

# Technologie internetowe

## lista zadań nr 9

1. Załóżmy, że budujesz UI dla serwisu Petstore i chcesz zaimplementować zarządzanie obiektami „Pet”. Przejrzyj specyfikację API dostępną pod adresem <https://petstore.swagger.io/v2/swagger.json> i wskaż, które endpointy byłyby pomocne przy budowie takiego klienta i co one reprezentują. Zastanów się też i wskaż 2 endpointy, które w całej specyfikacji najmniej spełniają przedstawione na wykładzie założenia dot. usługi typu REST i dlaczego.

[2p]

2. W przykładowym repozytorium <http://petstore.swagger.io/>:

- Utworzyć po jednej instancji dwóch powiązanych z sobą zasobów (np. *pet* i *store*)
- Wykonać zmiany stanu instancji korzystając z metod *POST*, *PUT*, *PATCH*, *DELETE*, wyjaśniając jednocześnie, co się dokładnie dzieje.

Cztery z powyższych operacji powtórzyć w Postmanie (lub analogicznym narzędziu), przy czym w połowie należy operować na JSON-ie, a w połowie na XML-u.

[2p]

3. Przemyśleć i zaprojektować HTTP API dla przychodni lekarskiej, gdzie oczekiwane są następujące operacje:

- pobranie listy pacjentów,
- pobranie listy wizyt wybranego pacjenta,
- pobranie statusu opłacenia wizyty,
- modyfikacje statusu opłacenia wizyty.

Następnie utworzyć mockup przygotowanego API korzystając z usługi <https://www.mockapi.io/> (lub analogicznej) i dodać przykładowe dane. Na końcu przygotować i zaprezentować odpowiednie testy w Postman (lub innym analogicznym narzędziu).

[3p]

4. Korzystając z Symfony i FOSRestBundle (lub analogicznej technologii), zaimplementować API opisane w poprzednim zadaniu. W ramach implementacji:

- Utworzyć odpowiednie obiekty biznesowe (encje), przy czym nie trzeba dodawać reguł walidacji,
- Repozytorium obiektów można zaimplementować jako tablice obiektów w pamięci (w szczególności nie trzeba konfigurować żadnego DBMS).

[3p]