

一般口演

## 一般口演8

## 病院情報システム3（オーダ・部門システム・指示等）

2018年11月23日(金) 16:00～17:30 G会場(5F 504+505)

## [2-G-3-2] Web APIの構築による電子カルテ情報の取得と利用の効率化の試み

○國方 淳<sup>1</sup>, 谷川 雅俊<sup>2</sup>, 横井 英人<sup>1,2</sup> (1.香川大学医学部附属病院 臨床研究支援センター, 2.香川大学医学部附属病院 医療情報部)

1.背景と目的 近年、多くのサービスやアプリケーションにおいて、それらの機能を外部から利用する際に Web APIが用いられている。Web APIは汎用的な HTTPプロトコルによってリクエストした結果を JSONなどの処理しやすい形のレスポンスデータとして返すことにより、柔軟なデータ処理を可能としている。近年は Web APIを機能として持つ電子カルテも増えてきているが、当院の電子カルテには実装されていないため、独自の Web APIの構築を試みた。2.方法 Webサーバーに Windows Server 2012 R2、プログラミング言語に PHPを用いて Web APIを構築した。Webサーバー上の PHPプログラムは ODBC(Open Database Connectivity)により電子カルテのレプリカデータベースにアクセスし、取得したデータは一定のフォーマットの JSON形式で返却する仕様とした。またセキュリティを考慮し、通信を HTTPSに限定するとともにデータの取得にはパスワードが必要とされるようにした。3.結果 電子カルテネットワーク上の機器から、単純な HTTPリクエストによって電子カルテ情報を取得することが可能となった。これを利用し、内製の Webアプリケーションシステムが電子カルテから患者の氏名や生年月日等の基本情報を取得する機能や、治験中の患者の入退院履歴を監視して予定外の入院時にアラートを出すなどの機能が実装できた。4.考察と結論 Web APIの実装により、APIで利用可能なデータの範囲であればプログラムからの直接利用が可能となり、特に電子カルテを利用する医療機関が独自のデータ処理を行いたい場合に有用である。また、今回は独自仕様の Web APIを構築したが、電子カルテのベンダーに依存しない共通規格の Web APIが存在すれば、同じプログラムを電子カルテの種類によらず適応することができ、公共の利益にも寄与するものと考えられる。

## Web API の構築による電子カルテ情報の取得と利用の効率化の試み

國方 淳<sup>\*1</sup>、谷川 雅俊<sup>\*2</sup>、横井 英人<sup>\*1,2</sup>

\*1 香川大学医学部附属病院 臨床研究支援センター

\*2 香川大学医学部附属病院 医療情報部

### Efficient information retrieval from electronic health records with Web API and its use

Jun Kunikata<sup>\*1</sup>, Masatoshi Tanigawa<sup>\*2</sup>, Hideto Yokoi<sup>\*1,2</sup>

\*1 Clinical Research Support Center, Kagawa University Hospital

\*2 Department of Medical Informatics, Kagawa University Hospital

Recently, many services and applications have Web API for making use of their data from external systems. Web API allows programmers to access data with general HTTP requests and to receive them as machine readable format like JSON. Although some EHR systems have Web API, EHR clients, mostly medical institutions, generally can't use the API easily for their own purposes. Therefore, we tried implementing our own Web API.

We used Windows Server as web server and PHP as programming language to implement our Web API. The API receives HTTP requests from clients and retrieves necessary data from EHR database with SQL, then returns retrieved data to clients as JSON format.

As a result of implementing our Web API, it became easy to retrieve several EHR data from the database and we have been able to use them in some applications efficiently. If there is standardized vendor-free Web API, external systems can use same interface to connect various EHR systems and it will bring about great benefit for many EHR clients.

Keywords: Web API, EHR, HTTP, JSON

#### 1. 背景と目的

近年、医療機関では電子カルテシステムの導入が進んでおり、数多くの診療情報が電子的に保存されている状況となっている。保存されたデータは電子カルテシステム内だけで利用されるものではなく、部門システムとの連携や医療機関独自のデータ利用のためにシステム外からのアクセスが必要となる場合も多い。

しかし、当院の電子カルテシステムでは部門システムが電子カルテシステムと連携しようとする場合、基本的にその部門システム専用のインターフェースの開発が必要であり、ごく限られたデータの授受についてもカスタマイズ費用が発生するのが実情である。

また、医療機関が独自のデータ参照機能を実現する手段としてはDWHの利用が最も一般的であり、DWHに格納されたデータを効率よく活用するためのBIツール等が開発されている他、DWHのデータを内製のアプリケーションにより活用する試みもなされている<sup>1)</sup>。しかし、DWHはデータの蓄積を目的としているため集計・解析に利用する際には非常に有用であるが、一般には電子カルテの現在の状態が反映されるまでにタイムラグが存在するため、診療業務で用いるためにデータを取得するという目的には適さない場合がある。

一方で、近年はAmazon, Facebook, Twitter等をはじめとするWebサービスだけでなく、業務システムにおいてもWeb APIによるデータ交換が一般的となってきている。Web APIの定義は厳密には定まっていないが、「HTTP (Hypertext Transfer Protocol)を利用してネットワーク越しに呼び出すAPI (Application Programming Interface)」のように定義され、内部の実装を知ることなく外部インターフェースを介してデータを利用できる仕組みを指す<sup>2)</sup>。Web APIはHTTPを利用するため汎用性が高く、多くのプログラミング言語で容易に利用が可能である。

しかし、現状ではWeb APIを備えた電子カルテシステムは少なく、また実装されていても医療機関が自由に利用できるものではない。そこで今回我々は電子カルテシステムの情報をデータベースから直接取得するWeb APIの構築を試みた。

#### 2. システム概要

##### 2.1 システム構成

本システムは電子カルテシステムが稼働している医療情報ネットワーク上のWebサーバ(Windows Server 2012 R2)上に構築した。その概要図を図1に示す。

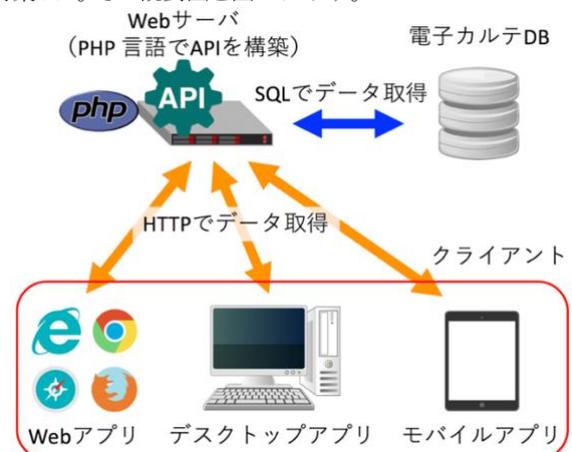


図1 システム概要図

クライアントはWebサーバからHTTPでデータを取得する。バックエンドではWebサーバが電子カルテDBからSQLでデータを取得するが、クライアントはこの部分を意識する必要はない。

本システムの Web API は PHP 言語で記述されたプログラムである。このプログラムはクライアントから HTTP リクエストを受け取り、その内容を元に電子カルテ DB に SQL を発行して必要なデータを取得する。取得したデータは JSON 形式のレスポンスとしてクライアントに送信される。

なお、電子カルテシステムのパフォーマンスへの影響を考慮し、Web API が実際にデータを取得する DB はレプリカ DB であるが、本系の DB とデータが同期されているため図 1 では単に電子カルテ DB とした。また、電子カルテ DB へのアクセスは読み取り専用である。

## 2.2 Web API の仕様

今回構築した Web API は実装実験が主目的であるため、レスポンスは必要最小限の構成となっている。また、現時点で実装しているリソースは網羅的なものではなく、アプリケーションでデータを取得する必要が生じた際に順次リソースの追加を行なっている。

### 2.2.1 リクエストの仕様

リクエストの URI の基本的な形式を図 2 に示す。セキュリティを考慮し、暗号化通信である HTTPS のみを許可し通常の HTTP で接続された場合はエラーを返す。本 API はデータの取得専用のため GET メソッドのみを使用する。

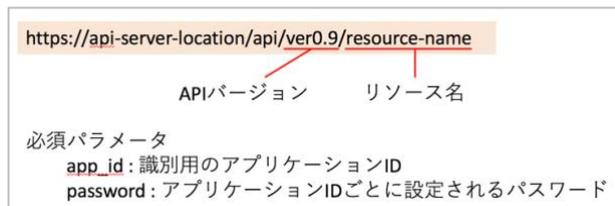


図 2 リクエストの基本的な形式

### 2.2.2 レスポンスの仕様

レスポンスは JSON 形式で返却される。レスポンスの基本的な形式を図 3 に示す。

```
{
  'app_id': '**** アプリケーションID ****',
  'data': {
    **** リソース特有のデータ ****
  },
  'description': '**** レスポンスの説明 ****',
  'requested_datetime': '**** リクエスト時刻 ****',
  'requested_parameters': {**** リクエストパラメータ ****},
  'resource_name': '**** リソース名 ****',
  'status': '**** レスポンスのステータス ****'
}
```

図 3 レスポンスの基本的な形式

### 2.2.3 リクエストとレスポンスの例

図 4 に患者情報を取得する場合、図 5 に直近の検査値 (CRTN:クレアチニン) を取得する場合のリクエストとレスポンスの例を示す。リクエストは一般的な Web ブラウザの URL として入力する形式で示している。また、図の説明文では必須パラメータを省略している。

#### Request

```
https://10.100.100.100/api/ver0.9/patient-info?pid=999999&app_id=test_app&password=*****
```

#### Response

```
{
  'app_id': 'test_app',
  'data': {
    'birth_date': '2012-08-10',
    'gender': 'M',
    'kana_name': 'テスト カンジヤ 0 0 1',
    'name': 'テスト 患者 0 0 1',
    'pid': '999999'
  },
  'description': 'Data found',
  'requested_datetime': '2018-08-10 13:55:29',
  'requested_parameters': {'pid': '999999'},
  'resource_name': 'patient-info',
  'status': 'success'
}
```

図 4 患者 ID から患者情報を取得するリクエストとレスポンス  
リソース名: patient-info  
パラメータ: pid (患者 ID)

#### Request

```
https://10.100.100.100/api/ver0.9/latest-labo-data?pid=999999&kensa_code=0000011500&app_id=test_app&password=*****
```

#### Response

```
{
  'app_id': 'test_app',
  'data': {
    'date': '2018-06-11',
    'labo_data': {
      'comment': '',
      'kensa_code': '0000011500',
      'name': 'CRTN',
      'normal_lower_limit': '0.70',
      'normal_upper_limit': '1.30',
      'pid': '999999',
      'type': '一般生化・血液',
      'unit': 'mg/dl',
      'value': '0.66'
    },
    'order_no': '99999999'
  },
  'description': 'Data found',
  'requested_datetime': '2018-07-05 15:25:18',
  'requested_parameters': {
    'pid': '999999',
    'kensa_code': '0000011500'
  },
  'resource_name': 'latest-labo-data',
  'status': 'success'
}
```

図 5 患者の直近の検査値を取得するリクエストとレスポンス  
リソース名: latest-labo-data  
パラメータ: pid (患者 ID), kensa\_code (検査コード)

## 3. システム評価

Web API の利用により、それまではデータベースの構造を把握した上で SQL を発行して取得していたデータを単純な

HTTP リクエストで取得できるようになった。以下に、本 API の利用例を示す。

### 3.1 外部システムでの患者登録

当院の臨床研究支援センターで利用している内製の治験進捗管理システムにおいて患者情報の入力ミスを防止するために、被験者登録の患者 ID 入力時に API で患者情報を取得し、患者氏名・性別を自動入力する機能を実装した。このシステムの被験者登録画面を図 6 に示す。

カルテ番号	111111111	入力を確認	キャンセル
氏名	テスト患者	入力を確認	キャンセル
カナ氏名	テストカンジャ	入力を確認	キャンセル
性別	<input checked="" type="radio"/> 男性 <input type="radio"/> 女性 <input type="radio"/> その他		未設定

図 6 治験進捗管理システムの被験者登録画面  
カルテ番号(患者 ID)を入力すると氏名と性別が自動入力される。API で患者 ID から患者情報を取得している。

### 3.2 患者の入退院状況チェック

治験に参加中の被験者の入退院情報、特に休日に緊急入院があったかどうかは有害事象の報告の観点から重要な情報である。そこで上記の治験進捗管理システムにおいて、入退院情報を API で取得して入院中の被験者を一覧表示するとともに、治験に参加中の被験者の入退院があった場合に担当者へ自動的に電子メールで通知する機能を実装した。入院中の被験者一覧画面を図 7 に示す。

カルテID	氏名	治験略称	被験者番号	入院日
950900	志保 敬雄	CR0142 薬物試験	95090001	2018年08月17日
1400022	滝本 正広	CR0142 薬物試験	95090002	2018年08月14日
954905	藤田 隆三	CR0142 薬物試験	95090003	2018年08月10日
380801	藤田 隆三	CR0142 薬物試験	95090004	2018年08月04日
954906	藤田 隆三	CR0142 薬物試験	95090005	2018年07月27日
380807	志保 敬雄	CR0142 薬物試験	95090006	2018年07月22日
950908	志保 敬雄	CR0142 薬物試験	95090007	2018年07月20日
380820	志保 敬雄	CR0142 薬物試験	95090008	2018年07月04日
950909	滝本 正広	CR0142 薬物試験	95090009	2018年06月27日
207888	志保 敬雄	CR0142 薬物試験	95090010	2018年06月21日

図 7 治験進捗管理システムの入院患者一覧画面  
入院中の被験者を一覧表示する。API で入退院歴を取得している。

### 3.3 腎機能低下患者の禁忌薬チェック

当院ではインシデントをきっかけに、腎機能障害のある患者に対して禁忌薬が処方されたことをチェックする方法を検討した。その結果、処方情報と直近の検査値(血清クレアチニンと eGFR)を API で取得することにより、禁忌薬マスタで指定されている医薬品が腎機能低下患者に投与されているかどうかを全ての処方についてチェックする Web アプリケーションを作成した。作成したアプリケーションの画面を図 8 に示す。対象とする検査はリクエストパラメータの検査コードで指定するため、禁忌薬マスタと検査コードを変更すれば肝機能障害などのチェックにも利用が可能である。

開始日	処方情報	薬剤名	用量	処方状況	検査値
2018-08-09	テナリア錠	2.0mg	3969015F1029	処方済	eGFR: 40.4 ml/min (採取日: 2018-05-10)
2018-08-09 13:42:01	テナリア錠	2.0mg	3969015F1029	処方済	
2018-08-09	ジヤスビア錠	1.0 mg	3969010F3037	処方済	
2018-08-09	グリメピリド錠	1mg (トーフ)	3961008F1250	処方済	
2018-08-09	ロザリド配合錠 L D (トーフ)		2149110F1279	処方済	
2018-08-09	フロセミド錠	4.0 mg (デバ)	2139005F2440	処方済	CRTN: 1.99 mg/dl (採取日: 2018-05-29)
2018-08-09	バイアスピリン錠	1.0 mg	3399007H1021	処方済	eGFR: 25.2 ml/min (採取日: 2018-05-29)
2018-08-09	クロピドグレル錠	7.5 mg SANIK	3399008F2110	処方済	
2018-08-09 13:41:49	オメプラゾール錠	10 (SW)	2329022H2090	処方済	
2018-08-09	アーカメイトゼリー	2.0% 2.5 g	2190016Q2026	処方済	
2018-08-09	マグミット錠	3.0 mg	2344009F2031	処方済	
2018-08-09	4.0 mg ニフェジミン C R 錠	ザハ	2171014G5068	処方済	

図 8 腎機能低下患者の禁忌薬チェックアプリ画面  
腎機能が低下している患者に禁忌薬が処方されている場合に警告を表示する。処方情報と直近の検査値を API で取得して判定を行なっている。

## 4. 考察

API のリソースを新たに実装する際には、Web サーバがデータベースからデータを取得するための SQL を定義する必要があるため、ある程度の工数が必要であった。しかし、一度リソースを実装すれば以降は同じリソースを取得する際に Web サーバに送信する HTTP リクエストのみを考慮すればよいため、再利用性の高い API を構築することにより大きなメリットが得られる。

また、電子カルテシステムのバージョンアップやベンダーの変更によりシステムの仕様に変更が生じたとしても、変更前と同じリソースを実装することができればクライアント側のプログラムには変更が不要であり、システムの変更に対する柔軟性も向上する。より理想的には、電子カルテシステムのバージョンやベンダーに依存しない共通規格の Web API が存在すれば、部門システムとの接続にかかる工数・コストの削減にも繋がるであろう。

しかし、電子カルテシステムのデータベースは一般には外部からのデータ取得をそれほど考慮していないため、データベースの構成によっては API で取得したいデータを効率よく取得できない場合があることには注意が必要である。

今回は電子カルテシステムのデータベースからデータを取得したが、即時性が問題とならないデータについては取得先を DWH や SS-MIX2<sup>3)</sup>等とする事も可能である。このように、リソースごとにデータの取得先が異なる場合でも、統一的なインターフェースを用いることができる点もメリットといえる。

## 5. 結論

Web API の構築により、医療機関にとって利活用の自由度の高いデータを単純な HTTP リクエストで取得することが可能となった。Web API が普及してきている近年の状況からも、電子カルテシステムの種類に依存しない標準化された Web API の策定が期待される。

## 参考文献

- 1) 岡本康幸. Data Warehouse を活用するための内製アプリケーションの開発. 医療情報学 2015; 35(5): 239-248.
- 2) 水野貴明. Web API The Good Parts. オライリー・ジャパン, 2014.
- 3) SS-MIX2 標準化ストレージ 構成の説明と構築ガイドライン Ver.1.2d, 2017.

[[https://www.jami.jp/jamistd/docs/SS-MIX2/SS-MIX2-V1.2d-1/SS-MIX2\\_Ver.1.2d\\_released\\_20170116/SS-MIX2\\_stdstrglVer.1.2d\\_20170116.pdf](https://www.jami.jp/jamistd/docs/SS-MIX2/SS-MIX2-V1.2d-1/SS-MIX2_Ver.1.2d_released_20170116/SS-MIX2_stdstrglVer.1.2d_20170116.pdf)].

