



**Wydział Elektroniki
i Technik Informatycznych**

POLITECHNIKA WARSZAWSKA

Dokumentacja Usługi Danych

Projekt część 1

Jakub Szczechowicz 303968

Hubert Maciak 303890
4T3

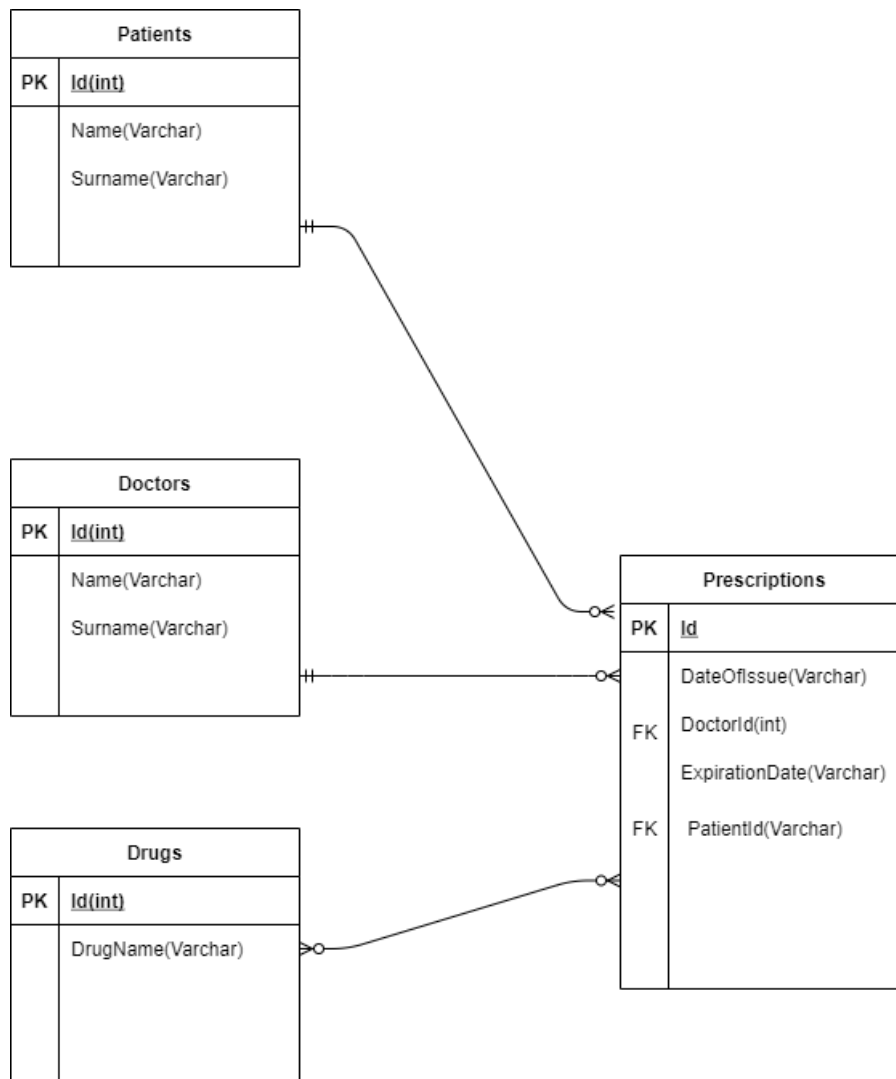
Politechnika Warszawska, Instytut Radioelektroniki i Technik Multimedialnych

18 kwietnia 2021

Spis treści

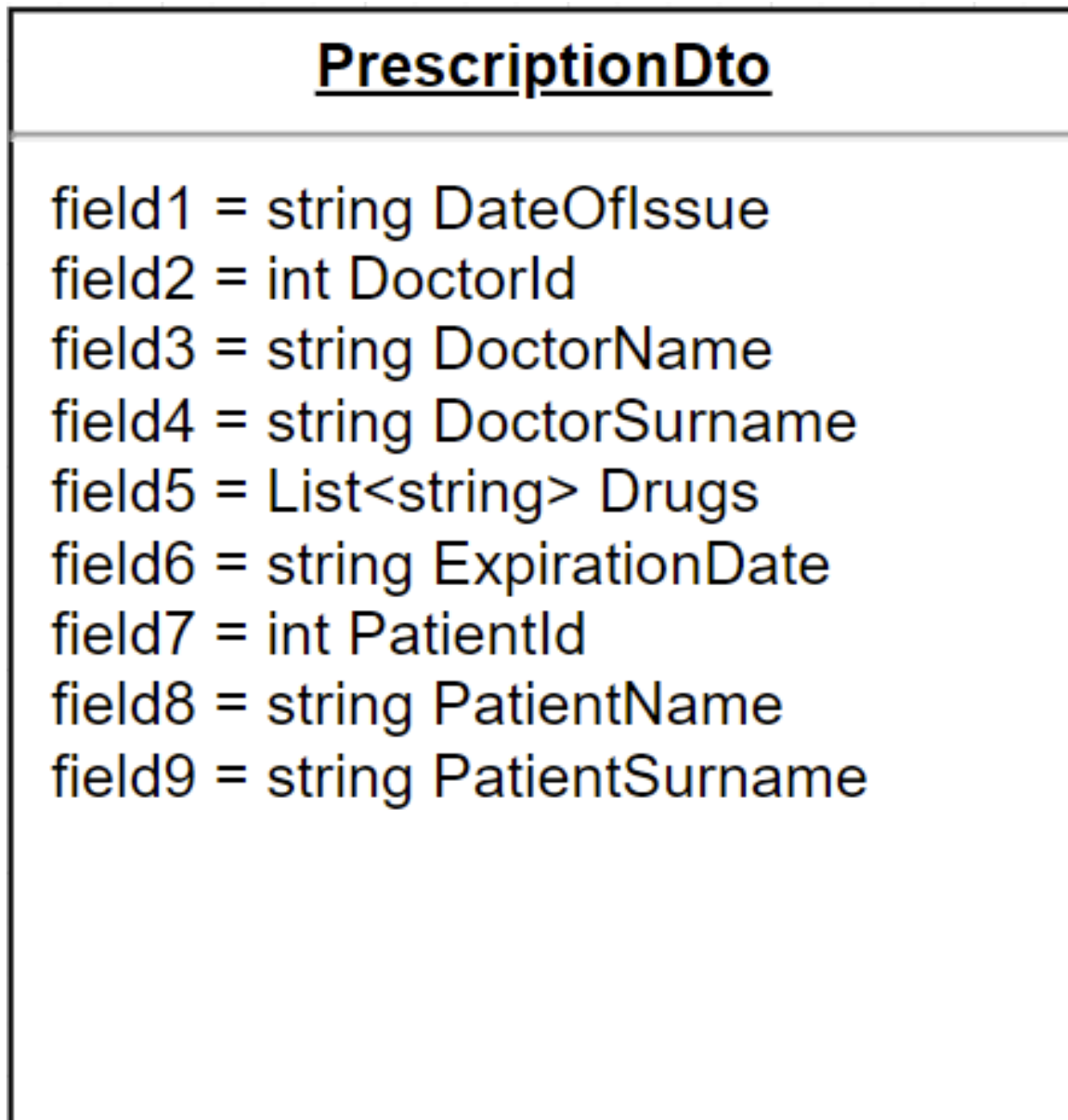
1. Diagram encji	3
2. Formalny model danych	3
3. Przykładowe dane	4
4. Dokumentacja interfejsu	6
5. Struktura rozwiązania	7
6. Skrypty Testujące curl	9

1. Diagram encji



Rys. 1. Diagram Recept

2. Formalny model danych



Rys. 2. Formalny model danych dla "Prescription"

3. Przykładowe dane

Skrypty zakładające bazę danych zostały umieszczone w */Miniprojekty/SkryptyZakladajaceBazeDanych*

	Id	DateOfIssue	DoctorId	ExpirationDate	PatientId
1	1	28.11.2009	1	12.12.2008	1
2	2	12.1.2011	1	24.1.2011	2
3	3	28.5.2012	2	31.6.2012	3
4	4	28.6.2020	3	28.7.2020	4
5	5	21.7.2020	4	29.7.2020	5
6	6	11.8.2020	4	21.8.2020	6
7	7	21.9.2020	2	29.9.2020	7
8	8	14.10.2020	3	28.10.2020	8
9	9	16.11.2020	5	24.11.2020	9
10	10	17.12.2020	6	27.12.2020	1
11	11	18.6.2020	5	25.6.2020	2
12	12	19.5.2020	7	26.5.2020	3
13	13	22.7.2020	7	28.7.2020	4
14	14	10.9.2020	5	19.9.2020	5

Rys. 3. Widok danych leków

	DrugId	DrugName
1	1	Paracetamol
2	2	Ibuprofen
3	3	Metafen
4	4	Acenol
5	5	Pawulon
6	6	Ketoprofen
7	7	Nolicin
8	8	Artresan

Rys. 4. Widok danych pacjentów

	PatientId	Name	Surname
1	1	Halina	Nowicka
2	2	Grażyna	Kowal
3	3	Elżbieta	Kowalska
4	4	Henryk	Kowalczyk
5	5	Jan	Zabrzewski
6	6	Marcin	Niemiec
7	7	Paweł	Polak
8	8	Marek	Małek
9	9	Adam	Jabłoński

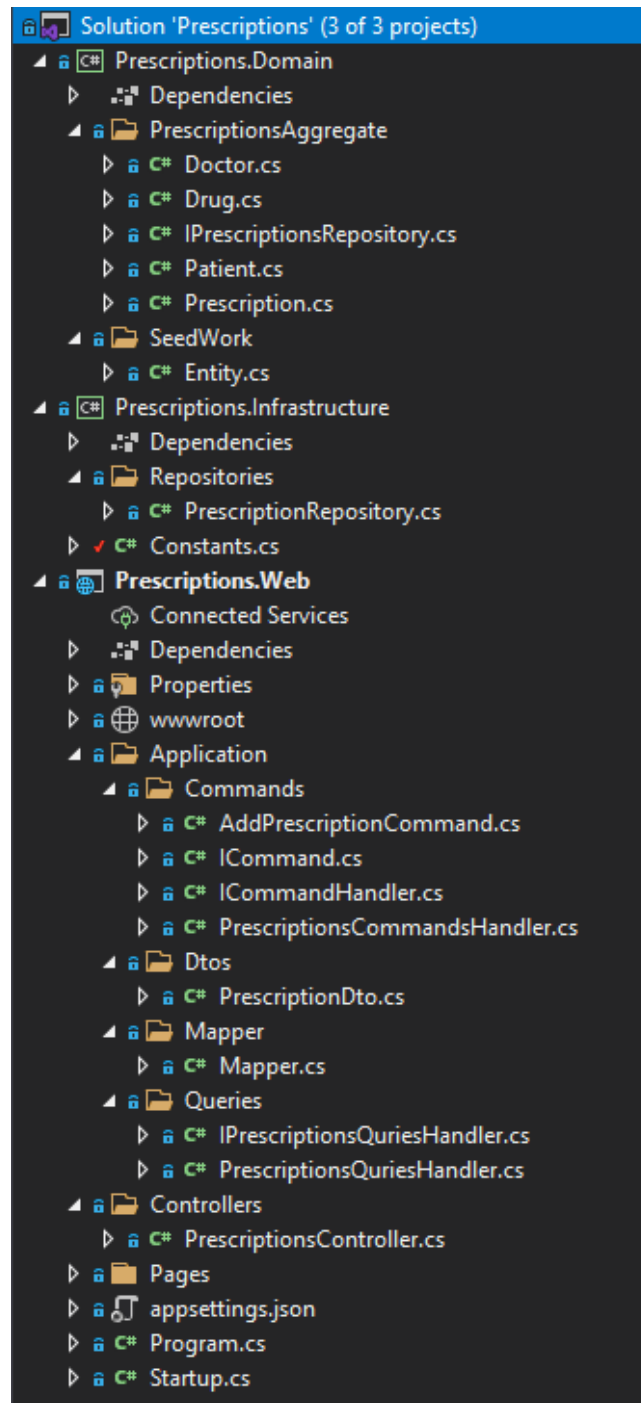
Rys. 5. Widok danych wizyt

4. Dokumentacja interfejsu

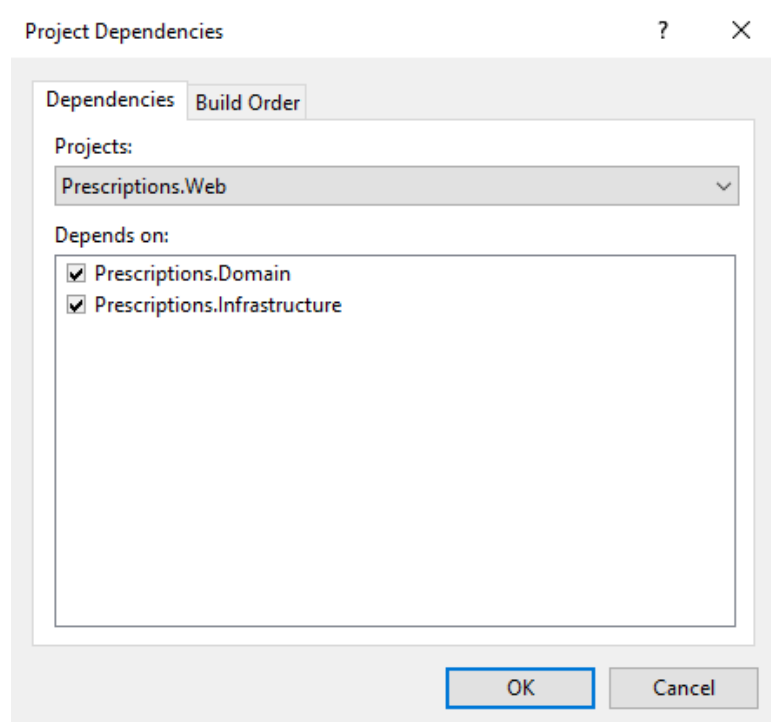
W naszym interfejsie zaimplementowaliśmy podstawowe funkcje pozwalające na korzystanie usługi danych. Ze względu na uniwersalność naszych funkcji jesteśmy w stanie dodać kolejne w oparciu o już istniejące. Przykładowe, zaimplementowane operacje na danych:

- AddPrescription(POST) (Dodanie kolejnej recepty z parametrami: imię i nazwisko pacjenta oraz doktora, lista leków, data wydania recepty, data po której recepta się przeterminuje
- GetByDoctorId(GET) (Wyszukanie recept wydanych przez doktora podając jego Id. (doctorId) Zwraca listę recept wydanych przez danego lekarza)
- GetByPatientId(GET) (Wyszukanie recept pacjenta podając jego Id. (patientId) Zwraca listę recept danego pacjenta)

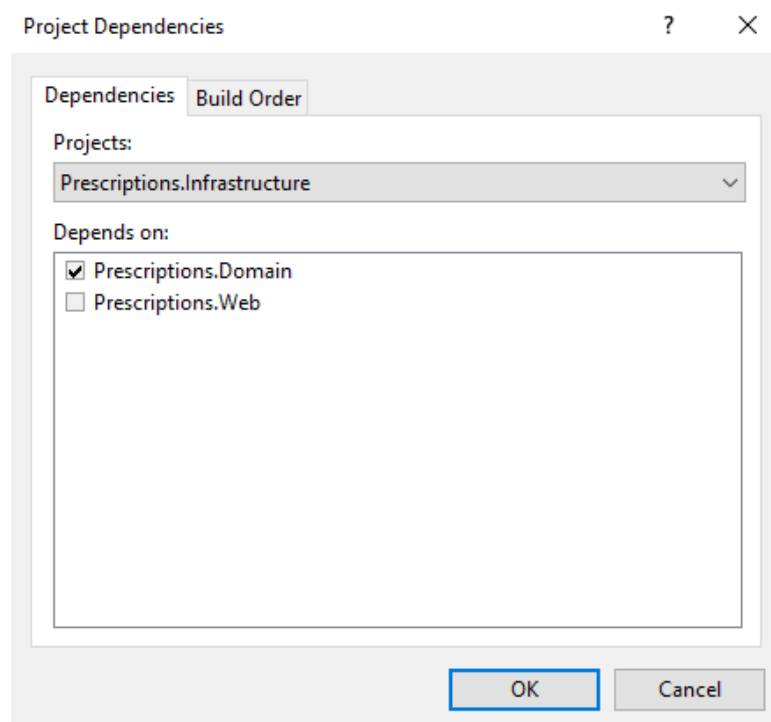
5. Struktura rozwiązania



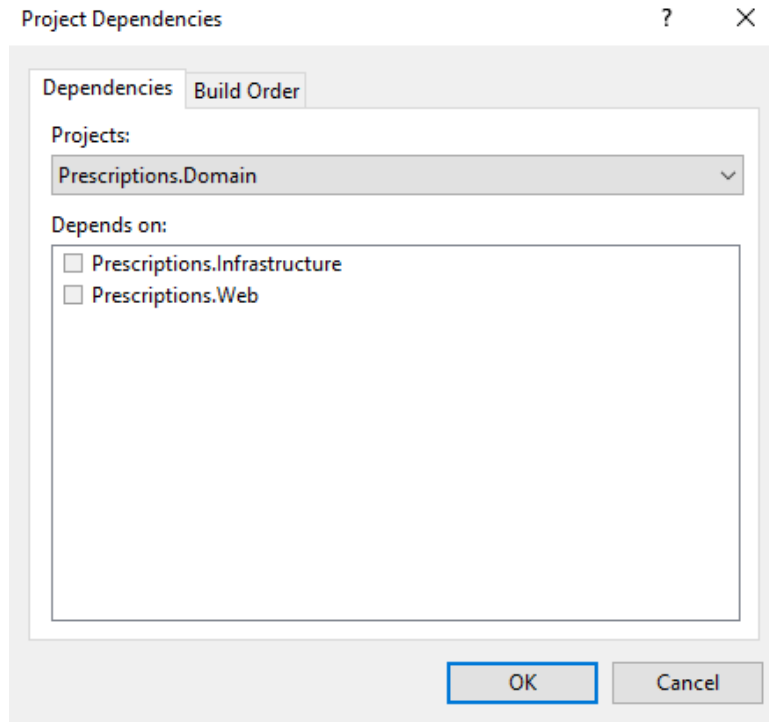
Rys. 6. Widok struktury usługi danych "Prescriptions"



Rys. 7. Widok zależności dla Prescriptions.Web



Rys. 8. Widok zależności dla Prescriptions.Infrastructure"



Rys. 9. Widok zależności dla Prescriptions.Domain"

6. Skrypty Testujące curl

Wydruk 1. Pobranie z aplikacji danych recept po id doktora

```
1 curl -X 'GET' \
2   'https://localhost:44303/prescriptions-by-doctor-id?doctorId=1' \
3   -H 'accept: text/plain '
```

Wydruk 2. Pobranie z aplikacji danych recept po id pacjenta

```
1 curl -X 'GET' \
2   'https://localhost:44303/prescriptions-by-patient-id?patientId=1' \
3   -H 'accept: text/plain '
```

Wydruk 3. Wysłanie nowej recepty do aplikacji danych

```
1 curl -X 'POST' \
2   'https://localhost:44303/prescription-add' \
3   -H 'accept: */*' \
4   -H 'Content-Type: application/json' \
5   -d '{
6     "dateOfIssue": "15.01.2020",
7     "doctorName": "Janusz",
8     "doctorSurname": "Nowak",
9     "drugs": [
10      "apap"
11    ],
12     "expirationDate": "30.01.2020",
13     "patientName": "Norbert",
14     "patientSurname": "Kowalski"
15  }'
```