

Số: 366 /BHXH-CNTT
V/v bổ sung hàm tra cứu trên Hệ thống
thông tin giám định bảo hiểm y tế

Hà Nội, ngày 31 tháng 01 năm 2018

Kính gửi: - Bảo hiểm xã hội các tỉnh, thành phố trực thuộc Trung ương;
- Bảo hiểm xã hội Bộ Quốc phòng;
- Bảo hiểm xã hội Công an Nhân dân.
(Sau đây gọi chung là BHXH các tỉnh)

Căn cứ Quyết định số 917/QĐ-BHXH ngày 20/6/2016 của Tổng Giám đốc BHXH Việt Nam về Ban hành Công tiếp nhận dữ liệu Hệ thống thông tin Giám định Bảo hiểm y tế phiên bản 2.0. Để hỗ trợ cơ sở khám, chữa bệnh (cơ sở KCB) trong việc tra cứu tự động giá trị sử dụng thẻ bảo hiểm y tế (BHYT) có kết quả đồng nhất với kết quả tra cứu trực tiếp trên Hệ thống thông tin Giám định bảo hiểm y tế và tự động tra cứu thông tin cơ bản về cơ sở KCB trong danh mục cơ sở KCB. BHXH Việt Nam đã bổ sung 02 hàm tra cứu gồm: **NhanThongTinCSKCB** và **KQNhanLichSuKCB2018** (hướng dẫn chi tiết tại phụ lục kèm theo).

Đề nghị BHXH các tỉnh thông báo đến các cơ sở y tế có ký hợp đồng khám chữa bệnh với cơ quan BHXH để các cơ sở y tế chủ động trong việc điều chỉnh phần mềm quản lý tại bệnh viện. Trong quá trình thực hiện, nếu có vướng mắc kịp thời báo cáo về BHXH Việt Nam để xem xét, giải quyết. /

Nơi nhận:

- Như trên;
- Bộ Y tế;
- Tổng Giám đốc (để b/c);
- Các Phó Tổng Giám đốc;
- Các đơn vị: VP, TT, CSYT, GĐB, GĐN;
- Lưu: VT, CNTT.

**KT. TỔNG GIÁM ĐỐC
PHÓ TỔNG GIÁM ĐỐC**



Phạm Lương Sơn



PHỤ LỤC

(Kèm theo công văn số 366 /BHXH-CNTT ngày 31 / 01/ 2018 của BHXH Việt Nam)

A. HÀM NHẬN THÔNG TIN CƠ SỞ KHÁM CHỮA BỆNH

1. Mô tả

Request URL	HMAC http://egw.baohiemxahoi.gov.vn/api/egw/NhanThongTinCSKCB
Request Method	POST
Request Header	Content-Type: application/json
Query Parameters	username={username}&password={password}&token={token}&id_token={id_token}&macskcb={ macskcb } (Tham khảo mục 2)
Form data	

Cơ sở khám chữa bệnh kiểm tra tính đúng đắn của thông tin thẻ của bệnh nhân.

2. Thông điệp yêu cầu

Dữ liệu đầu vào	Tên Field	Kiểu dữ liệu	Ghi chú
Mã cơ sở KCB	macskcb	Chuỗi ký tự	Mã cơ sở khám chữa bệnh cần tra cứu
Tên đăng nhập	username	Chuỗi ký tự	Tài khoản đăng nhập
Mật khẩu	password	Chuỗi ký tự	Mật khẩu đăng nhập
Token	token	Chuỗi ký tự	Token được trả về từ trước
TokenID	id_token	Chuỗi ký tự	Token ID được trả về ở hàm trước

3. Thông điệp trả lời

Cấu trúc:

- Header status:
- Header Content-Type:application/json;charset=UTF-8
- Json data

Trong đó

- Có hai khả năng đối với thông điệp trả lời: Trường hợp có lỗi trong quá trình xử lý hoặc trường hợp xử lý thành công.
- Trường hợp lỗi: Xem danh mục lỗi sẽ được gửi trả lại.

Trường	Kết quả
Header status:500	An unexpected error occurred

- Trường hợp thành công: Hệ thống trả về:

Trường	Kết quả
Header status	200
Header Content-	application/json;charset=UTF-8

Type	
Json data	<p>Trả về là object KQNhanThongTinCSKCB bao gồm các thuộc tính sau:</p> <ul style="list-style-type: none"> + maketqua: Trả về mã kết quả giao dịch (Thành công, lỗi) + message : Thông báo + macskcb : Mã cơ sở khám chữa bệnh + ten : Tên cơ sở + hang : Hạng bệnh viện (0-đặc biệt,1-hạng 1,2-hạng 2,3-hạng3,4-hạng 4,5-chưa phân hạng) + tuyen : Tuyển chuyên môn kỹ thuật (1-trung ương,2-tỉnh,3-huyện,4-xã) + diachi : Địa chỉ + tenchuquan : Tên cơ quan chủ quản + ngaysua : Ngày sửa + macha : Mã cơ sở cha + hieuluc : Hiệu lực (1 - có,0 - không) + ngayngunghoatdong : Ngày ngừng hoạt động + matinh : Mã tỉnh + mahuyen : Mã huyện

Mô tả mã kết quả trả về:

Mã kết quả	Mô tả
200	Nhận thông tin thành công
205	Sai định dạng tham số truyền vào
206	Không tìm thấy thông tin cơ sở
401	Lỗi xác thực tài khoản
500	Lỗi server

4. Mẫu hàm ứng dụng

- Code mẫu c#

```
private static void TestNhanThongTinCSKCB()
{
    using (var client = new HttpClient())
    {
        client.BaseAddress = new Uri(serviceUrl);
        client.DefaultRequestHeaders.Accept.Clear();
        client.DefaultRequestHeaders.Accept.Add(new
MediaTyepWithQualityHeaderValue("application/json"));
        string username = username;
        string password = password; // dạng MD5
        ApiToken input = new ApiToken { username = username, password =
password };
        var values = new Dictionary<string, string>
        {
            { "username", username },
            { "password", password }
        };
        var content = new FormUrlEncodedContent(values);
```

```

        HttpResponseMessage response = client.PostAsync("api/token/take",
content).Result;
        if (response.IsSuccessStatusCode)
        {
            KQPhienLamViec plv =
response.Content.ReadAsAsync<KQPhienLamViec>().Result;
            var key = plv.APIKey;

            //HTTP POST
            string data2 =
string.Format("username={0}&password={1}&token={2}&id_token={3}&macskcb={4}",
username, password, key.access_token, key.id_token, macskcb);
            HttpResponseMessage response2 =
client.PostAsJsonAsync("api/egw/NhanThongTinCSKCB?" + data2, new { }).Result;
            if (response2.IsSuccessStatusCode)
            {
                string result = response2.Content.ReadAsStringAsync().Result;
                try
                {
                    var kqua =
(KQNhanThongTinCSKCB)JsonConvert.DeserializeObject<KQNhanThongTinCSKCB>(result);
                    if (kqua.maketqua == "200")
                    {
                        //
                    }
                }
                catch (Exception ex)
                {
                    //throw;
                }
            }
        }
    }
}

```

- Code mẫu Java

```

@SuppressWarnings("unchecked")
public Map<String,String> getToken(String user, String pass) {
    try {
        StringBuilder uri = new StringBuilder();
        HttpHeaders headers = new HttpHeaders();
        headers.set("Accept", MediaType.APPLICATION_JSON_VALUE);
        uri.append("http://egw.baohiemxahoi.gov.vn/api/token/take");
        Map<String, String> dataTk = new HashMap<>();
        dataTk.put("username", user);
        dataTk.put("password", pass.toUpperCase());
        String body = new Gson().toJson(dataTk);
        HttpResponseMessage response = post(uri.toString(), body);
        ObjectMapper objectMapper = new ObjectMapper();
        String json_string =
EntityUtils.toString(response.getEntity());
        Map<String,Object> result =
objectMapper.readValue(json_string, Map.class);
        Map<String,String> apiKey = (Map<String,
String>)result.get("APIKey");
        if(Objects.nonNull(apiKey)){
            return apiKey;
        }
    } catch (UnsupportedOperationException | IOException e) {
        e.printStackTrace();
    }
}

```

```

        return null;
    }

    @SuppressWarnings("unchecked")
    @Override

    @SuppressWarnings("unchecked")
    @Override
    public void nhanThongTinCSKcbPost() throws org.apache.http.ParseException,
    IOException {
        String uid = user name;
        String pwd = password;
        String macskcb = Mã cơ sở khám chữa bệnh;
        Map<String,String> akiKey = getToken(uid, pwd);
        if(Objects.nonNull(akiKey)){
            StringBuilder uri = new StringBuilder();
            HttpHeaders headers = new HttpHeaders();
            headers.set("Accept", MediaType.APPLICATION_JSON_VALUE);

            uri.append("http://egw.baohiemxahoi.gov.vn/api/egw/NhanThongTinCSKCB?");
            uri.append("token=" + akiKey.get("access_token"));
            uri.append("&id_token=" + akiKey.get("id_token"));
            uri.append("&username=" + uid);
            uri.append("&password=" + pwd.toUpperCase());
            uri.append("&macskcb=" + macskcb);
            Map<String, String> dataRequest = new HashMap<>();
            String body = new Gson().toJson(dataRequest);
            HttpResponse response = post(uri.toString(), body);
            ObjectMapper objectMapper = new ObjectMapper();
            String json_string =
            EntityUtils.toString(response.getEntity());
            Map<String,Object> result =
            objectMapper.readValue(json_string, Map.class);
            KQNhanThongTinCSKCB kq = objectMapper.convertValue(result,
            KQNhanThongTinCSKCB.class);
            System.out.println(kq);
        }
    }
}

```

- Class hỗ trợ

```

public class ApiToken
{
    [DataMember]
    public string username { get; set; }
    [DataMember]
    public string password { get; set; }
}

public class KQPhienLamViec
{
    public string maKetQua { get; set; }
    public ApiKey APIKey { get; set; }
}

public class ApiKey
{
    [DataMember]
    public string access_token { get; set; }
    [DataMember]
    public string id_token { get; set; }
}

```

```

        [DataMember]
        public string token_type { get; set; }
        [DataMember]
        public string username { get; set; }
        [DataMember]
        public DateTime expires_in { get; set; }
    }

    public class KQNhanThongTinCSKCB
    {
        public string maketqua { get; set; }
        public string message { get; set; }
        public string macskcb { get; set; }
        public string ten { get; set; }
        public string hang { get; set; }
        public string tuyen { get; set; }
        public string diachi { get; set; }
        public string tenchuquan { get; set; }
        public string ngaysua { get; set; }
        public string macha { get; set; }
        public short? hieuluc { get; set; }
        public string ngayngunghoatdong { get; set; }
        public string matinh { get; set; }
        public string mahuyen { get; set; }
    }

```


B. BỔ SUNG HÀM KIỂM TRA LỊCH SỬ KHÁM CHỮA BỆNH

1. Mô tả

Request URL	HMAC http://egw.baohiemxahoi.gov.vn/api/egw/NhanLichSuKCB2018
Request Method	POST
Request Header	Content-Type: application/json
Query Parameters	token={token}&id_token={id_token}&username={username}&password={password} (Tham khảo mục 2)
Form data	Đối tượng “theBHYT2018” bao gồm { “maThe”: { mã thẻ }, “hoTen”: { họ tên }, “ngaySinh”: { ngày sinh } } (Tham khảo mục 2)

Cơ sở khám chữa bệnh kiểm tra tính đúng đắn của thông tin thẻ của bệnh nhân.

2. Thông điệp yêu cầu

Dữ liệu đầu vào	Tên Field	Kiểu dữ liệu	Ghi chú
Mã thẻ	maThe	Chuỗi ký tự	Mã thẻ
Họ tên	hoTen	Chuỗi ký tự	Họ tên chủ thẻ
Ngày sinh	ngaySinh	Chuỗi ký tự	Ngày sinh (DD/MM/YYYY) /Trường hợp chỉ có năm sinh (YYYY).
Tên đăng nhập	username	Chuỗi ký tự (an 5)	Tài khoản đăng nhập
Mật khẩu	password	Chuỗi ký tự (an 6..10)	Mật khẩu đăng nhập
Token	token	Chuỗi ký tự	Token được trả về từ trước
TokenID	id_token	Chuỗi ký tự	Token ID được trả về ở hàm trước

3. Thông điệp trả lời

Cấu trúc:

- Header status:
- Header Content-Type:application/json;charset=UTF-8
- Json data

Trong đó

- Có hai khả năng đối với thông điệp trả lời: Trường hợp có lỗi trong quá trình xử lý hoặc trường hợp xử lý thành công.
- Trường hợp lỗi: Xem danh mục lỗi sẽ được gửi trả lại.

Trường	Kết quả
Header status:500	An unexpected error occurred

- Trường hợp thành công: Hệ thống trả về:

Trường	Kết quả
Header status	200
Header Content-Type	application/json;charset=UTF-8
Json data	<p>Trả về là object KQNhanLichSuKCB2018 bao gồm các thuộc tính sau:</p> <ul style="list-style-type: none"> + maKetQua: Trả về mã kết quả giao dịch (Thành công, lỗi) + ghiChu : Ghi chú + maThe : Mã thẻ hiện tại + hoTen : Họ tên + ngaySinh : ngày sinh + gioiTinh : giới tính + diaChi : Địa chỉ + maDKBD : Mã đăng kí ban đầu + cqBHXH : Tên cơ quan bảo hiểm xã hội + gtTheTu : Thời gian thẻ có giá trị sử dụng + gtTheDen : Hạn thẻ + maKV : Mã khu vực + ngayDu5Nam : Ngày đủ năm năm liên tục + maSoBHXH : Mã số + maTheCu : Mã thẻ cũ + maTheMoi : Mã thẻ mới + gtTheTuMoi : Thời gian thẻ mới có giá trị sử dụng + gtTheDenMoi : Hạn thẻ mới + dsLichSuKCB2018: Danh sách object lịch sử khám chữa bệnh, mỗi object bao gồm (= null nếu thông tin thẻ không chính xác) <pre> { - maHoSo: mã hồ sơ, để tra cứu thông tin chi tiết - maCSKCB: mã cơ sở khám chữa bệnh , - ngayVao: ngày vào (định dạng yyyyMMddhhmm), - ngayRa: ngày ra (định dạng yyyyMMddhhmm), - tenBenh: tên bệnh , - tinhTrang: tình trạng ra viện, - kqDieuTri: kết quả điều trị } </pre> <ul style="list-style-type: none"> + dsLichSuKT2018: Danh sách object lịch sử tra cứu thẻ, mỗi object bao gồm (= null nếu thông tin thẻ không chính xác) <pre> { - userKT: user tra cứu - thoiGianKT: thời gian tra cứu (định dạng yyyyMMddhhmm), - thôngBao: thông báo - maLoi : mã lỗi } </pre>

Mô tả mã kết quả trả về:

Mã kết quả	Mô tả
000	Thẻ còn giá trị sử dụng
001	Thẻ do BHXH Bộ Quốc Phòng quản lý, đề nghị kiểm tra thẻ và thông tin giấy tờ tùy thân
002	Thẻ do BHXH Bộ Công An quản lý, đề nghị kiểm tra thẻ và thông tin giấy tờ tùy thân
003	Thẻ cũ hết giá trị sử dụng nhưng đã được cấp thẻ mới
004	Thẻ cũ còn giá trị sử dụng nhưng đã được cấp thẻ mới
010	Thẻ hết giá trị sử dụng
051	Mã thẻ không đúng
052	Mã tỉnh cấp thẻ(kí tự thứ 4,5 của mã thẻ) không đúng
053	Mã quyền lợi thẻ(kí tự thứ 3 của mã thẻ) không đúng
050	Không thấy thông tin thẻ bhyt
060	Thẻ sai họ tên
061	Thẻ sai họ tên(đúng kí tự đầu)
070	Thẻ sai ngày sinh
100	Lỗi khi lấy dữ liệu sổ thẻ
101	Lỗi server
110	Thẻ đã thu hồi
120	Thẻ đã báo giảm
121	Thẻ đã báo giảm. Giảm chuyển ngoại tỉnh
122	Thẻ đã báo giảm. Giảm chuyển nội tỉnh
123	Thẻ đã báo giảm. Thu hồi do tăng lại cùng đơn vị
124	Thẻ đã báo giảm. Ngừng tham gia
130	Trẻ em không xuất trình thẻ
205	Lỗi sai định dạng tham số truyền vào
401	Lỗi xác thực tài khoản

4. Mẫu hàm ứng dụng

- Code mẫu c#

```
private static void TestNhanLichSuKCBBS()
{
    using (var client = new HttpClient())
    {
        //string serviceUrl = "http://egw.baohiemxahoi.gov.vn/";
        client.BaseAddress = new Uri(serviceUrl);
        client.DefaultRequestHeaders.Accept.Clear();
        client.DefaultRequestHeaders.Accept.Add(new
        MediaTypeWithQualityHeaderValue("application/json"));
        string username = user cơ sở (xxxxx_BV);
        string password = pass đã mã hóa MD5;
        ApiToken input = new ApiToken { username = username, password =
password };
        var values = new Dictionary<string, string>
        {
            { "username", username },
            { "password", password }
        };
    }
}
```

```

        var content = new FormUrlEncodedContent(values);
        //var data = string.Format("username={0}&password={1}", username,
password);
        HttpResponseMessage response = client.PostAsync("api/token/take",
content).Result;
        if (response.IsSuccessStatusCode)
        {
            KQPhienLamViec plv =
response.Content.ReadAsAsync<KQPhienLamViec>().Result;
            var key = plv.APIKey;

            //HTTP POST
            string data2 =
string.Format("token={0}&id_token={1}&username={2}&password={3}", key.access_token,
key.id_token, username, password);
            HttpResponseMessage response2 =
client.PostAsJsonAsync("api/egw/NhanLichSuKCB2018?" + data2,
new ApiTheBHYT2018 { maThe = mã thẻ ,hoTen = họ tên ,
ngaySinh = ngày sinh }).Result;
            if (response2.IsSuccessStatusCode)
            {
                string result = response2.Content.ReadAsStringAsync().Result;
                try
                {
                    var kqua =
(KQNhanLichSuKCBBS)JsonConvert.DeserializeObject<KQNhanLichSuKCBBS>(result);
                }
                catch (Exception ex)
                {
                    //throw;
                }
            }
        }
    }
}

```

- Code mẫu JAVA:

```

@SuppressWarnings("unchecked")
public Map<String,String> getToken(String user, String pass) {
    try {
        StringBuilder uri = new StringBuilder();
        HttpHeaders headers = new HttpHeaders();
        headers.set("Accept", MediaType.APPLICATION_JSON_VALUE);
        uri.append("http://egw.baohiemxahoi.gov.vn/api/token/take");
        Map<String, String> dataTk = new HashMap<>();
        dataTk.put("username", user);
        dataTk.put("password", pass.toUpperCase());
        String body = new Gson().toJson(dataTk);
        HttpResponseMessage response = post(uri.toString(), body);
        ObjectMapper objectMapper = new ObjectMapper();
        String json_string =
EntityUtils.toString(response.getEntity());
        Map<String,Object> result =
objectMapper.readValue(json_string, Map.class);
        Map<String,String> apiKey = (Map<String,
String>)result.get("APIKey");
        if(Objects.nonNull(apiKey)){
            return apiKey;
        }
    } catch (UnsupportedOperationException | IOException e) {
        e.printStackTrace();
    }
}

```

```

    }
    return null;
}

@SuppressWarnings("unchecked")
@Override
public void getDataPost() throws org.apache.http.ParseException,
IOException {
    String uid = username;
    String pwd = password;
    Map<String,String> akiKey = getToken(uid, pwd);
    if(Objects.nonNull(akiKey)){
        StringBuilder uri = new StringBuilder();
        HttpHeaders headers = new HttpHeaders();
        headers.set("Accept", MediaType.APPLICATION_JSON_VALUE);

        uri.append("http://egw.baohiemxahoi.gov.vn/api/egw/NhanLichSuKCB2018?");
        uri.append("token=" + akiKey.get("access_token"));
        uri.append("&id_token=" + akiKey.get("id_token"));
        uri.append("&username=" + uid);
        uri.append("&password=" + pwd.toUpperCase());
        Map<String, String> dataRequest = new HashMap<>();
        dataRequest.put("maThe", mã thẻ);
        dataRequest.put("hoTen", họ tên);
        dataRequest.put("ngaySinh", ngày sinh);
        String body = new Gson().toJson(dataRequest);
        HttpResponse response = post(uri.toString(), body);
        ObjectMapper objectMapper = new ObjectMapper();
        String json_string =
EntityUtils.toString(response.getEntity());
        Map<String,Object> result =
objectMapper.readValue(json_string, Map.class);
        KQNhanLichSuKCBBS kq = objectMapper.convertValue(result,
KQNhanLichSuKCBBS.class);
        System.out.println(kq);
    }
}

```

- Class hỗ trợ

```

public class ApiToken
{
    [DataMember]
    public string username { get; set; }
    [DataMember]
    public string password { get; set; }
}

public class KQPhienLamViec
{
    public string maKetQua { get; set; }
    public ApiKey APIKey { get; set; }
}

public class ApiKey
{
    [DataMember]
    public string access_token { get; set; }
    [DataMember]
    public string id_token { get; set; }
    [DataMember]
    public string token_type { get; set; }
}

```



```

        [DataMember]
        public string username { get; set; }
        [DataMember]
        public DateTime expires_in { get; set; }
    }

    public class ApiTheBHYT2018
    {
        [DataMember]
        public string maThe { get; set; }
        [DataMember]
        public string hoTen { get; set; }
        [DataMember]
        public string ngaySinh { get; set; }
    }

    public class KQNhanLichSuKCBBS
    {
        public string maKetQua { get; set; }
        public string ghiChu { get; set; }
        public string maThe { get; set; }
        public string hoTen { get; set; }
        public string ngaySinh { get; set; }
        public string gioiTinh { get; set; }
        public string diaChi { get; set; }
        public string maDKBD { get; set; }
        public string cqBHXH { get; set; }
        public string gtTheTu { get; set; }
        public string gtTheDen { get; set; }
        public string maKV { get; set; }
        public string ngayDu5Nam { get; set; }
        public string maSoBHXH { get; set; }
        public string maTheCu { get; set; }
        public string maTheMoi { get; set; }
        public string gtTheTuMoi { get; set; }
        public string gtTheDenMoi { get; set; }
        public List<LichSuKCB2018> dsLichSuKCB2018 { get; set; }
        public List<LichSuKT2018> dsLichSuKT2018 { get; set; }
    }

    public class LichSuKCB2018
    {
        public string maHoSo { get; set; }
        public string maCSKCB { get; set; }
        public string ngayVao { get; set; }
        public string ngayRa { get; set; }
        public string tenBenh { get; set; }
        public string tinhTrang { get; set; }
        public string kqDieuTri { get; set; }
    }

    public class LichSuKT2018
    {
        public string userKT { get; set; }
        public string thoiGianKT { get; set; }
        public string thongBao { get; set; }
        public string maLoi { get; set; }
    }

```