

# **Interfejs wymiany danych poprzez CGM ePortal Web Service**

Wersja 3.0.0

16.01.2018

## Spis treści

1.	Architektura integracji.....	3
2.	Użytkownik/pacjent .....	3
2.1	Pobranie danych pacjenta .....	3
2.2	Dodanie nowego pacjenta.....	4
3.	Rezerwacja wizyt .....	5
3.1	Pobranie listy lekarzy.....	5
3.2	Pobranie listy jednostek organizacyjnych .....	6
3.3	Pobranie listy usług .....	6
3.4	Wyszukiwanie wolnych przedziałów .....	6
3.5	Rezerwacja wolnego przedziału (wizyty).....	7
4.	Wizyty pacjenta .....	8
4.1	Lista wizyt pacjenta .....	8
4.2	Anulowanie wizyty .....	8
5.	eSamokontrola .....	9
5.1	Lista dostępnych pomiarów .....	9
5.2	Lista zleconych pomiarów .....	10
5.3	Pobranie formularza dla wybranego pomiaru .....	10
5.4	Zapisanie formularza dla wybranego pomiaru.....	12
5.5	Pobranie listy pomiarów z wypełnionymi formularzami.....	13
6.	Dokumentacja pacjenta.....	13
6.1	Dodanie załącznika do HIS.....	13
7.	eWyniki.....	14
7.1	Pobranie listy wyników pacjenta.....	14
7.2	Pobranie pliku wyniku .....	14

## 1. Architektura integracji

CGM ePortal (CN) jest aplikacją webową uruchomioną na serwerze. CN wystawia WebService oparty o protokół SOAP do integracji z partnerami. Połączenie pomiędzy serwerem aplikacyjnym CN a serwerami aplikacyjnymi partnerów jest oparte o VPN. Obsługiwane VPNy to

- a) openVPN (preferowany)
- b) IPSEC

Usługi partnerów komunikują się z WebService'm poprzez połączenie https co zapewnia dodatkową ochronę danych. Wszystkie metody sieciowe są dostępne poprzez interfejs opisany przez plik WSDL znajdujący się na serwerze aplikacyjnym CN. Po nawiązaniu połączenia VPN partner będzie znał adres do serwera CN. Mając adres serwera CN usługa będzie dostępna pod adresem [https://VPN\\_SERVER\\_ADDRESS:10443/eportal-ws/EPortalService](https://VPN_SERVER_ADDRESS:10443/eportal-ws/EPortalService) a plik WSDL pod adresem [https://VPN\\_SERVER\\_ADDRESS:10443/eportal-ws/EPortalService?wsdl](https://VPN_SERVER_ADDRESS:10443/eportal-ws/EPortalService?wsdl).

## 2. Użytkownik/pacjent

Aby pracować w obrębie pacjenta / użytkownika (rezerwować wizyty, pobierać dane pacjenta itp.) aplikacja wymaga identyfikatora pacjenta.

### 2.1 Pobranie danych pacjenta

Nazwa metody: getPatientData

Opis: Metoda zwraca obiekt dane pacjenta

Parametry wejściowe:

Nazwa	Typ	Opis
Pesel	String	Pesel użytkownika/pacjenta

Wartości zwracane: Obiekt PatientData

Nazwa	Typ	Opis
UserId	Integer	Identyfikator użytkownika/pacjenta
Name	String	Imię pacjenta
Surname	String	Nazwisko pacjenta
Pesel	String	Pesel pacjenta
DayOfBirth	Date	Data urodzenia pacjenta
Address	Obiekt Address	Dane adresowe pacjenta

Obiekt Address

Nazwa	Typ	Opis
AddressID	Integer	Identyfikator adresu
ContactID	Integer	Identyfikator danych kontaktowych
Street	String	Ulica
HomeNo	String	Numer domu
FlatNo	String	Numer mieszkania
Place	String	Miejscowość
PostalCode	String	Kod pocztowy
Email	String	Email pacjenta
Phone	String	Numer telefonu pacjenta

## 2.2 Dodanie nowego pacjenta

Nazwa metody: savePatient

Opis: Metoda dodaje pacjenta do systemu

Parametry wejściowe: Obiekt PatientData

Nazwa	Typ	Opis
UserId	Integer	Identyfikator użytkownika/pacjenta
Name	String	Imię pacjenta
Surname	String	Nazwisko pacjenta
Pesel	String	Pesel pacjenta
DayOfBirth	Date	Data urodzenia pacjenta
Address	Obiekt Address	Dane adresowe pacjenta

Obiekt Address

Nazwa	Typ	Opis
AddressID	Integer	Identyfikator adresu
ContactID	Integer	Identyfikator danych kontaktowych
Street	String	Ulica
HomeNo	String	Numer domu
FlatNo	String	Numer mieszkania
Place	String	Miejscowość
PostalCode	String	Kod pocztowy
Email	String	Email pacjenta
Phone	String	Numer telefonu pacjenta

Wartości zwracane: Identyfikator nowoutworzonego pacjenta lub jeden z wyjątków:

- PatientAlreadyExists – w przypadku gdy pacjent o podanym pesel istnieje już w systemie
- EmptyPesel – w przypadku gdy nie został przekazany numer pesel
- EmptyContactData – w przypadku gdy nie zostały przekazane dane email i phone

Nazwa	Typ	Opis
-------	-----	------

PatientId	Integer	Identyfikator danych kontaktowych
-----------	---------	-----------------------------------

### 3. Rezerwacja wizyt

Aby dokonać rezerwacji wizyty należy wyszukać wolne przedziały na grafiku po czym zarezerwować odpowiedni przedział. Do wyszukiwania wolnych przedziałów wymagane jest podanie id usługi oraz opcjonalnie , id lekarza i jednostki organizacyjne. Do wyszukiwania lekarzy , jednostki organizacyjnej i usług służy wspólny obiekt kryteriów zawężających obszar wyszukiwania opisany poniżej

Obiekt kryteriów zawężających obszar wyszukiwania

Nazwa	Typ	Opis
OrgUnitIds (opcjonalnie)	List< Integer >	Zawęża wyszukiwanie do jednostek organizacyjnych podanych w liście
DoctorId (opcjonalnie)	Integer	Zawęża wyszukiwanie do lekarza
ServiceId (opcjonalnie)	Integer	Zawęża wyszukiwanie do usługi

#### 3.1 Pobranie listy lekarzy

Nazwa metody: getDoctors

Opis: Metoda zwraca listę dostępnych lekarzy

Parametry wejściowe: Obiekt kryteriów zawężających obszar wyszukiwania opisany wyżej. Dla przykładu gdy chcemy pobrać listę lekarzy którzy wykonują usługę „Porada lekarska” o id 32 w jednostce organizacyjnej „Poradnia ortopedyczna” o id 123 to przekazujemy w polu OrgUnitIds listę z jedną wartością 123 a w polu ServiceId wartość 32.

Wartość zwracana: Lista obiektów Doctor

Nazwa	Typ	Opis
Id	Integer	Identyfikator lekarza
Name	String	Imię lekarza
Surname	String	Nazwisko lekarza
Title	String	Tytuł naukowy
NationalIdNo	String	Pesel

### 3.2 Pobranie listy jednostek organizacyjnych

Nazwa metody: getOrgUnits

Opis: Metoda zwraca listę dostępnych jednostek

Parametry wejściowe: Obiekt kryteriów zawężających obszar wyszukiwania opisany wyżej

Wartość zwracana: Lista obiektów OrgUnitDataDto

Nazwa	Typ	Opis
OrgUnitId	Integer	Id jednostki medycznej
OrgUnitName	String	Nazwa jednostki
OrgUnitAddress	String	Pełny adres jednostki
Message	String	Dodatkowe informacje
Phone	String	Telefon do jednostki
Fax	String	Fax do jednostki
AddressId	Integer	Identyfikator adresu jednostki
StatCode	String	Kod resortowy jednostki
Description	String	Opis jednostki medycznej
MapCoordinateX	Float	Koordinata do pozycji GPS jednostki
MapCoordinateY	Float	Koordinata do pozycji GPS jednostki

### 3.3 Pobranie listy usług

Nazwa metody: getServices

Opis: Metoda zwraca listę dostępnych usług

Parametry wejściowe: Obiekt kryteriów zawężających obszar wyszukiwania opisany wyżej

Wartość zwracana: Lista obiektów Service

Nazwa	Typ	Opis
Id	Integer	Id usługi
Name	String	Nazwa usługi
Code	String	Kod usługi

### 3.4 Wyszukiwanie wolnych przedziałów

Nazwa metody: getFreeVisits

Opis: Metoda zwraca listę dostępnych przedziałów czasowych na które można zarezerwować wizytę

Parametry wejściowe: Obiekt SearchVisitsCriteria – w przypadku braku wymaganych pól zwracany jest wyjątek InsufficientCriteriaException

Nazwa	Typ	Opis
OrgUnitId (opcjonalne)	Integer	Id jednostki organizacyjnej
ServiceId (wymagane)	Integer	Id usługi
DoktorId (opcjonalnie)	Integer	Id lekarza
StartDt (wymagane)	Date	Data od kiedy zacząć szukanie wolnego terminu

Wartość zwracana: Lista obiektów TimeSlot

Nazwa	Typ	Opis
TimeSlotId	Integer	Id usługi
ServiceID	Integer	Nazwa usługi
ServiceName	String	Kod usługi
StartDt	DateTime	Data+czas początku wizyty
StopDt	DateTime	Data+czas zakończenia wizyty
OrgUnitID	Integer	Identyfikator jednostki medycznej
DoctorId	Integer	Identyfikator lekarza
ClinicName	String	Nazwa jednostki medycznej

### 3.5 Rezerwacja wolnego przedziału (wizyty)

Nazwa metody: saveVisit

Opis: Metoda rezerwująca wizytę

Parametry wejściowe: Obiekt Reservation

Nazwa	Typ	Opis
UserId	Integer	Id użytkownika
TimeSlotId	Integer	Id przedziału czasowego
DoktorId	Integer	Id lekarza
Date	DateTime	Data+czas początku wizyty
ServiceId	Integer	Id usługi
OrgUnitId	Integer	Id jednostki organizacyjnej

Wartość zwracana: Identyfikator techniczny wizyty. W przypadku błędów możliwe wyjątki:

- InsufficientCriteriaException – wyjątek w przypadku braku wymaganych pól
- SomeoneWasFasterException – wyjątek w przypadku gdy w międzyczasie została zarezerwowana wizyta

## 4. Wizyty pacjenta

### 4.1 Lista wizyt pacjenta

Nazwa metody: `getPatientVisits`

Opis: Metoda zwracająca listę wizyt danego pacjenta

Parametry wejściowe: Identyfikator pacjenta Można go uzyskać znając pesel poprzez metodę `getPatientData`

Nazwa	Typ	Opis
UserId (wymagane)	Integer	Id jednostki organizacyjnej

Wartość zwracana: Lista obiektów `Visit`

Nazwa	Typ	Opis
Uuid	Integer	Identyfikator wizyty
Status	Enum <code>VisitStatus</code>	Status usługi
StartDt	DateTime	Data+czas początku wizyty
StopDt	DateTime	Data+czas zakończenia wizyty
Gab	Obiekt <code>OrgUnitInfo</code>	Dane gabinetu
Clinic	Obiekt <code>OrgUnitInfo</code>	Dane poradni
Doctor	Obiekt <code>Doctor</code>	Dane lekarza (obiekt opisany w metodzie <code>getDoctor</code> )

Enum `VisitStatus` zawiera następujące wartości

- `REGISTERED` – oznacza wizytę zarezerwowaną przez ePortal
- `CANCELED` – oznacza wizytę anulowaną przez ePortal

Obiekt `OrgUnitInfo`

Nazwa	Typ	Opis
Id	Integer	Identyfikator jednostki organizacyjnej
Code	String	Kod jednostki
Name	String	Nazwa jednostki
Message	String	Dodatkowe informacje
MapCoordinateX	Float	Koordynata do pozycji GPS jednostki
MapCoordinateY	Float	Koordynata do pozycji GPS jednostki
StatCode7thPart	String	7 część kodu resortowego jednostki

### 4.2 Anulowanie wizyty

Nazwa metody: `cancelVisit`



Opis: Metoda anulująca zarezerwowaną wizytę

Parametry wejściowe: Identyfikator wizyty

Nazwa	Typ	Opis
Uuid (wymagane)	String	Id jednostki organizacyjnej

Wartość zwracana: Brak. W przypadku wystąpienia błędów możliwe wyjątki:

- VisitCancelationFailure – błąd systemu podczas anulowania
- VisitNotExists – wizyta nie istnieje

## 5. eSamokontrola

Pomiary eSamokontroli są definiowane przez użytkowników systemu. Do każdego pomiaru dołączony jest formularz tworzone przez użytkowników systemu który wypełnia pacjent. Każdy formularz może zawierać wiele atrybutów.

### 5.1 Lista dostępnych pomiarów

Nazwa metody: getSelfControlMeasurements

Opis: Metoda zwraca listę dostępnych pomiarów do wykonania przez pacjenta

Parametry wejściowe: Brak

Wartość zwracana: Lista obiektów SelfControlMeasurementDto

Nazwa	Typ	Opis
OrderId	Integer	Identyfikator zlecenia (jeśli pomiar jest zleceniem)
PerfServiceId	Integer	Identyfikator powiązany z usługą
ServiceId	Integer	Identyfikator usługi
Name	String	Nazwa pomiaru
PerfDate	DateTime	Data+czas pomiaru
DescrDataTypeSysFctId	Integer	Id potrzebne do pobrania definicji podpętego formularza
DescrDataTypeFormDefId	Integer	Id potrzebne do pobrania definicji podpętego formularza
DescrDataTypeId	Integer	Id potrzebne do pobrania definicji podpętego formularza
DescrDataId	Integer	Id potrzebne do pobrania definicji podpętego formularza
DescrDataFormInstanceId	Integer	Id potrzebne do pobrania definicji podpętego formularza

## 5.2 Lista zleconych pomiarów

Nazwa metody: `getOrderedSelfControlMeasurements`

Opis: Metoda zwraca listę zleconych pomiarów do wykonania przez pacjenta

Parametry wejściowe: Identyfikator pacjenta

Nazwa	Typ	Opis
UserId	Integer	Identyfikator użytkownika

Wartość zwracana: Lista obiektów `SelfControlMeasurementDto`

Nazwa	Typ	Opis
OrderId	Integer	Identyfikator zlecenia (jeśli pomiar jest zleceniem)
PerfServiceId	Integer	Identyfikator powiązany z wykonywaną usługą
ServiceId	Integer	Identyfikator usługi
Name	String	Nazwa pomiaru
PerfDate	DateTime	Data+czas pomiaru
DescrDataTypeSysFctId	Integer	Id potrzebne do pobrania definicji podpiętego formularza
DescrDataTypeFormDefId	Integer	Id potrzebne do pobrania definicji podpiętego formularza
DescrDataTypeId	Integer	Id potrzebne do pobrania definicji podpiętego formularza
DescrDataId	Integer	Id potrzebne do pobrania definicji podpiętego formularza
DescrDataFormInstanceId	Integer	Id potrzebne do pobrania definicji podpiętego formularza

## 5.3 Pobranie formularza dla wybranego pomiaru

Nazwa metody: `getFormMeasurementDefinition`

Opis: Metoda zwraca definicje formularza podpiętego do pomiaru

Parametry wejściowe:

Nazwa	Typ	Opis
DescrDataTypeSysFctId	Integer	Id pobrane z metody <code>getSelfControlMeasurements</code> lub <code>getOrderedSelfControlMeasurements</code>
DescrDataTypeFormDefId	Integer	Id pobrane z metody <code>getSelfControlMeasurements</code> lub <code>getOrderedSelfControlMeasurements</code>
DescrDataTypeId	Integer	Id pobrane z metody <code>getSelfControlMeasurements</code> lub <code>getOrderedSelfControlMeasurements</code>
DescrDataId	Integer	Id pobrane z metody <code>getSelfControlMeasurements</code> lub <code>getOrderedSelfControlMeasurements</code>

DescrDataFormInstanceId	Integer	Id pobrane z metody getSelfControlMeasurements lub getOrderedSelfControlMeasurements
UserId	Integer	Identyfikator pacjenta

Wartość zwracana: Definicja formularza w formacie json

Nazwa	Typ	Opis
FormDefinition	String	Definicja formularza w formacie json

Definicja formularza w formacie json

```
[{
  "Form":{
    "ID":958, -- id formularza
    "FName":"Ciśnienie krwi", -- nazwa formularza
    "H":null,
    "TypeID":6, --typ formularza
    "Attributes":[ --lista atrybutów formularza
      {
        "ID":"8043", --id atrybutu
        "FName":"Skurczowe", --nazwa atrybutu
        "Validation":{ --dane do walidacji
          "LowerAlarm":100,
          "ID":5157,
          "LowerNorm":100,
          "UpperNorm":150,
          "Minimum":1,
          "UpperAlarm":150
        },
        "Values":[
          {
            "Descr":{
              "Value":null -- wartość atrybutu
            },
          }
        ],
        "Position":{ --pozycja w gridzie
          "W":1, --dlugosc pola
          "X":0, --numer kolumny
          "Y":1 --numer wiersz
        },
        "Required":0,
        "TypeID":3, --typ atrybutu
        "Dictionary":null,
      }
    ]
  }
}]
```

```

    },
  ],
  "AdditionalServices":0,
  "FormInstanceId":"6953",
},
}]

```

Typy atrybutów

- 1 – pole tekstowe typu input
- 2 – pole wyboru daty
- 3 – pole typu numer
- 4 – pole typu decimal
- 5 – pole typu TextArea
- 7 – pole typu label
- 9 – pole typu select
- 11 – pole typu checkBox
- 16 – pole typu radioGroup
- 21 – pole typu formatowany text

## 5.4 Zapisanie formularza dla wybranego pomiaru

Nazwa metody: saveFormMeasurement

Opis: Metoda zapisująca formularz do danego pomiaru

Parametry wejściowe:

Nazwa	Typ	Opis
DescrDataTypeSysFctId	Integer	Id pobrane z metody getSelfControlMeasurements lub getOrderedSelfControlMeasurements
DescrDataTypeFormDefId	Integer	Id pobrane z metody getSelfControlMeasurements lub getOrderedSelfControlMeasurements
DescrDataTypeId	Integer	Id pobrane z metody getSelfControlMeasurements lub getOrderedSelfControlMeasurements
DescrDataId	Integer	Id pobrane z metody getSelfControlMeasurements lub getOrderedSelfControlMeasurements
DescrDataFormInstanceId	Integer	Id pobrane z metody getSelfControlMeasurements lub getOrderedSelfControlMeasurements
UserId	Integer	Identyfikator pacjenta
Attributes	String	Wartosci z formularza w formacie json

Wartość zwracana: Brak

Wartości z formularza w formacie json

```
{
```

```

"8043"."125", -- ID atrybutu : Wartość
"8044"."75", -- ID atrybutu : Wartość
"8045"."67" -- ID atrybutu : Wartość
.....
}

```

## 5.5 Pobranie listy pomiarów z wypełnionymi formularzami

Nazwa metody: `getSelfControlFilledMeasurements`

Opis: Metoda zwraca listę pomiarów z wypełnionymi formularzami

Parametry wejściowe:

Nazwa	Typ	Opis
StartDt	Date	Data od kiedy chcemy pobrać pomiary
StopDt	Date	Data do kiedy chcemy pobrać pomiary
ServiceId	Integer	Identyfikator usługi (typ pomiaru)

Wartość zwracana: Lista obiektów `SelfControlFilledMeasurementDto`

Nazwa	Typ	Opis
OrderId	Integer	Identyfikator zlecenia (jeśli pomiar jest zleceniem)
PerfServiceId	Integer	Identyfikator powiązany z wykonywaną usługą
ServiceId	Integer	Identyfikator usługi
Name	String	Nazwa pomiaru
InsertDate	DateTime	Data+czas pomiaru
FilledForm	String	Wypełniony formularz w formacie json opisany w metodzie <code>getFormMeasurementDefinition</code>

## 6. Dokumentacja pacjenta

Pacjenci mogą dodawać do HIS'a pliki w postaci pdf, jpg. Mogą to być np. skany skierowań, wyników badań itp. Pliki te są później widoczne przez lekarzy i mogą być dołączone do dokumentacji medycznej pacjenta

### 6.1 Dodanie załącznika do HIS

Nazwa metody: `savePatientAttachment`

Opis: Metoda dodaje plik pacjenta do systemu HIS.

Parametry wejściowe: Obiekt `UserDocument`

Nazwa	Typ	Opis
-------	-----	------

UserId	Integer	Identyfikator użytkownika
DocumentType	String	Wartość zgodna ze standardem MIME określająca format przesyłanych danych
DocumentName	String	Nazwa pliku
Content	Binary in BASE64	Dokument zapisany w postaci binarnej zakodowany w Base64

Wartość zwracana: Brak

## 7. eWyniki

### 7.1 Pobranie listy wyników pacjenta

Nazwa metody: getPerfServiceDocuments

Opis: Metoda pobiera listę wyników pacjent z systemu HIS

Parametry wejściowe:

Nazwa	Typ	Opis
UserId	Integer	Identyfikator użytkownika

Wartość zwracana: Lista obiektów UserPerfServiceDocument

Nazwa	Typ	Opis
PerfServiceId	Integer	Identyfikator wyniku
PrintOutSysFctId	Integer	Identyfikator potrzebny do wydruku
PerfServiceDate	DateTime	Data+czas wykonania wyniku
ServiceId	Integer	Identyfikator usługi połączonej z wynikiem
Name	String	Nazwa usługi powiązanej z wynikiem
OrgUnitId	Integer	Id jednostki organizacyjnej
OrgUnitName	String	Nazwa jednostki organizacyjnej
StatusId	Integer	Id statusu wyników
EventId	Integer	Identyfikator świadczenia
EpisodeId	Integer	Identyfikator zestawu świadczeń
OrderId	Integer	Identyfikator zlecenia
VisitId	Integer	Identyfikator wizyty
isConsulted	Boolean	Czy wynik został już skonsultowany

### 7.2 Pobranie pliku wyniku

Nazwa metody: getPerfServiceDocumentContent

Opis: Metoda zwraca plik wyniku.

Parametry wejściowe:

Nazwa	Typ	Opis
UserId	Integer	Identyfikator pacjenta
PerfServiceId	Integer	Identyfikator wyniku
PrintOutSysFctId	Integer	Identyfikator potrzebny do wydruku
EventId	Integer	Identyfikator świadczenia
EpisodeId	Integer	Identyfikator zestawu świadczeń
OrderId	Integer	Identyfikator zlecenia
VisitId	Integer	Identyfikator wizyty

Wartość zwracana:

Nazwa	Typ	Opis
Content	Binary in BASE64	Dokument zapisany w postaci binarnej zakodowany w Base64