# HealthCare Authenticator (HCA)

Quick Start Guide - September 2024

1	Intro	oduction	.3	
	1.1	Authentication	.3	
	1.2	Identity verification	4	
	1.2		· -	
	1.3		.4	
	1.4	Connect with a OneKey / OWA account	.4	
	1.5	Get a Free Trial Subscription	.4	
2	Clie	Client-side integration		
	2.1	Requirements	.6	
	2 2	Downloading HCA SDK	6	
	2.2		.0	
	2.3	Integration details	.6	
	2.3.1	Initialize the HCA SDK	6	
	2.3.2	Sign-up	9	
	2.3.3	Sign-in	11	
	2.3.4	Call backs	15	
	2.3.5	Custom locale	16	
	2.3.6	Custom button labels	16	
	2.4	HCA Discovery end point	16	
	241	Using HCA Discovery end noint	16	
	2.4.2	Using HCA UserInfo end point	17	
3	Serv	er-side integration	18	
3	Serv	er-side integration1	18	
3	<i>Serv</i> 3.1	rer-side integration	1 <i>8</i> 18	
3	Serv 3.1 3.2	Per-side integration	1 <i>8</i> 18 18	
3	Serv 3.1 3.2 3.2.1	Requirements	18 18 18 18	
3	Serv 3.1 3.2 3.2.1 3.2.2	Per-side integration	18 18 18 18 20	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3	Per-side integration	<b>18</b> 18 18 18 20 20	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4	Per-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1	18 18 18 20 20 21	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3	Per-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>20</li> <li>21</li> <li>23</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1	Per-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>20</li> <li>21</li> <li>23</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         MFA form       1	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3	Per-side integration	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3 3.4	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>24</li> <li>25</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3 3.4 3.5	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1	<ol> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>25</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3 3.4 3.5 Fedd	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1         Others       1	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>26</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3 3.4 3.5 Fedd	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1         Others       1         Provision entegration       1	<ol> <li>18</li> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>26</li> <li>26</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3 3.3 3.4 3.5 Fedd 4.1	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         Consent form       1         Sign-in       1         Others       1         Requirements       1	<ol> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>26</li> <li>26</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3 3.4 3.5 Fedd 4.1 4.2	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1         Others       1         eration integration       1         Requirements       1         Integration details       1	<ol> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3.1 3.3.2 3.3.3 3.4 3.5 Fedd 4.1 4.2 4.2.1	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1         Others       1         eration integration       1         Requirements       1         Integration details       1         Configuring HCA as a new identity provider       1	<ol> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> <li>26</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3 3.3 3.3 3.4 3.5 Fede 4.1 4.2 4.2.1 4.2.2	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1         Others       1         eration integration       1         Requirements       1         Configuring HCA as a new identity provider       1         Using HCA Discovery end point       1	<ol> <li>18</li> <li>18</li> <li>20</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>25</li> <li>26</li> <li>26</li> <li>26</li> <li>27</li> </ol>	
3	Serv 3.1 3.2 3.2.1 3.2.2 3.2.3 3.2.4 3.3 3.3 3.3 3.3 3.4 3.5 Fede 4.1 4.2.1 4.2.2 4.2.3	rer-side integration       1         Requirements       1         Integration details       1         Configuring HCA as identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1         Using HCA Me end points       1         Sign-up       1         Sign-up form       1         MFA form       1         Consent form       1         Sign-in       1         Others       1         eration integration       1         Requirements       1         Configuring HCA as a new identity provider       1         Using HCA Discovery end point       1         Using HCA UserInfo end point       1	<ol> <li>18</li> <li>18</li> <li>20</li> <li>21</li> <li>23</li> <li>24</li> <li>24</li> <li>25</li> <li>26</li> <li>26</li> <li>27</li> <li>27</li> </ol>	

4.3	Sign-up	
4.3.	1 Sign-up form	
4.3.	2 MFA form	
4.3.	3 Consent form	
4.4	Sign-in	
4.5	Others	

# **1** Introduction

In this quick start guide you will find information about how to integrate HealthCare Authenticator in your web site or mobile application. There are different ways to perform such an integration depending on the technology your web site or mobile is using. As examples:

- If your web site uses Javascript, you will opt for a Client-side integration using HCA JS or Angular JS SDK using OpenId protocol and a Proof Key for Code Exchange (PKCE).
- If your web site uses a CMS (WordPress, Drupal...) you will opt for a Server-side integration using OpenId protocol.
- If your web site already has an authentication solution installed, then you will choose a Federation integration using OpenID protocol.



HCA modes of integration

### **1.1 Authentication**

In all cases, HCA will manage fully users' authentication, including Sign-up, Multi-Factor authentication, Consent gathering, Sign-in and Forget my password processes.



Examples of screens provided by HCA

## **1.2 Identity verification**

HCA uses Sign-up data to perform automatic identity verification based on OneKey data. It can also continue the process with manual identity verification performed by our research associates located all around the world. HCA provides an endpoint that allows to get identity verification status.



HCA identity verification process

### **1.3 OneKey Profile**

Once HCP identity has been verified, HCA endpoint can provide with additional data coming from our OneKey database.



Example of data from HCP OneKey profile

### 1.4 Connect with a OneKey / OWA account

Once an HCP will have created his OneKey account, he will be able to use his OneKey credentials to connect to any other web site or mobile application using HCA or OWA (previous IQVIA version of HCA). If the HCP had already an OWA account, then he can do the same with his OWA credentials.



HCP can connect directly with his OneKey account

### 1.5 Get a Free Trial Subscription

To get a HealthCare Authenticator Free Trial, please read the Free Trial Quick Start Guide.

# Note that:

- HCA Free Trial version is a standard English version without any customization and no access to OneKey Data.
- We have set a default endpoint URI to <u>http://localhost:8080</u> to allow you a local test.

You can provide us with different URIs (generally one for your development environment and one for your production environment). Bear in mind that if you do not use localhost, those URIs must be **https** ones.

In return we will provide you with:

- A Client ID that is required to authenticate requests from the HCA components and pre-built screens within your web site or mobile app.
- Claims URLs
- Discovery end point (OpenID connect Discovery URL): https://auth.onekeyconnect.com/auth.onekeyconnect.com/b2c\_1a\_hca\_signup\_signin/v2.0 /.well-known/openid-configuration

# 2 Client-side integration

### 2.1 Requirements

You should feel comfortable working with HTML5, CSS3, and JavaScript (JQuery and JQuery Mobile) at the very least.

### 2.2 Downloading HCA SDK

HealthCare Authenticator SDK source code and binary files are available at the following GitHub location: <u>https://github.com/orgs/hca-sdk</u>

To integrate the HCA SDK within a website (using Angular JS) or JavaScript app, you can either:

• Load the HCA SDK directly from our CDN, by the following line of code inside your HTML template. We recommend placing it at the end of the body tag to avoid blocking the website initial rendering.

Example:

<body> <script src="https://static. healthcaresdks.com/hca/v1/hca-sdk.js"></script> </body>

• Download HCA SDK from GitHub and host it by your own.

### 2.3 Integration details

#### 2.3.1 Initialize the HCA SDK

To initialize the HCA SDK, you need to add the following script at the end of the body:

```
JavaScript SDK within your JS/HTML app or website

<body>

<!-- HCA SDK Set Configuration -->

<script>

hcaSdk.setHcaSdkConfig(<clientId>[,<displaySignInButton>, <displaySignUpButton>, <scopes>,

<knownAuthorities>, <tenantDoman>, <policyId>, <signupPolicyId>, <apimSubscriptionKey>,

<apiBasePath>, <errorRedirectUrl>, <redirectUrl>]);

</script>

</body>
```

### JavaScript SDK within your Angular app

### // On the angular AppComponent, client should implement OnInit

export class AppComponent implements OnInit {
 isAccountLogged: boolean = false;
 accountProfile!: UserProfile;
 isProfileReady: boolean = false;

#### // In the OnInit Method configure the SDK:

#### ngOnInit() {

this.accountProfile = this.initaccountProfile(); // HCA Set Configuration hcaSdk.setHcaSdkConfig(<clientId>, <displaySignInButton>, <displaySignUpButton>, <scopes>, <knownAuthorities>, <tenantDoman>, <policyId>, <signupPolicyId>, <apimSubscriptionKey>, <apiBasePath>, <errorRedirectUrl>, <redirectUrl>); hcaSdk.setLoginCallBack(this.resultLogin.bind(this)); } onLogin() { hcaSdk.signIn(); } onSignup() { hcaSdk.signUp(); } initaccountProfile(): UserProfile { let userProfile: UserProfile = { title: ", firstName: ", lastName: ", email: ", phone: ",

```
trustLevel: 0
};
return userProfile;
}
```

}

```
export interface UserProfile {
title: string;
firstName: string;
lastName: string;
email: string;
phone: string;
trustLevel: number
```

- ClientID (string, required): is the unique ID that identifies the application that we provided you with.
- displaySignInButton (boolean, **optional**, default: true): controls whether the sign-in button is displayed. Set to false to hide the sign-in button.
- displaySignUpButton (boolean, optional, default: true): controls whether the sign-up button is displayed. Set to false to hide the sign-up button.
- Scopes (array, optional, default: ["<u>https://auth.onekeyconnect.com/x/profile.basic</u>"]): is a list of scopes that defines the information that the application would like to access to.
- knownAuthorities (array, **optional**, default: ["auth.onekeyconnect.com"]): Known authorities define the trusted authentication providers.
- tenantDomain (string, **optional**, default: "auth.onekeyconnect.com"): The domain of the tenant used for authentication.
- policyld (string, **optional**, default: "b2c\_1a\_hca\_signup\_signin"): The policy identifier for combined sign-up and sign-in flow.
- signupPolicyId (string, optional, default: "b2c\_1a\_hca\_signuponly"): The policy identifier for sign-up only, used when the user needs to create a new account.
- apimSubscriptionKey (string, optional, default: ""): API Key of the application used for accessing the API.
- apiBasePath (string, **optional**, default: <u>https://api.healthcaresdks.com/api/hca/user/me</u>): The base path of the API used to retrieve user information.
- errorRedirectUrl (string, optional, default: ""): The URL to which users will be redirected in case of an error during the authentication. If left empty, no redirection will occur.
- redirectUrl (string, optional, default: ""): The URL to which users will be redirected after successful authentication. If left empty, users will be redirected to the page where the authentication is initiated.

From your configuration interface, you can provide a list of scopes separated by a comma:

- Mandatory scope: Openid
- Mandatory scope: <u>https://auth.onekeyconnect.com/x/profile.basic</u> is configured by default.
- <u>https://auth.onekeyconnect.com/x/profile.basic</u> will give you access to:
  - UserID
  - TrustLevel
- o <u>https://auth.onekeyconnect.com/x/profile.extended</u> will give you access to:
  - UserID
  - Onekey ID
  - TrustLevel
  - Title
  - Firstname
  - Lastname
  - ProfessionalType
  - WorkplaceName
  - Address
  - PostalCode
  - City
  - County
  - Country
  - PhoneNumber
  - ProfessionalCode
  - Specialities



Note that:

- Using the scope "profile.extended" requires an existing OneKey subscription or a specific OneKey contract.
- You have to replace the scope "profile.extended" by "profile.basic". You can't have both scopes in the same configuration.

### 2.3.2 Sign-up

2.3.2.1 Integration of the Sign-up button

Display the 'Sign-up' button UI on your app or web site.

To display a Sign-up button (that will disappear after a user successfully signed up) you should add this script:



JavaS	cript SDK within your A	ngular app			
// In t	he AppComponent HTML a	add the Signup	Call:		
<li< td=""><td>class="nav-item"&gt;<a< td=""><td>href="#"</td><td>class="btn-signup</td><td>waves-effect</td><td>waves-light"</td></a<></td></li<>	class="nav-item"> <a< td=""><td>href="#"</td><td>class="btn-signup</td><td>waves-effect</td><td>waves-light"</td></a<>	href="#"	class="btn-signup	waves-effect	waves-light"
(click)	="onSignup()">Sign-up <td>&gt;</td> <td></td> <td></td> <td></td>	>			

#### 2.3.2.2 Sign-up form

When a user clicks on the Sign-up button, he accesses to the Sign-up form. The Sign-up form contains fields to information on user's profile:

- First Name
- Last Name
- Email
- Phone (Workplace)
- Country (List of countries)
- Postal code (Workplace)
- City (Workplace)
- Professional type (List of OneKey professional types)
- Specialty (List of OneKey specialties)
- Professional Code

Sign Up Create your OneKey Account				
			First name *	Last name *
			Business email	Workplace Phone
Workplace Country*	Workplace Postcode			
Workplace City *	Professional Type *			
Specialty*	RN			
*Mandatory fields				
Cont	inue			
Already have an	account? Login			
Personal information collected in the registration form is solely the eclaration law you have the right to access this information, make and the contact Deakay Connect # 2024 AU John	right of IQVIA and its approved clients. As stated in the local data indiments and cancel your registration at any time. This can be done via link below. researced. Naad hale? Contact us			

Note that: First Name, Last Name, Country, City, Professional type and Specialty are mandatory for identity verification purpose (automatic and manual).

#### 2.3.2.3 MFA form

During the Sign-up process, Multi-Factor Authentication (MFA) is activated and mandatory. The user has to enter his email address to receive a verification code, to enter in the form to verify his email. Then he can create his password.

	Sign Up	
Thank you for filling of your username. Y	out your information! Please provi ou will receive a verification code registration.	le your account email that will be used as o verify your email and to finalise your
Account Email (Username)		
dr-ocadou@yopmail.c	com	
	Send verification	code
Password		
Confirm password		
Confirm password		
Confirm password By signing up, you acc	ept OneKey HealthCare Authentic	tor Terms of use and Privacy Policy.
Confirm password By signing up, you acc	ept OneKey HealthCare Authentic	tor Terms of use and Privacy Policy.
Confirm password By signing up, you acc I agree to receive s communications fi	ept OneKey HealthCare Authentic scientific communications, invitati rom IQVIA and its partners.	tor Terms of use and Privacy Policy.
Coefirm password By signing up, you acc l agree to receive s communications fr	ept OneKey HealthCare Authentic clientific communications, invitati rom IQVIA and its partners.	tor Terms of use and Privacy Policy. Ins to events, and marketing Sign Up
Confirm password By signing up, you acc I agree to receive s communications fr	ept OneKey HealthCare Authentic icientific communications, invitati room IQVIA and its partners. ancel	tor Terms of use and Privacy Policy. Ins to events, and marketing Sign Up
Confirm password By signing up, you acco I agree to receive s communications fr	ept OneKey HealthCare Authentici scientific communications, invitati rom IQVIA and its partners. ancel	tor Terms of use and Privacy Policy. Ins to events, and marketing Sign Up

### 2.3.2.4 Consent form

At the last step of the sign-up process, the user will have to accept the Data share consent. The Data share consent allows HealthCare Authenticator to share the user's sign-up details with your company and allows you to collect your users' consents.

<b>Chercey</b> Connect		
All	ow OneKey Conne	ct to access your profile Data
Allow OneKey Connect to access your data to quickly set up your profile on its website. You can access and modify your shared data at any time from your OneKey account.		
Please provide us with the	following consents:	
If accept the Terms of Use and Privacy Policy of the Customer     Is accept to receive Marketing Emails from the Customer		
		*Mandatory neids
Ca	incel	Allow

*Note that:* You can create and collect up to 10 mandatory and 10 optional consents displayed in the Data share consent page.

#### 2.3.3 Sign-in

63

#### 2.3.3.1 Integration of the Sign-in Button

Display the "Sign-In" button UI on your app or web site.

To display a Sign-in button (that will turn to a logout after a user successfully logged in / Name of logged user + icon) you should add this script:

```
JavaScript SDK within your JS/HTML app or website

<body>

<div>

<!-- Place this div where want to display the Sign-In button -->

<div id="hca_signin">

</div>

</div>

</body>
```

#### JavaScript SDK within your Angular app

// In the AppComponent HTML add the LogIn Call: <a href="#" class="btn-login waves-effect waves-light" (click)="onLogin()">Log in</a>

### 2.3.3.2 Sign-in Form

When a user clicks on the sign-in button, he accesses to the sign-in form. The sign-in form contains:

- Username, which is the email he used during HCA MFA process.
- Password, which is the password he defined during HCA MFA process.

Onekey	Connect
Log	gin
Connect with your OneKey/OWA credentia	als or get a link to login without password.
Usemarne (Email)	
Password	
Keep me signed in Not recommended for a bits or shared computers.	
nor recommended for possion on amarina composition	
Login	Send me a link
Reset or Cree Don't have an ad	ate password ccount? Sign-up

#### Note that:

- The user can also log in with his OWA credentials (previous version of HCA).
- From the sign-in form, the user can also access to Forget Password form and to Sign-up form.

#### 2.3.3.3 Handle login

To define the return function to handle the login, you should add this script:

<body></body>
HCA SDK Set Configuration
<script></td></tr><tr><td>hcaSdk.setLoginCallBack(updateUILogin);</td></tr><tr><td></script>
The callback function below displays the ID of the logged user:
function updateUILogin(account) {
console.log(account.userId);
1

#### 2.3.3.4 Connected user

When user successfully signs in, the sign-in button can be replaced by user's information (icon, first name, last name, connection's date & hour).

To do that, please use data coming from basic OneKey Connect user's profile.



*Note that:* An example of HCA's integration is available on GitHub, on which you will see how this capability has been implemented.

#### 2.3.3.5 Service calls for additional data 1

The method is AccountLogged checks if the user is logged in or not (return true if the user is logged in and false if not:

hcaSdk.isAccountLogged()

Examples:

```
JavaScript SDK within your JS/HTML app or website
```

function isLogged() {
 console.log("isLogged:" + hcaSdk.isAccountLogged());

}

#### JavaScript SDK within your Angular app

```
isLogged() {
```

```
console.log(hcaSdk.isAccountLogged());
```

### 2.3.3.6 Sign-out link

To allow the user to sign-out, please use the following function:

JavaScript SDK within your JS/HTML app or website

hcaSdk.signOut()

JavaScript SDK within your Angular app onLogout() { hcaSdk.signOut(); }

#### 2.3.3.7 Service calls for additional data 2

The method getProfile() retrieves the profile data of the logged user.

```
Example:
```

```
JavaScript SDK within your JS/HTML app or website
function getProfile() {
    hcaSdk.getProfile(callbackProfile);
}
function callbackProfile(data) {
    console.log(data);
```

#### JavaScript SDK within your Angular app

```
getProfilePromise = () => {
  return new Promise((resolve, reject) => {
    hcaSdk.getProfile((data: any, err: any) => {
    if (err) return reject(err)
    resolve(data)
  })
})
```



• /api/hca/user/me/account: user's Sign-up data, including validated email in MFA process from HCA:

Me/account fields	Description
ld, userld	ID of the user
trustLevel	User's trust level (level of identity verification)
firstName	User's first name
lastName	User's last name
email	Email and username
businessEmail	User's business email
phone	Workplace phone
country	User's country
zipCode	Workplace postal code
city	Workplace city
profession	Professional type code
specialty	Specialty code
uci	Professional code

 /api/hca/user/me/profile: user's OneKey data. Available OneKey data depends on scopes defined when configuring HCA (Cf. 2.3.1) and on existing OneKey subscription / contract and identity verification process' status or result:

For a "profile.basic" scope, you will have access to:

Me/profile fields	Description
userId	ID of the user
trustLevel	User's trust level

For a "profile.extended" scope, you will have access to:

Me/profile fields	Description
userId	HCA internal user ID
ucis	User's local professional code
	User's Trust level (level of identity verification
trustLevel	validation) which comes from HCA database or ID
	verification process
	Code of Professional Title of the Healthcare
title (code, label, locale)	Professional (HCP). Ex. Doctor, Professor or
	Courtesy title (Mr., Mrs., Miss)
firstName	HCP's First Name in local Alphabet
lastName	HCP's Last Name in local Alphabet
suffixName	HCP's Suffix Name in local Alphabet
professionalType (code, label,	HCP's profession label in sign up's form's locale
locale)	The spinlession laber in sign-up's form's locale

specialties	HCP's profession local OneKey code
intlPhono	International Phone number of the healthcare
Internone	professional at a workplace.
workplaceName	Official name for the Workplace
ZipCode	Workplace Postal code or zip code
City (code, label, locale)	City Name
Country	ISO-2 country code like FR, GB, DE
localPhone	workplace international phone
IntlFax	workplace international fax
business name	workplace name
husinossAddross	long label of workplace address (i.e. full readable
DusinessAduress	address)
npiNumber	map of user uci (ex: {"adeli":1234, "rpps": 123})

#### 2.3.3.8 First Sign-in & Consent

If a user signs-in for the first time on your web site or mobile app using his OneKey/OWA credentials (and didn't Sign-up on your web site), then HCA will proceed to an automatic Sign-in.

In that case, user will have to accept the Data share consent page for us to provide you with data coming from HCA sign-up and OneKey details.

### Note that:

At the end of the sign in process, our authentication server calls back the redirect URI that was given in the parameters. This will trigger a reload of the webpage, the SDK will process the information and make it available to your code. It is important to consider this workflow when using SDK's function to access user information. This information is only available once the redirect URI is called.

### 2.3.4 Call backs

There are 3 more call back functions in additions the ones (setLoginCallback) described above:

- A call back function to access Token:
   // HCA Set callback function to handle access Token hcaSdk.setTokenCallBack(updateUIToken);
- A call back function to handle Cancel action: // HCA Set callback function to handle cancel action hcaSdk.setCancelCallBack(onCancel);

Example: When a user cancels his sign-in or Sign-up process, you need to intercept HCA error message to provide user with a message and/or redirect him (e.g. to your home page).

 A call back function to handle Cancel action:
 // HCA Set callback function to handle errors hcaSdk.setErrorCallBack(onError);

### 2.3.5 Custom locale

The **hcaSdk.setLocaleParams(locale)** method allows you to specify the language or regional settings (locale) for the HCA's sign-in / sign-up page.

// Set locale params for HCA's signin / signup pages hcaSdk.setLocaleParams("fr-FR");

### 2.3.6 Custom button labels

To customize the text labels for the sign-in, sign-up, and sign-out buttons, you could use the **hcaSdk.setLabels(<signIn>, <signUp>, <signOut>)** methods.

// Set text labels for the sign-in, sign-up, and sign-out buttons
hcaSdk.setLabels("Sign In with OneKey", "Sign Up with OneKey", "Sign Out from OneKey");

### 2.4 HCA Discovery end point

#### 2.4.1 Using HCA Discovery end point

The <u>HealthCare Authenticator Discovery end point</u> defines a mechanism for you to discover HCA and obtain information needed to interact with it. It enables you to:

- Verify the identity of the end-user based on the authentication performed by Authorization Server,
- Obtain basic profile information about the end-user in an interoperable and REST-like manner (Cf. below: email, displayName, family\_name, trustLevel...),
- Obtain OAuth 2.0 endpoint locations (Cf. below: userinfo\_endpoint, end\_session\_endpoint...).



(More details on: https://openid.net/specs/openid-connect-discovery-1 0.html)

### 2.4.2 Using HCA UserInfo end point

This HCA Discovery endpoint includes UserInfo endpoint URL, here:

https://auth.onekeyconnect.com/auth.onekeyconnect.com/b2c\_1a\_hca\_signup\_signin/openid/ v2.0/userinfo

METHOD: GET
URL:
https://auth.onekeyconnect.com/auth.onekeyconnect.com/openid/v2.0/userinfo?p=b2c 1a hca signup
signin
HEADER: Bearer <token></token>
This User where and sint is an OAuth 2.0 exetested recourse where you can retrieve concented

This UserInfo endpoint is an OAuth 2.0 protected resource where you can retrieve consented claims, or assertions, about the end-user authenticated with his HCA sign-up/OneKey account. The claims are typically packaged in a JSON object where the sub member denotes the subject (end-user) identifier.

You need to call UserInfo endpoint using signed JWT (JSON Web Token) that you obtain during authentication process. It will return a JSON object including UserInfo Fields with data coming from the Sign-up and MFA process:

UserInfo field	Description	Source
objectID	UserID	НСА
displayName	Last Name	Sign-up form
givenName	First Name	Sign-up form
surname	Last Name	Sign-up form
zipCode	Postal Code	Sign-up form
city	City name	Sign-up form
country	Country	Sign-up form
phone	Phone number	Sign-up form
businessEmail	Email provided	Sign-up form
profession	Profession Type code	Sign-up form
specialty	Specialty code	Sign-up form
email	Email and username	Sign-up form (MFA)
uci	Professional Code	Sign-up form
	OneKey ID (If the client has access	
oneKeyID	to OneKey extended profile and if	HCA / OneKey
	user has been identified)	
trustLevel	User's trust level	НСА

# 3 Server-side integration

### 3.1 Requirements

### CMS (Content Management System) users

- You must add a CMS (e.g. Drupal or WordPress) OpenID Connect (OIDC / openidconnect) Client plugin (e.g. <u>https://plugins.miniorange.com/wordpress-oauth-client-setup</u> or <u>https://plugins.miniorange.com/drupal</u>) to your web site.
- This plugin will be integrated with the HealthCare Authenticator. It will act as a Service provider to establish a trust between HCA (which is the OAuth/OpenID/JWT Identity Provider) and your CMS. This will allow users to quickly and securely login to your CMS site using HCA.

*Note that:* On our website, from the <u>Developers page</u>, Videos section, you will find tutorial videos for WordPress/Drupal integration.

### 3.2 Integration details

### 3.2.1 Configuring HCA as identity provider

To configure HCA as identity provider, you will need to log as an administrator and follow your CMS OpenID Connect Client plugin documentation. In general, you will be required to add a new identity provider (OAuth provider) by providing the following information:

- Identity provider name: OneKey
- Client ID: this refers to the Client ID you got from your account on our web site (<u>https://www.healthcaresdks.com/en/authenticator</u>).
- Client Secret: this refers to the Client Secret you got from your account on our web site. It will be used to create a Policy Key.



Examples of configuration: <u>https://plugins.miniorange.com/wordpress-oauth-client-setup</u>

From your configuration interface, you can provide a list of scopes separated by a space:

- o Mandatory scope: Openid
- Mandatory scope: <u>https://auth.onekeyconnect.com/x/profile.basic</u> is configured by default.
- <u>https://auth.onekeyconnect.com/x/profile.basic</u> will give you access to:
  - UserID
  - TrustLevel
- o <u>https://auth.onekeyconnect.com/x/profile.extended</u> will give you access to:
  - UserID
  - Onekey ID
  - TrustLevel
  - Title
  - Firstname
  - Lastname
  - ProfessionalType
  - WorkplaceName
  - Address
  - PostalCode
  - City
  - County
  - Country
  - PhoneNumber
  - ProfessionalCode
  - Specialities

### Note that:

- Using the scope "profile.extended" requires an existing OneKey subscription or a specific OneKey contract.
- You have to replace the scope "profile.extended" by "profile.basic". You can't have both scopes in the same configuration.

Generally, the solution you use will allow you to test the configuration. In that case, you will see all the values returned by HCA (First Name, Last Name, Email) to CMS in a table (Cf. below). Once you see all the required values, then you will have to proceed User Attribute / Role Mapping which is mandatory for enabling users to successfully login into CMS.

Attribute Name		ttribute Value
np	16766	
shf	16766	
rer	1.9	
	https://auth.onekeyconnect.com 085bb52898u@v2.0/	r5e 1947
ub	4678ad05-30	EN794-0
nd	de5e9218-55	(Taaticeb
icr .	h2c_la_hca_signup_signin	
ie.	16766	
wth_time	1678	
scope	openid https://auth.onekeyconsact.com/s/profile.extended	
mail	email from sign-up form)	
dφ	lqvia-hca-aser	
nerkt	4678ai Joh	128794-0
tisplayName	First name from sign-up form)	
lamily_name	Last name from sign-up form)	
rustLevel	Trust level from OneKey	
id .	9642986 885	665289849
diad. h	0.000	

Test configuration example

### 3.2.2 Using HCA Discovery end point

The <u>HealthCare Authenticator Discovery end point</u> defines a mechanism for you to discover HCA and obtain information needed to interact with it. It enables you to:

- Verify the identity of the end-user based on the authentication performed by Authorization Server,
- Obtain basic profile information about the end-user in an interoperable and REST-like manner (Cf. below: email, displayName, family\_name, trustLevel...),
- Obtain OAuth 2.0 endpoint locations (Cf. below: userinfo\_endpoint, end\_session\_endpoint...).



(More details on: https://openid.net/specs/openid-connect-discovery-1 0.html)

### 3.2.3 Using HCA UserInfo end point

This HCA Discovery endpoint includes UserInfo endpoint URL, here:

https://auth.onekeyconnect.com/auth.onekeyconnect.com/b2c 1a hca signup signin/openid/v2.0 /userinfo

METHOD: GET
URL:
https://auth.onekeyconnect.com/auth.onekeyconnect.com/openid/v2.0/userinfo?p=b2c 1a hca signup
signin
HEADER: Bearer <token></token>
This UserInfo endpoint is an OAuth 2.0 protected resource where you can retrieve consented

This UserInfo endpoint is an OAuth 2.0 protected resource where you can retrieve consented claims, or assertions, about the end-user authenticated with his HCA sign-up/OneKey account.

The claims are typically packaged in a JSON object where the sub member denotes the subject (end-user) identifier.

You need to call UserInfo endpoint using signed JWT (JSON Web Token) that you obtain during authentication process. It will return a JSON object including UserInfo Fields with data coming from the Sign-up and MFA process:

UserInfo field	Description	Source
objectID	UserID	НСА
displayName	Last Name	Sign-up form
givenName	First Name	Sign-up form
surname	Last Name	Sign-up form
zipCode	Postal Code	Sign-up form
city	City name	Sign-up form
country	Country	Sign-up form
phone	Phone number	Sign-up form
businessEmail	Email provided	Sign-up form
profession	Profession Type code	Sign-up form
specialty	Specialty code	Sign-up form
email	Email and username	Sign-up form (MFA)
uci	Professional Code	Sign-up form
	OneKey ID (If the client has access	
oneKeyID	to OneKey extended profile and if	HCA / OneKey
	user has been identified)	
trustLevel	User's trust level	НСА

### 3.2.4 Using HCA Me end points

Me API has 2 endpoints to allow you to retrieve user's Sign-up data and user's OneKey data. You need to call the Me endpoints using signed JWT (JSON Web Token) that you obtain during authentication process.

### 3.2.4.1 me/account

METHOD: GET
URL: https://apim-prod-westeu-onekey.azure-api.net/api/hca/user/me/account
HEADER: Bearer <token></token>

/api/hca/user/me/account: user's Sign-up data, including validated email in MFA process from HCA:

Me/account fields	Description
ld, userld	ID of the user
trustLevel	User's trust level (level of identity verification)
firstName	User's first name
lastName	User's last name
email	Email and username
businessEmail	User's business email

phone	Workplace phone
country	User's country
zipCode	Workplace postal code
city	Workplace city
profession	Professional type code
specialty	Specialty code
uci	Professional code

### 3.2.4.2 me/profile

METHOD: GET
URL: <u>https://apim-prod-westeu-onekey.azure-api.net/api/hca/user/me/profile</u>
HEADER: Bearer <token></token>

/api/hca/user/me/profile: user's **OneKey data**. Available OneKey data depends on scopes defined when configuring HCA (Cf. <u>3.2.1</u>) and on existing OneKey subscription / contract and identity verification process' status or result:

For a "profile.basic" scope, you will have access to:

Me/profile fields	Description
userId	ID of the user
trustLevel	User's trust level

For a "profile.extended" scope, you will have access to:

Me/profile fields	Description	
userId	HCA internal user ID	
ucis	User's local professional code	
	User's Trust level (level of identity verification	
trustLevel	validation) which comes from HCA database or ID	
	verification process	
	Code of Professional Title of the Healthcare	
title (code, label, locale)	Professional (HCP). Ex. Doctor, Professor or	
	Courtesy title (Mr., Mrs., Miss)	
firstName	HCP's First Name in local Alphabet	
lastName	HCP's Last Name in local Alphabet	
suffixName	HCP's Suffix Name in local Alphabet	
professionalType (code, label,	UCD's profession label in sign un's form's lessle	
locale)	her's profession laber in sign-up's form's locale	
specialties	HCP's profession local OneKey code	
intiphene	International Phone number of the healthcare	
Internone	professional at a workplace.	
workplaceName	Official name for the Workplace	
ZipCode	Workplace Postal code or zip code	
City (code, label, locale)	City Name	

Country	ISO-2 country code like FR, GB, DE
localPhone	workplace international phone
IntlFax	workplace international fax
business name	workplace name
businessAddress	long label of workplace address ( ie full readable
	address )
npiNumber	map of user uci ( ex : { "adeli":1234, "rpps": 123 } )

# 3.3 Sign-up

### 3.3.1 Sign-up form

When a user clicks on the Sign-up button, he accesses to the Sign-up form. The Sign-up form contains fields to information on user's profile:

- First Name
- Last Name
- Email •
- Phone (Workplace) •
- Country (List of countries)
- Postal code (Workplace) •
- City (Workplace)
- Professional type (List of OneKey professional types) •
- Specialty (List of OneKey specialties)
- **Professional Code** •

	Sign Up
	Create your OneKey Account
First name *	Last name *
Business email	Workplace Phone
Workplace Country * UNITED KINGDOM	Workplace Postcode
Workplace City *	Professional Type *
Specialty *	RN
*Mandatory fields	
	Continue
	Already have an account? Login

Note that: First Name, Last Name, Country, City, Professional type and Specialty are mandatory for identity verification purpose (automatic and manual).

### 3.3.2 MFA form

During the Sign-up process, Multi-Factor Authentication (MFA) is activated and mandatory. The user has to enter his email address to receive a verification code, to enter in the form to verify his email. Then he can create his password.

	OneKey Connect
	Sign Up
Thank you fo your user	or filing out your information! Please provide your account email that will be used a mame. You will receive a verification code to verify your email and to finalise your registration.
Account Email (U dr-ocadou@y	Januared) yoppmail.com
	Send verification code
Password	
Confirm passwor	ni
Confirm passwor	nd
Confirm passwor By signing up, l agree to communic	nd Lyou accept OneKey HealthCare Authenticator Terms of use and Privacy Policy. receive scientific communications, invitations to events, and marketing cations from IQVIA and its partners.
Confirm passwor By signing up, lagree to communic	nt A, you accept OneKey HealthCare Authenticator Terms of use and Privacy Policy. receive scientific communications, invitations to events, and marketing cations from IQVIA and its partners. Cancel Sign Up
Confirm passwor By signing up, lagree to communic	et a, you accept OneKey HealthCare Authenticator Terms of use and Privacy Policy. receive scientific communications, invitations to events, and marketing cations from 10VIA and its partners. Cancel Sign Up Already have an account' Login

### 3.3.3 Consent form

At the last step of the sign-up process, the user will have to accept the Data share consent. The Data share consent allows HealthCare Authenticator to share the user's sign-up details with your company and allows you to collect your users' consents.

	One	Connect
	Allow OneKey Conne	ect to access your profile Data
Allow OneK nodify your	ey Connect to access your data to quick shared data at any time from your Onel	ly set up your profile on its website. You can access and Key account.
Please prov	ide us with the following consents:	
□ *l accept	the Terms of Use and Privacy Policy of	the Customer
I accept t Mandatory finance	o receive Marketing Emails from the Cu elds	stomer
	Cancel	Allow

1 Note that: You can create and collect up to 10 mandatory and 10 optional consents displayed in the Data share consent page.

### 3.4 Sign-in

When a user clicks on the sign-in button, he accesses to the sign-in form. The sign-in form contains:

- Username, which is the email he used during HCA MFA process.
- Password, which is the password he defined during HCA MFA process.

OneKey	Connect
Lo	ogin
Connect with your OneKey/OWA credent	ials or get a link to login without password.
Usemame (Email)	
Password	
Keep me signed in     Not recommended for public or shared computers	
	Send me a link
Reset or Cro Don't have an a	eate password account? Sign-up

#### Note that:

- The user can also log in with his OWA credentials (previous version of HCA).
- From the sign-in form, the user can also access to Forget Password form and to Sign-up form.

### 3.5 Others

#### Data from Sign-up process

After Sign-up process HCA will give access to data from Sign-up and MFA form and from identity verification if purchased (OneKey record) using endpoints described above.

#### Cancel

When a user cancels his sign-in or Sign-up process, you need to intercept HCA error message to provide user with a message and/or redirect him (e.g. to your home page).

# **4** Federation integration

### 4.1 Requirements

Your authentication solution must be OpenID compatible.

### 4.2 Integration details

### 4.2.1 Configuring HCA as a new identity provider

To configure HCA as a new identity provider within your authentication system, you will need to log as an administrator and follow corresponding documentation.

In general, you will be required to add a new identity provider (OAuth provider) by providing the following information:

- Identity provider name: OneKey
- Client ID: this refers to the **Client ID** you got from your account on our web site (https://www.healthcaresdks.com/en/authenticator).
- Client Secret: this refers to the Client Secret you got from your account on our web site. It will be used to create a Policy Key.

From your configuration interface, you can provide a list of scopes separated by a space:

- Mandatory scope: Openid
- Mandatory scope: <u>https://auth.onekeyconnect.com/x/profile.basic</u> is configured by default.
- <u>https://auth.onekeyconnect.com/x/profile.basic</u> will give you access to:
  - UserID
  - TrustLevel
- o <u>https://auth.onekeyconnect.com/x/profile.extended</u> will give you access to:
  - UserID
  - Onekey ID
  - TrustLevel
  - Title
  - Firstname
  - Lastname
  - ProfessionalType
  - WorkplaceName
  - Address
  - PostalCode
  - City
  - County
  - Country
  - PhoneNumber
  - ProfessionalCode
  - Specialities



### Note that:

- Using the scope "profile.extended" requires an existing OneKey subscription or a specific OneKey contract.
- You have to replace the scope "profile.extended" by "profile.basic". You can't have both scopes in the same configuration.
- User Information Fields: User Information Fields are the claims related to the OneKey account. Your authentication solution requests these fields from HCA SDK when a user is authenticated with his OneKey account.

### 4.2.2 Using HCA Discovery end point

The <u>HealthCare Authenticator Discovery end point</u> defines a mechanism for you to discover HCA and obtain information needed to interact with it. It enables you to:

- Verify the identity of the end-user based on the authentication performed by Authorization Server,
- Obtain basic profile information about the end-user in an interoperable and REST-like manner (Cf. below: email, displayName, family\_name, trustLevel...),
- Obtain OAuth 2.0 endpoint locations (Cf. below: userinfo\_endpoint, end\_session\_endpoint...).



(More details on: https://openid.net/specs/openid-connect-discovery-1 0.html)

### 4.2.3 Using HCA UserInfo end point

This HCA Discovery endpoint includes UserInfo endpoint URL, here:

https://auth.onekeyconnect.com/auth.onekeyconnect.com/b2c 1a hca signup signin/openid/v2.0 /userinfo

METHOD: GET
URL: https://auth.onekeyconnect.com/auth.onekeyconnect.com/openid/v2.0/userinfo?p=b2c_1a_hca_signup
HEADER: Bearer <token></token>

This UserInfo endpoint is an OAuth 2.0 protected resource where you can retrieve consented claims, or assertions, about the end-user authenticated with his HCA sign-up/OneKey account. The claims are typically packaged in a JSON object where the sub member denotes the subject (end-user) identifier.

You need to call UserInfo endpoint using signed JWT (JSON Web Token) that you obtain during authentication process. It will return a JSON object including UserInfo Fields with data coming from the Sign-up and MFA process:

UserInfo field	Description	Source
objectID	UserID	НСА
displayName	Last Name	Sign-up form
givenName	First Name	Sign-up form
surname	Last Name	Sign-up form
zipCode	Postal Code	Sign-up form
city	City name	Sign-up form
country	Country	Sign-up form
phone	Phone number	Sign-up form
businessEmail	Email provided	Sign-up form
profession	Profession Type code	Sign-up form
specialty	Specialty code	Sign-up form
email	Email and username	Sign-up form (MFA)
uci	Professional Code	Sign-up form
	OneKey ID (If the client has access	
oneKeyID	to OneKey extended profile and if	HCA / OneKey
	user has been identified)	
trustLevel	User's trust level	НСА

### 4.2.4 Using HCA Me end points

Me API has 2 endpoints to allow you to retrieve user's Sign-up data and user's OneKey data. You need to call the Me endpoints using signed JWT (JSON Web Token) that you obtain during authentication process.

### 4.2.4.1 me/account

METHOD: GET
URL: https://apim-prod-westeu-onekey.azure-api.net/api/hca/user/me/account
HEADER: Bearer <token></token>

/api/hca/user/me/account: user's Sign-up data, including validated email in MFA process from HCA:

Me/account fields	Description
ld, userld	ID of the user
trustLevel	User's trust level (level of identity verification)
firstName	User's first name
lastName	User's last name
email	Email and username
businessEmail	User's business email
phone	Workplace phone
country	User's country
zipCode	Workplace postal code
city	Workplace city
profession	Professional type code
specialty	Specialty code
uci	Professional code

#### 4.2.4.2 me/profile

METHOD: GET URL: <u>https://apim-prod-westeu-onekey.azure-api.net/api/hca/user/me/profile</u> HEADER: Bearer <token>

/api/hca/user/me/profile: user's **OneKey data**. Available OneKey data depends on scopes defined when configuring HCA (Cf. <u>3.2.1</u>) and on existing OneKey subscription / contract and identity verification process' status or result:

For a "profile.basic" scope, you will have access to:

Me/profile fields	Description
userId	ID of the user
trustLevel	User's trust level

For a "profile.extended" scope, you will have access to:

Me/profile fields	Description
userId	HCA internal user ID
ucis	User's local professional code
	User's Trust level (level of identity verification
trustLevel	validation) which comes from HCA database or ID
	verification process
	Code of Professional Title of the Healthcare
title (code, label, locale)	Professional (HCP). Ex. Doctor, Professor or
	Courtesy title (Mr., Mrs., Miss)
firstName	HCP's First Name in local Alphabet
lastName	HCP's Last Name in local Alphabet
suffixName	HCP's Suffix Name in local Alphabet
professionalType (code, label,	HCP's profession label in sign un's form's locale
locale)	HCF's profession laber in sign-up's form's locale
specialties	HCP's profession local OneKey code
intlPhono	International Phone number of the healthcare
Interione	professional at a workplace.
workplaceName	Official name for the Workplace
ZipCode	Workplace Postal code or zip code
City (code, label, locale)	City Name
Country	ISO-2 country code like FR, GB, DE
localPhone	workplace international phone
IntlFax	workplace international fax
business name	workplace name
husinessAddress	long label of workplace address ( ie full readable
DUSITIESSAUULESS	address )
npiNumber	map of user uci ( ex : { "adeli":1234, "rpps": 123 } )

### 4.3 Sign-up

### 4.3.1 Sign-up form

When a user clicks on the Sign-up button, he accesses to the Sign-up form. The Sign-up form contains fields to information on user's profile:

- First Name
- Last Name
- Email
- Phone (Workplace)
- Country (List of countries)
- Postal code (Workplace)
- City (Workplace)
- Professional type (List of OneKey professional types)
- Specialty (List of OneKey specialties)
- Professional Code

Oneney	oonnoor
Sign	l Up
Create your On	eKey Account
First name *	Last name *
Business email	Workplace Phone
Workplace Country * UNITED KINGDOM	Workplace Postcode
Workplace City *	Professional Type *
Specialty*	RN
*Mandatory fields	
Cont	inue
Already have an	account? Login
Personal information collected in the registration form is solely the leclaration law you have the right to access this information, make ame the contact	right of IQVIA and its approved clients. As stated in the local data indments and cancel your registration at any time. This can be done via link below.

Note that: First Name, Last Name, Country, City, Professional type and Specialty are mandatory for identity verification purpose (automatic and manual).

### 4.3.2 MFA form

During the Sign-up process, Multi-Factor Authentication (MFA) is activated and mandatory. The user has to enter his email address to receive a verification code, to enter in the form to verify his email. Then he can create his password.

	Sign Up	
Thank you for fil your usernan	ling out your information! Please provide ne. You will receive a verification code to registration.	your account email that will be used as verify your email and to finalise your
Account Email (Userna dr-ocadou@yopn	ame) nail.com	
	Send verification co	de
Descoursed		
Password		
Confirm password		
Confirm password	I accent OneKey HealthCare Authenticat	or Terms of use and Privacy Policy
Confirm password	u accept OneKey HealthCare Authenticat elve scientific communications, invitation ons from IQVIA and its partners.	or Terms of use and Privacy Policy. s to events, and marketing
Confirm password	u accept OneKey HealthCare Authenticat elve scientific communications, invitation nns from IQVIA and its partners. Cancel	or Terms of use and Privacy Policy, is to events, and marketing Sign Up
Confirm password — By signing up, you I agree to reco communicatio	u accept OneKey HealthCare Authenticat eive scientific communications, invitatior nas from IQVIA and its partners. Cancel Aiready have an account	or Terms of use and Privacy Policy. Is to events, and marketing Sign Up

### 4.3.3 Consent form

At the last step of the sign-up process, the user will have to accept the Data share consent. The Data share consent allows HealthCare Authenticator to share the user's sign-up details with your company and allows you to collect your users' consents.

Chercey Connect				
	Allow OneKey	Connect to a	access your profile Data	
llow OneKey ( nodify your sh	Connect to access your dat ared data at any time from	a to quickly set up your <mark>OneKey acco</mark>	your profile on its website. You can access and unt.	
lease provide	us with the following const	ents:		
*I accept the I accept to re Mandatory field:	e Terms of Use and Privacy eceive Marketing Emails fro s	Policy of the Cust om the Customer	omer	
	Cancel		Allow	

*Note that:* You can create and collect up to 10 mandatory and 10 optional consents displayed in the Data share consent page.

### 4.4 Sign-in

When a user clicks on the sign-in button, he accesses to the sign-in form. The sign-in form contains:

- Username, which is the email he used during HCA MFA process.
- Password, which is the password he defined during HCA MFA process.

OneKey	Connect
L	ogin
Connect with your OneKey/OWA credent	tials or get a link to login without password.
Usemame (Email)	
Password	
Keep me signed in Not recommended for public or shared computers	
	Send me a link
Reset or Cr Don't have an	eate password account? Sign-up

### Note that:

- The user can also log in with his OWA credentials (previous version of HCA).
- From the sign-in form, the user can also access to Forget Password form and to Sign-up form.

### 4.5 Others

#### Data from Sign-up process

After Sign-up process HCA will give access to data from Sign-up and MFA form and from identity verification if purchased (OneKey record) using endpoints described above.

#### Cancel

When a user cancels his sign-in or Sign-up process, you need to intercept HCA error message to provide user with a message and/or redirect him (e.g. to your home page).

# 5 Get a Pro version & Go live!

Once you tested your website or mobile app with HCA free trial solution, we will upgrade your free trial version in Pro version. With the Pro version, you will be able to:

- Select the countries needed for your website or mobile app.
- Customize your HCA pages:
  - **Graphical:** from your account on <u>https://www.healthcaresdks.com/en/</u>, in the tab "Customisation" > "Graphical" (see screenshot below), you can customize the logo, the fonts, the colors, etc. à Link to the graphical customisation guide <u>HERE</u> which will help you use the customization tool.

tyles		
General Settings	Pages background	~
<b>T⊤</b> Headings & Text		
Buttons & Links	Logo	~
Form Fields	Custom font	~
	Link to your website	~

 Consents: we will send you an Excel file to complete with the consents you will display to the users in the data share consent page in all languages needed. Once we receive all the information needed, we will implement the consent customization.

After this last step, you can do the final sanity check, and you are ready to go live!