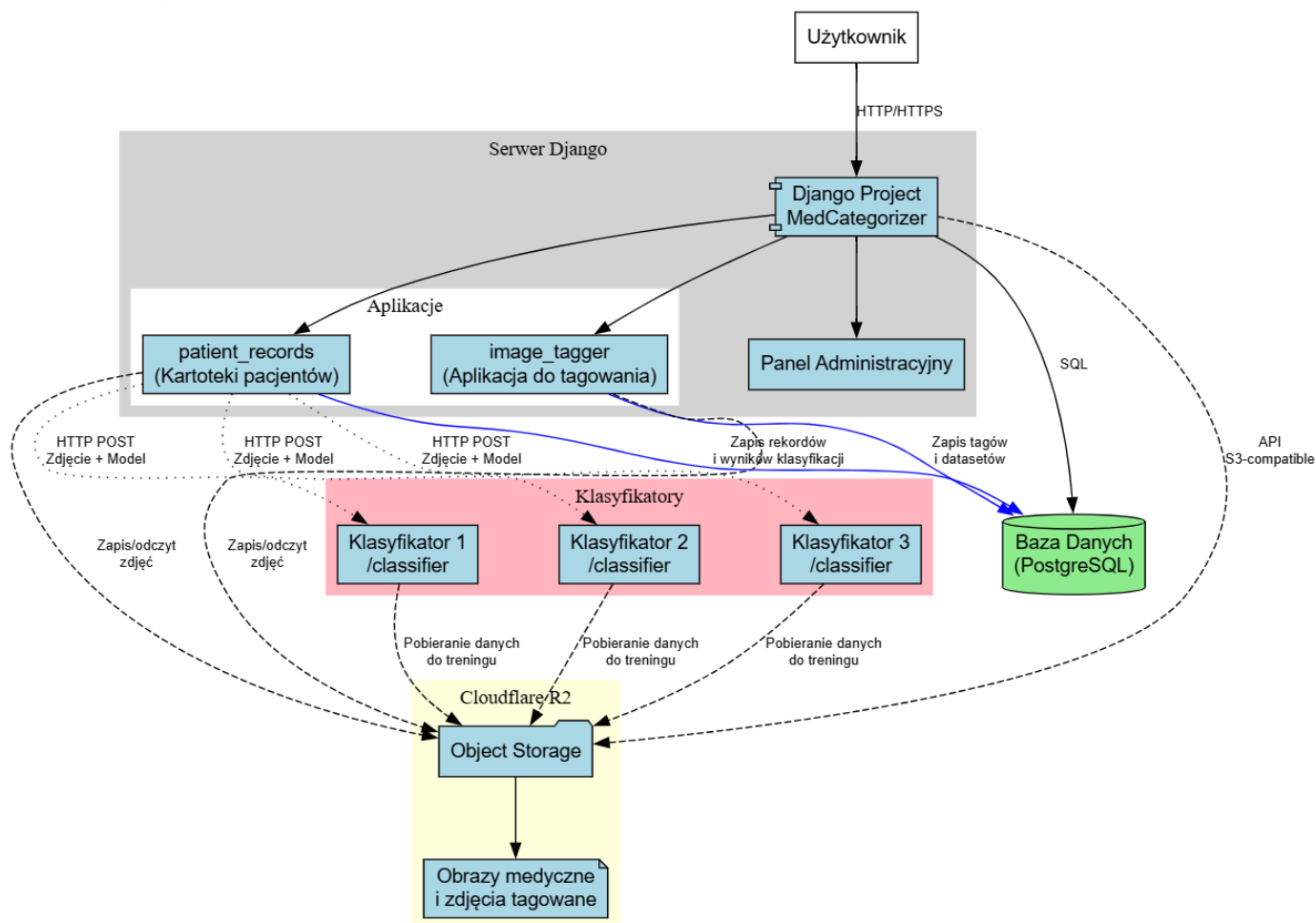
1. Wstęp .....	1
2. Architektura aplikacji .....	1
3. Panel administracyjny .....	3
3.1. Przesyłanie zbioru obrazów do oznaczenia .....	3
4. Panel użytkownika .....	5
4.1. Rejestracja .....	5
4.2. Logowanie .....	5
4.3. Oznaczanie zdjęć i historia oznaczeń .....	6
5. Kartoteki pacjentów .....	6
5.1. Dodawanie nowej kartoteki pacjenta .....	6
5.2. Notatki nt. badanego pacjenta .....	6
5.3. Żądanie sklasyfikowania obiektu .....	6
6. Klasyfikator .....	6

### 1. Wstęp

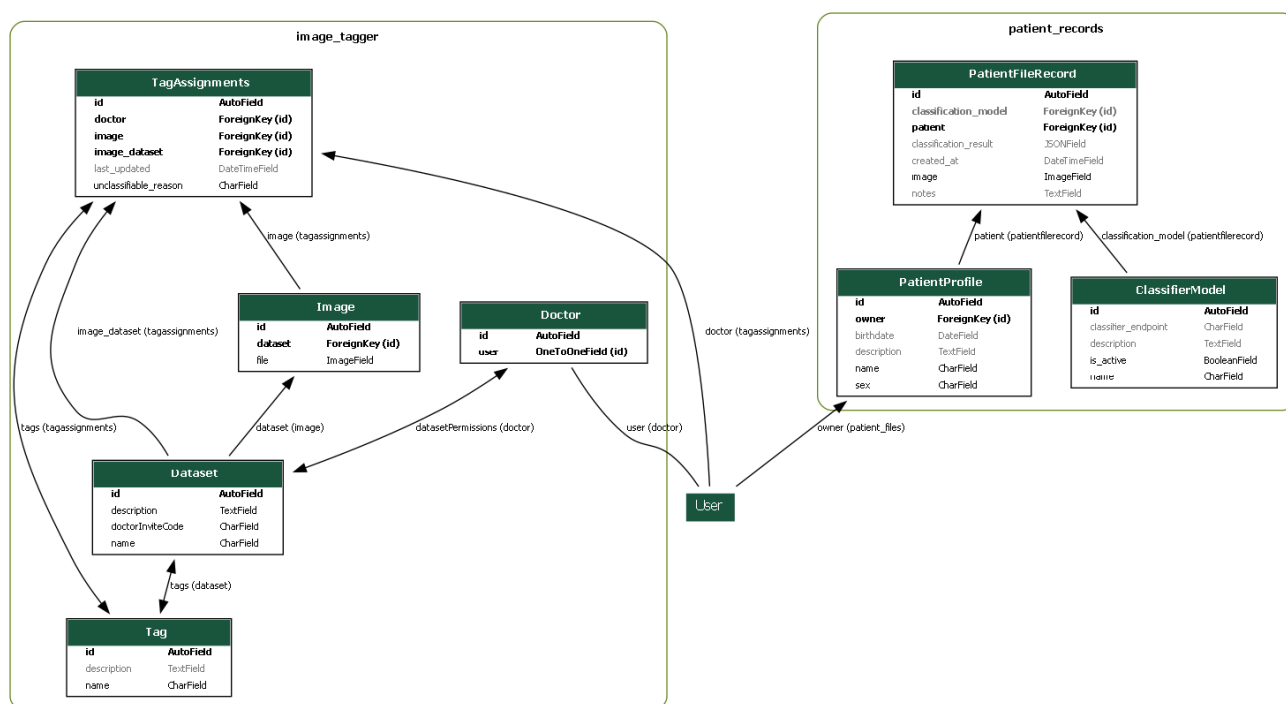
Aplikacja webowa MedCategorizer jest narzędziem bazującym na ramie uprzednio wydanej aplikacji MedTagger i stanowi jej rozszerzenie. MedCategorizer pozwala gromadzić zdjęcia laryngoskopowe pacjentów w ramach konta pacjenta a także je tagować (celem stworzenia bazy obrazów uczących klasyfikator) lub klasyfikować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

### 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 kartoteki pacjentów. Do przechowywania danych użytkowników, danych z kartoteki pacjentów, wyników klasyfikacji i oznaczenia 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. Pozwala to na jego łatwą wymianę, wybór innego typu klasyfikatora lub zwiększenie jego wydajności 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 to zapisuje je wraz z danymi z kartoteki pacjenta do bazy danych PostgreSQL.



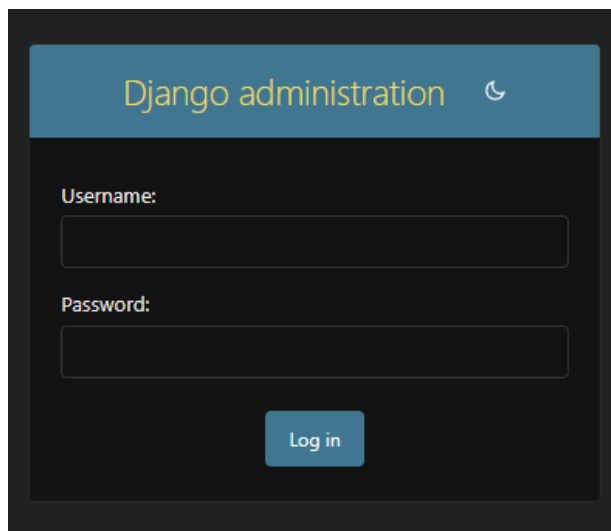
Rysunek 1: Schemat architektury aplikacji



Rysunek 2: Schemat bazy danych

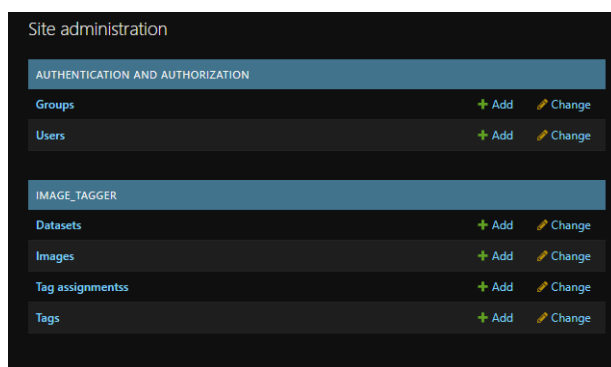
### 3. Panel administracyjny

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:



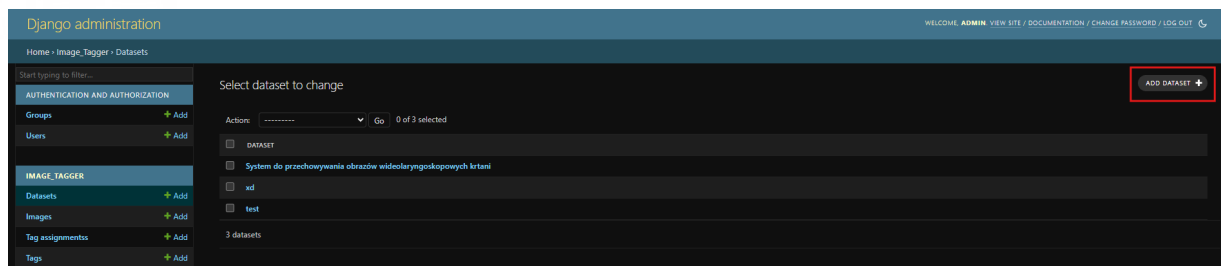
Rysunek 4: Sekcje panelu administracyjnego

[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

#### 3.1. Przesyłanie zbioru obrazów do oznaczenia

Aby utworzyć i przesłać zbiór danych (obrazów) do oznaczania, należy przejść do sekcji **Datasets** i kliknąć przycisk **Add dataset**.

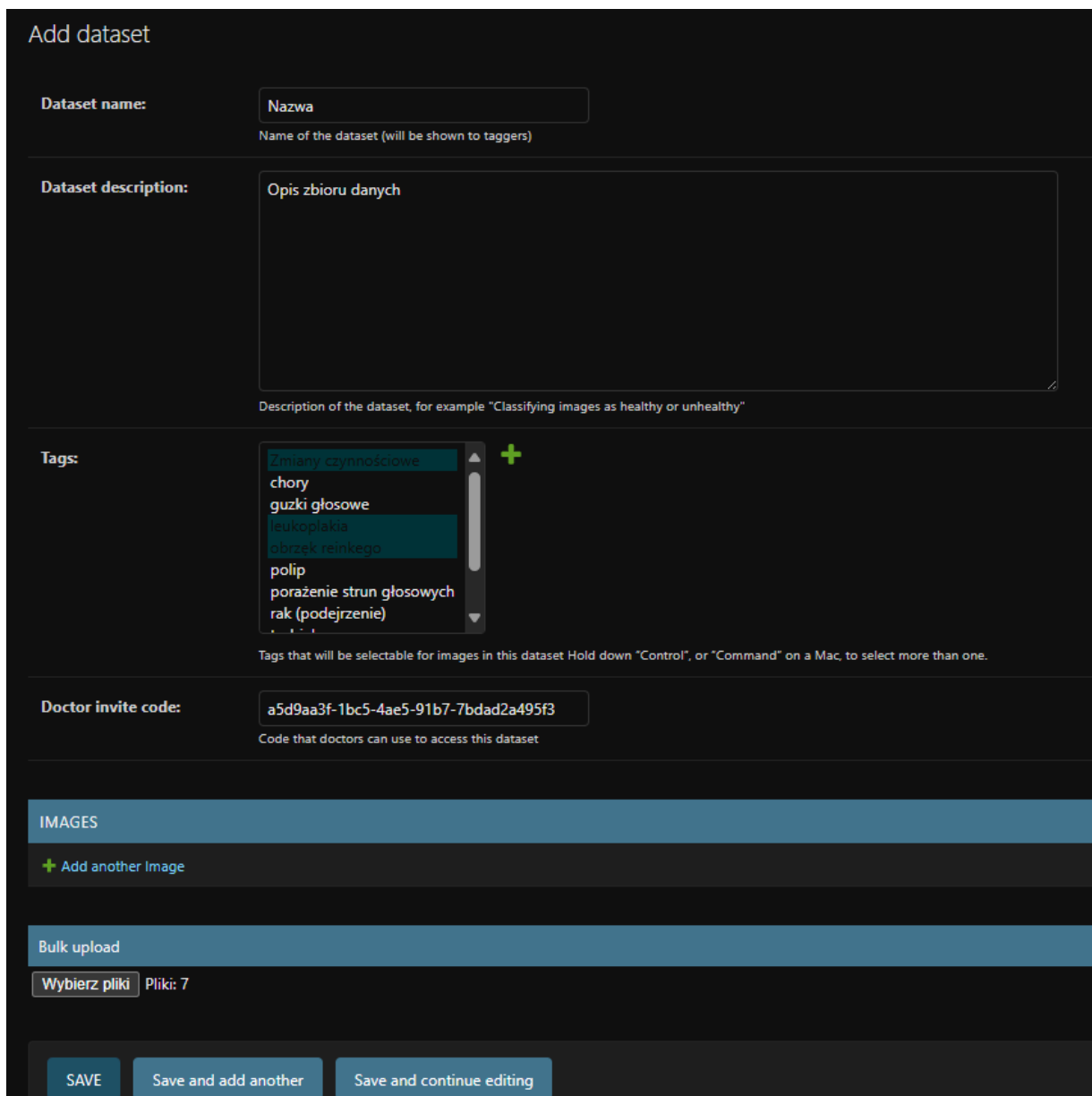


Rysunek 5: Widok listy zbiorów danych

Następnie należy podać:

1. Nazwę zbioru danych
2. Opis zbioru danych
3. Tagi dostępne dla zbioru danych
  - Tagi można dodawać i usuwać z poziomu sekcji **Tags** lub klikając zielony przycisk +.

Oraz przesłać (pojedynczo przyciskiem *Add another Image* lub zbiorowo poprzez sekcję *Bulk upload*) obrazy do oznaczenia.



Rysunek 6: Widok listy zbiorów danych

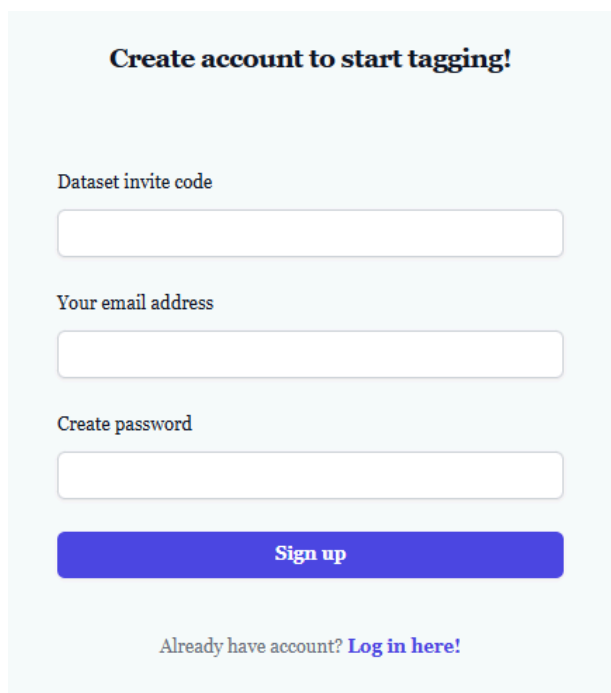
Pole **Doctor invite code** zawiera unikalny kod, pozwalający na dostęp do zbioru danych. Należy go przekazać lekarzowi, który ma oznaczyć zdjęcia.

Na koniec należy kliknąć przycisk **Save**. Zbiór danych zostanie dodany do listy zbiorów danych.

Nie jest wymagane ręczne tworzenie kont użytkowników. Lekarz, który otrzyma kod zaproszenia, może zarejestrować się samodzielnie, podając kod w formularzu rejestracyjnym.

## 4. Panel użytkownika

### 4.1. Rejestracja



**Create account to start tagging!**

Dataset invite code

Your email address

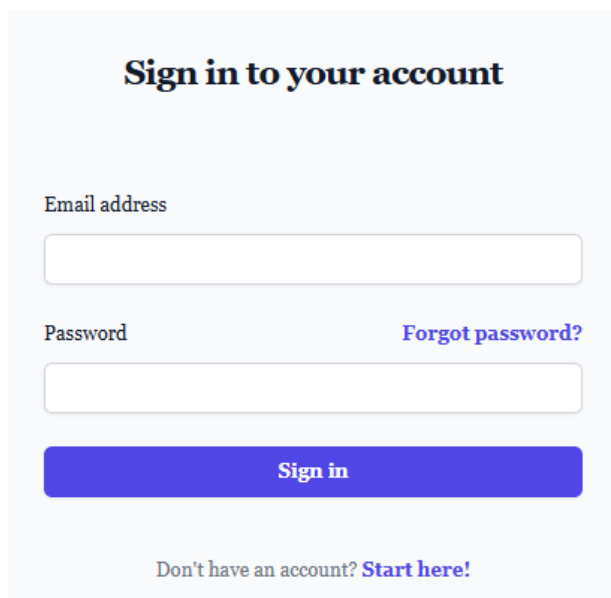
Create password

**Sign up**

Already have account? [Log in here!](#)

Rysunek 7: Formularz rejestracji

### 4.2. Logowanie



**Sign in to your account**

Email address

Password [Forgot password?](#)

**Sign in**

Don't have an account? [Start here!](#)

Rysunek 8: Formularz logowania

Model logowania się do aplikacji jest jeden, spójny dla wszystkich podaplikacji.

#### **4.3. Oznaczanie zdjęć i historia oznaczeń**

Niniejsze funkcjonalności są udsotępnione lekarzom i opisane w dokumentacji aplikacji MedTagger. [JB: czy okej?]

### **5. Kartoteki pacjentów**

#### **5.1. Dodawanie nowej kartoteki pacjenta**

[JB: oczekuję na zrealizowaną funkcjonalność]

#### **5.2. Notatki nt. badanego pacjenta**

[JB: oczekuję na zrealizowaną funkcjonalność]

#### **5.3. Żądanie sklasyfikowania obiektu**

Lekarz [JB: czy na pewno] dodając zdjęcie do kartoteki pacjenta, może zlecić jego sklasyfikowanie przy użyciu wybranego klasyfikatora. Pod względem technicznym, wysyłane jest zapytanie HTTP zawierające nazwę klasyfikatora oraz adres URL zdjęcia do sklasyfikowania. W odpowiedzi zwracany jest wynik klasyfikacji w formacie JSON, który może zostać dalej obrobiony, po czym zostaje on przeniesiony do bazy danych. [JB: tu obrazek]

### **6. 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.