Integration Specification
Documedis CDS.CE

The target groups of this document are project owners, product managers and software developers interested in adding CDS functionality to their individual software products using Documedis CDS.CE, a CE-certified module of the Documedis platform.

Do not redistribute. For use of INDEX/Documedis customers only.

1. Table of content

1. Table of content .............................................................................................................................................. 2
2. Introduction ..................................................................................................................................................... 3
3. CDS.CE ........................................................................................................................................................... 4
   3.1. Product Status ........................................................................................................................................ 4
   4.1. The CDS check types ............................................................................................................................. 6
   4.2. The CDS relevancies ............................................................................................................................. 7
5. Data vs. Software-as-a-Service (SaaS) .......................................................................................................... 8
6. Functions of the CDS.CE module ................................................................................................................... 9
   6.1. Using and understanding the CDS module ............................................................................................ 9
       6.1.1. ClinicalDecisionSupport (5) – the CDS check viewer .................................................................. 9
   6.2. Integrating the Documedis medication software into your clinical information system ................... 10
       6.2.1. General information: Request header fields for Identification/Authorization/Styling .............. 10
       6.2.2. The usage of the CDS functions of the medication module in your local IT system ............ 11
       6.2.3. The CHMED medication plan object – the core of the transmitted data .............................. 12
       6.2.4. Release management, CE-certified modules and APP and API routes .................................. 19
       6.2.5. System requirements for the APP .............................................................................................. 19
       6.2.6. APP and API for "ClinicalDecisionSupport CDS.CE" (5/6/7) ............................................... 20
       6.2.7. Access control ............................................................................................................................. 28
       6.2.8. Request error codes .................................................................................................................... 29
       6.2.9. Error handling / Offline behavior .................................................................................................. 29
       6.2.10. SLA / Availability / Performance .............................................................................................. 29
       6.2.11. Acceptance Criteria for verification of installation ................................................................... 30
       6.2.12. Support organization ................................................................................................................... 30
       6.2.13. Additional tools ............................................................................................................................ 31
7. Appendix ....................................................................................................................................................... 32
   7.1. APP UI integration: Receiving notification from the viewer ................................................................. 32
       7.1.1. For CDS.CE .................................................................................................................................. 32
   7.2. Example eMediplan .................................................................................................................................. 32
8. Version History .............................................................................................................................................. 33
2. Introduction

The INDEX databases published by HCI Solutions AG contain commercial and scientific information in German and French about pharmaceutical products and other products needed in the Swiss health industry; they also include the addresses of most of the service providers in the industry such as hospitals, doctors and companies. Depending on the needs of the individual user groups, the various INDEX products deliver data about 150'000-250'000 articles and 100'000 addresses. An overview of the available products can be found on our website at www.hcisolutions.ch/index

Based on the INDEX database, the product "Documedis" offers various functionalities to access this data through web interfaces as software-as-a-service (SaaS) for quick and seamless integration into 3rd party applications such as Clinical Information Systems (CIS/EHR) or web platforms like compendium.ch and pharmavista.ch. It supplements the existing plain data INDEX SOAP XML webservice with:

- Fully integrated software modules as web-based standalone Single Page Applications (SPA), referenced as "APP", featuring their own algorithms and advanced layouts.
- Additional PDF generators for all relevant read-only views of the SPA.
- Numerous modular microservices as web-based JSON REST interfaces, featuring all necessary interfaces to implement the functionalities of the SPA yourself, referenced as "API".

This allows you to quickly add new core functionalities to your existing software, saving the effort to download, import and update the INDEX data yourself and the need to write, test and finance your own implementation.

Documedis is split up into business domain modules, each with its own APP, API and documentation:

- ProductInfo: Detailed product and article information
- ProductSearch: Intelligent search variants to find products, including full-text search and filtering
- Register: Hierarchic search variants to find products through tree browsing
- News: Product news for a product or by subject area
- Medication: Medication plan viewer and editor
- CDS.CE: A risk-based Clinical Decision Support check (CDS), a CE-certified medical device.
- ServiceProvider: Information about medical professionals in the Swiss Health Sector

The modules are available from the following URLs and subsites:

<table>
<thead>
<tr>
<th>Release</th>
<th>Route</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>/api/app</td>
<td>Current Documedis showcase, as used in compendium.ch and pharmavista.net Not certified. Moving target, no release management / continuous integration. Uses latest version of the CDS.CE, the CE-certified CDS-Check module.</td>
</tr>
<tr>
<td>2018-01</td>
<td>/api/app</td>
<td>Non-certified modules of Documedis, under strict release management. For use in noncritical business domains of 3rd party medical information systems. Read-only product/serviceprovider information, medication editor.</td>
</tr>
</tbody>
</table>

The target audience for this document are business analysts and software developers interested in integrating Documedis modules in their software products. In this manual, we explain the CDS.CE module functionalities available through the APP and the API.

Early 2018, the first Documedis modules are being made available to selected market leaders for pilot projects.
3. CDS.CE

The main objective of the CDS.CE module is to provide a high quality CDS risk check as a CE-certified medical software module for integration in Clinical/Medical information systems to increase patient security related to medication risks. Please be aware that this is a SUPPORT system only – the final responsibility for the medication must always remain with the prescribing doctor!

The APP offers the following functionalities in human user interface oriented formats:

- Display the result of a CDS check in HTML format
- Display the result of a CDS check in PDF format

The API offers the following functionalities in a JSON/REST-based machine oriented format:

- Do a CDS check and get a summarized result (to display a single icon summary)
- Do a CDS check and get a detailed result (to display results for each medication)
- Get lists of risks by check type or query keyword (to simplify proper patient risk encoding)

To properly use these applications, they must be integrated with a medication editor in a primary system. To showcase such use-cases, to have a resource-rich functionality and to provide software modules for various use cases, the Documedis platform features a (non-certified) medication module. This includes various functionalities to import, setup, edit, print and transfer a medication plan. It also includes a function to verify a medication plan using the (CE-certified) CDS check module.

3.1. Product Status

The initial release 2018-01 of the CDS.CE module runs as follows:

- All CDS data is production ready. It already covers most of the relevant medications while our editorial team is hard at work to add the remaining data for the most current medications in the Swiss hospital market.
- The CDS.CE software module is a CE-certified medical device since early 2018.
4. Clinical Decision Support – The basic concept

Clinical Decision Support (CDS) offers a set of medical data to support the doctor during the medication process and to add quality assurance to this process. Everything centers on the patient and his wellbeing; medication can then be optimally adjusted to his personal needs and prescription errors might be reduced.

The CDS data in Documedis consists of a huge set of possible risk parameters from various medication domains plus adjusted calculation parameters for an optimum dosage. The data is prepared by the scientific editors of HCI Solutions AG. In this process, the various drugs are systematically encoded and then checked for diverse prescription risks such as absolute and relative contra-indications. Additionally, dosage information is converted to structured data.

Currently, this CDS data can be used to check for the following patient-medication risks:

- Allergies (and cross-allergies)
- Reproduction (conceiving, pregnancy, lactation)
- Dosage warning (in case of kidney or liver insufficiencies)
- Maximum Dosages (for various dosages such as single dose and daily dose)
- Nutrition (food interactions)
- Doping (Illegal substances in sport)
- Special patient groups (elderly)

The CDS services also come with interactions and double dosage tests, as available in INDEX elements such as INTERACTIONS and PRD/GENGRPCD.

With CDS software, individual patient risks (e.g. a penicillin allergy) or personal properties (e.g. pregnancy) are being encoded by the doctor and then saved in the patient dossier. During the prescription process, the information system compares these personal risk parameters with the risk parameters of the individual drugs prescribed. If such a parameter matches, this result hints to a possible risk. The software then presents these results to the doctor, e.g. using the CDS risk and relevance icons and displaying the individual warning texts (these are available in German and French).
### 4.1. The CDS check types

All in all, the CDS data currently allows you to check the following types of risks and various types of dosages for the majority of the drugs relevant in the Swiss health market. New data is being added daily, detailed statistics are available upon request.

In addition, the CDS software includes additional checks using other data from INDEX and one external source.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Check Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🌱</td>
<td>allergyExcipient</td>
<td>Excipient allergies and intolerances: warns about drug risks related to excipients.</td>
</tr>
<tr>
<td></td>
<td>Caution: incomplete best-effort, as excipients are not always available in the data sources (Fachinformation).</td>
<td></td>
</tr>
<tr>
<td>🌱</td>
<td>allergySubstance</td>
<td>Substance allergies and intolerances: warns about allergic and pseudo-allergic reaction and cross-reactions to substances. Additionally, checks are made for intolerances, e.g. fructose intolerance. However, there is no check for products of supplemental therapy directions such as homeopathy.</td>
</tr>
<tr>
<td>🚴</td>
<td>doping</td>
<td>Drugs related to doping: For athletes, the prescribed drugs can be cross-checked for non-permitted substances as listed by WADA.</td>
</tr>
<tr>
<td>💉</td>
<td>doubleMedication</td>
<td>Double medication / posology issues: warns about using multiple medications containing the same or similar substances.</td>
</tr>
<tr>
<td>🚗</td>
<td>driving</td>
<td>Driving / operating a machine: warns about the influence of a medication when driving a car or operating a machine.</td>
</tr>
<tr>
<td>🎽</td>
<td>elderly</td>
<td>Medication for elderly people: warns about risks related to age</td>
</tr>
<tr>
<td>🔮</td>
<td>interaction</td>
<td>Drug-Drug interactions (Datasource ABDATA, pharmaSuisse, HCI)</td>
</tr>
<tr>
<td>🔮</td>
<td>interactionFlycicleCH</td>
<td>Drug-Drug interactions (Datasource AiDKlinik / Flycicle). Only available with hospINDEX, needs a separate subscription option.</td>
</tr>
<tr>
<td>🍗</td>
<td>liverInsufficiency</td>
<td>Liver insufficiency: warns about possible individual dosage adjustments for patients with a liver insufficiency.</td>
</tr>
<tr>
<td>🍗</td>
<td>nutrition</td>
<td>Nutrition: Interactions between drugs and food.</td>
</tr>
<tr>
<td>🍗</td>
<td>posology</td>
<td>Double Medication and maximum dosage</td>
</tr>
<tr>
<td>🍗</td>
<td>renalInsufficiency</td>
<td>Renal insufficiency: warns about possible individual dosage adjustments for patients with a kidney insufficiency.</td>
</tr>
<tr>
<td>⚖️</td>
<td>reproduction</td>
<td>Reproduction: warns about drug risks related to reproduction. This includes specific risks during pregnancy and lactation but also those to women of childbearing potential and men of fathering age in general</td>
</tr>
</tbody>
</table>
4.2. The CDS relevancies

Each CDS risk for a product is encoded with a relevance code RLV and an associated icon. This tells the user how grave the risk is and how he should adapt the treatment (in addition, each risk also features additional detail information). The Display Level defines the display/sort priority to be used if you have to decide which relevance shall be displayed first or in aggregates and is also used in the “hideAbove” variable of the API calls.

To optimize the user experience, please reuse these icons in your software! Using them is mandatory, due to safety and risk reasons. The icons are available for free to our INDEX customers as SVG, PNG (40x40) and ICO (16/24/32) files through the resources listed in the chapter “6.2.13 Additional tools”.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Relevance RLV (CDS software)</th>
<th>RLV Code (CDS data)</th>
<th>Display Level</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Technical problem" /></td>
<td><strong>Service not available.</strong> Due to technical reasons, this check is not available. Please contact your technical support.</td>
<td>500</td>
<td>0</td>
</tr>
<tr>
<td><img src="image" alt="High risks" /></td>
<td><strong>Stop! Contraindication!</strong> One or more of the data sources explicitly mention an absolute contraindication of the product for this risk type!</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td><img src="image" alt="High risks" /></td>
<td><strong>Caution! Major risk!</strong> One or more of the data sources explicitly mention a relative contraindication of the product for this risk type.</td>
<td>2</td>
<td>200</td>
</tr>
<tr>
<td><img src="image" alt="Potential risks" /></td>
<td><strong>Stop! Not enough input!</strong> The data sources for this product contain a relevant check, but the patient data is incomplete due to missing risk parameters. Please complete your patient data.</td>
<td>-1</td>
<td>300</td>
</tr>
<tr>
<td><img src="image" alt="Potential risks" /></td>
<td><strong>Caution! No data available!</strong> The data sources of this product were not yet researched for this risk, therefore no automatic check is possible. (The more the CDS data grows, the less this will appear).</td>
<td>0</td>
<td>400</td>
</tr>
<tr>
<td><img src="image" alt="Low risks" /></td>
<td><strong>Caution: Known risk.</strong> One or more of the data sources explicitly mention a light relative contraindication of the product for this risk type.</td>
<td>3</td>
<td>500</td>
</tr>
<tr>
<td><img src="image" alt="Low risks" /></td>
<td><strong>No information.</strong> The data sources explicitly contain no information about this risk type for the product.</td>
<td>6</td>
<td>600</td>
</tr>
<tr>
<td><img src="image" alt="Low risks" /></td>
<td><strong>Conflicting scientific results.</strong> The data sources contain conflicting information about this risk type of the product (only for check &quot;interactionFlycicleCH&quot;)</td>
<td>70</td>
<td>700</td>
</tr>
<tr>
<td><img src="image" alt="No risks" /></td>
<td><strong>No Risk known.</strong> None of the data sources used by the medical editors at HCI mention any kind of risk of this type.</td>
<td>99</td>
<td>800</td>
</tr>
<tr>
<td><img src="image" alt="Nothing to display" /></td>
<td><strong>Not applicable (n/a).</strong> This product is not relevant for this CDS check.</td>
<td>100</td>
<td>1000</td>
</tr>
<tr>
<td><img src="image" alt="Nothing to display" /></td>
<td><strong>Above threshold.</strong> The check resulted in a risk above the desired display priority.</td>
<td>null</td>
<td>null</td>
</tr>
</tbody>
</table>
5. Data vs. Software-as-a-Service (SaaS)

To offer CDS through different integration variants, it is available both as INDEX data and a CE-certified software service, the Documedis CDS.CE module.

You can therefore either download the INDEX CDS data as usual, then integrate it into your database and program the CDS check yourself. This is the longest, riskiest integration process, the largest effort and highest investment cost – but allows perfection in-process integration and independence. You can still use our Documedis CDS.CE SaaS for testing and quality purpose (verification).

However, with CDS as SaaS using our new Documedis web platform, we offer additional usage scenarios that allow for much faster integration, at low risk and low cost. The exact scenario depends on the IT system, where you encode the patient, his risks and his medication:

- Normally, in a larger professional environment, this will be your local IT system (1). You only need to extend this to encode patient risks based on our free, public CDSCODE list of risks available as JSON objects through a public REST API (4). You then call either
  - The CE-certified CDS check webservice (7) to get a JSON data object as the test result, either as a quick summarized check or a more detailed result. You then adjust your GUI according to the check results, using our free icon sets.
  - The CE-certified CDS check webapplication (5) to get a HTML webpage with the test result, so that you don’t need any additional investment in software functionality.
  - The CE-certified CDS check webapplication (6) to get a PDF document as the result, so that you can archive/display the check result without any additional investment in software functionality.

- As an alternative, for demos or for very basic use cases for users without their own medication editor (1), the (non-certified!) medication module of Documedis includes an online editor function, the “Medicationplan” (2). This allows owners of a Swiss-Rx-Login (or another accepted identity provider such as HIN) to edit patients and medications online and use the CDS checks (5) and CDS reports (6) from the CE-certified CDS module. In that use case, please be aware that using the medication editor is not certified and not part of the CDS.CE module!
6. Functions of the CDS.CE module

6.1. Using and understanding the CDS module

Prerequisites: A modern browser that supports HTML5.

The CDS.CE module combines API (with JSON REST API) and APP (a HTML-based single-page application) functionalities. The API offers a number of web services for CDS checks, while the APP offers a number of functionalities for data viewing and PDF generation by end users.

The APP does not offer any stand-alone functionalities, and can therefore not be explored directly. The best method to learn about it is by working with the Documedis medication module, as this offers an integrated CDS check function using the CDS.CE module. Please refer to the separate documentation “Documedis Medication”.

The following documentation is therefore primarily an explanation on what the CDS.CE module offers.

6.1.1. ClinicalDecisionSupport (5) – the CDS check viewer

For each risk type and medication of the patient, the CDS check shows an individual result using a standard set of icons. As a user, you can switch between the different checks and the different products.

Typical integration scenario examples on how to call the CDS app:
- From your own IT system (1), providing properly encoded patient and medication data in your request.
- From the eMediplan (2) of the Documedis medication module, where the user edits patient data.

In addition, it is also possible to print the CDS itself as a report in PDF format.
6.2. Integrating the Documedis medication software into your clinical information system

For the target architecture, we strongly suggest that you manage patients, their CDS risks and medication plans inside your own local IT system (1), then call our CDS services (7) and viewers (5/6) as needed.

The Documedis services must be called through SSL-secured HTTP POST and GET requests, depending on the exact functionality. For the medication editor and the CDS check, you must POST a specific JSON object to a defined URL and include a number of headers, either as request headers or, if impossible due to technical reasons, as FORM fields. In the latter case, you must put the JSON into a special field named documedisJSON.

The details of the specific request JSON are specified per APP and per API route URL, in this documentation and/or in the Swagger-based API docs of each API instance. The headers are defined globally.

6.2.1. General information: Request header fields for Identification/Authorization/Styling

Please use the following headers in your requests, so that we can properly identify your system and style the UI.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Content / Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorization*</td>
<td>Bearer token, e.g. &quot;Bearer yourBase64TokenReceivedFromHCI&quot;</td>
</tr>
<tr>
<td>(When used as Header field)</td>
<td>Each INDEX customer with a professional level subscription (or a software company contract) must use his INDEX-login to generate access tokens for his organization on the INDEX website. It is a good practice to generate separate tokens for each instance using Documedis and use proper descriptions. The token description will be shown in the footer of the HTML and PDF views. Depending on your current contract details, HCI might also provide you directly with a token.</td>
</tr>
<tr>
<td>access_token</td>
<td>(When used as Form parameter)</td>
</tr>
<tr>
<td>HCI-CustomerId*</td>
<td>**The identifier of the INDEX subscriber using this software instance. See &quot;6.2.7 Access control&quot; for more. If the end user has an INDEX login, this is the numeric part of his INDEX username. If the end user does not have an INDEX login, but only has a subscription, this is the GLN.</td>
</tr>
<tr>
<td>HCI-Index*</td>
<td>**The INDEX variant that shall be used by the service (e.g. &quot;hospINDEX&quot;). Casing is irrelevant. Used by HCI to verify with internal subscriber data (check if CustomerId has matching INDEX subscription).</td>
</tr>
<tr>
<td>HCI-SoftwareOrgId*</td>
<td>**The PartnerId of the software company responsible for this software. Must have INDEX contract. The number part of your <a href="mailto:EPNxxxxxx@hcisolutions.ch">EPNxxxxxx@hcisolutions.ch</a> account. Might be verified by HCI with subscriber data to check if the software company indeed has the necessary access level.</td>
</tr>
<tr>
<td>HCI-SoftwareOrg*</td>
<td>**The Description of the software provider responsible for this software, must have an INDEX software provider subscription. The manufacturer of the application (e.g. &quot;YourCompany GmbH&quot;) [base64].</td>
</tr>
<tr>
<td>HCI-Software*</td>
<td><strong>The name of the application instance / installation that is calling the HCI services (e.g. &quot;KIS Unispital Bern&quot; or &quot;Medfolio KSSG&quot;). Needed to identify the system in case of usage problems. Name must be known by IT-responsible of the installation operator as identified in HCI-CustomerId.</strong> [base64]. Good practice would be to use the same string as the one stored in the token description.</td>
</tr>
<tr>
<td>HCI-UserId</td>
<td>**The validated GLN of the HealthCareProfessional or the validated (!) Swiss-Rx-Login of the user of your system or the internal identifier of the user in the system described in the HCI-Software field. For personalized functions (e.g. 7601001234567 or <a href="mailto:maxmiller@insel.ch">maxmiller@insel.ch</a> or mm63) [base64].</td>
</tr>
<tr>
<td>HCI-UserName</td>
<td>**Display name of the user (e.g. &quot;Max Miller&quot;) [base64]</td>
</tr>
<tr>
<td>HCI-UserAccessType</td>
<td>**Professional &quot;level&quot; of the end user using the system, depending on his qualification and the security level (&quot;trustworthiness&quot;) of that data. See chapter &quot;Access Control&quot; for details.</td>
</tr>
<tr>
<td>HCI-UserAuthenticationLevel</td>
<td>**This string can be displayed by HCI to an APP end user, in the HTML or PDF footer. **</td>
</tr>
</tbody>
</table>
The header field values must use ASCII encoding according to the standard. Exempt are the few custom "HCI-*" fields that optionally also accept [base64] encoding; these must use UTF-16 (Windows/.NET default). Use the base64 encoding if your values can contain characters outside the ASCII range, such as umlauts. In such a case, provide the base64-string inside square brackets (ASCII 91 for [“ and ASCII 93 for ”]) as value.

There is a special use case where you can't use pure REST and put header fields in the HTTP request due to technical reasons: when calling the APP, the Documedis CDS HTML viewer, from your own web page. In such a case, our infrastructure supports that you can instead post all necessary "HCI-*" header fields and the token as FORM fields / key value pairs (<form method='POST' enctype='text/plain'>), using field names identical to the header fields. However, some small adjustments are necessary:

- The Authorization header field must instead be called "access_token" and contain only the token without any "Bearer" prefix (as defined in the oAuth2 RFC).
- The JSON data defining the request content can't be placed directly in the body of the request, but instead must also be transmitted as the value of another form field, per definition called "documedisJSON".
- There is no need for base64-encoding of the field values, the encoding remains as UTF-8 (which is the default for JSON).
- The “Accept-Encoding” header depends on the settings of the client browser.
- The “Accept-Language” header can be replaced by adding the language to the APP route, as the first element of the URL (e.g. https://ce.documedis.hcisolutions.ch/cds/2018-01/app/fr or /de-CH).
- The “Accept” header is not directly relevant, as it is only used in a sub use-case of the one mentioned above, when you want to generate a PDF instead of the HTML. In that case, the accept header is used only inside the JavaScript ajax call to generate the PDF.

6.2.2. The usage of the CDS functions of the medication module in your local IT system

While the user is working on the current medication in his local system, the IT system might want to check the CDS result continuously – or at the end of the process, when the user wants to save the plan. To do this, use the “CDS check summarized” webservice (7), which will result in a global relevance value (good to display a simple major relevance result such as enabling/disabling a button or showing a warning icon) and an individual result for each desired check and product (good to display icons per product or per check in the local UI).

The complete “CDS check detailed” webservice (7) can be used if you also want to display detail information about the checks in the local UI – or if you want to completely rebuild the CDS check display.

If you rebuild the UI on your side, you MUST reuse the risk icons and relevance icons provided by HCI! They are available for free as SVG / ICO / PNG files. As users move between employers and different IT systems, risk-relevant iconography must be kept identical for maximum process security!

The "ClinicalDecisionSupport" tab (5) of the medication module is handy to show the result of the CDS check in a webpage, with all display, layout and interactivity functions provided by our SPA. This saves considerable programming effort.

Additionally, calling the same URL route with a request header "Accept: application/pdf" will a PDF-based CDS report for local storage or display.
6.2.3. The CHMED medication plan object – the core of the transmitted data

To be able to use the Medication/CDS services and viewers, the local IT system must be able to pass the medication plan to our services using the open CHMED format of the IG eMediplan. To do this, the local medication plan must use standardized identifiers for some of the data elements:

- The drugs of the patient medication must be encoded using an established identifier such as the GTIN from GS1 or the Pharmacode or Productnumber from HCI INDEX databases such as hospINDEX or medINDEX or “freetext” drugs (no CDS checks can be performed for these). Dosage, dosage unit and route of administration (ROA) must be encoded using appropriate INDEX data.
- The risks of the patient must be encoded using the free, public risk catalog CDSCODE available from HCI Solutions AG as part of a free INDEX software company contract or directly through the Documedis API, under the public URL https://ce.documedis.hcisolutions.ch/cds/2018-01/api/risks
- All other patient data can be converted on the fly to fit into the CHMED object. The major data elements that MUST be provided to enable a successful CDS check are: dateOfBirth/BDt, gender, weight and height.

The CHMED object itself is available as an open source specification. In its core, it is a JSON object of the patient/medication data. For transmission, it can be compressed and encoded and a header is added to identify the version details. This results in being a single string for transmission and allows inclusion in QR codes. The IG eMediplan publishes a PDF documentation of the format and a software library for Microsoft Windows (written in C#, source available on request) to help construct and serialize/deserialize the object. All necessary information is available from the website of the IG eMediplan (http://emediplan.ch).

In addition, you can use the editor of the Documedis Medication module to create your own CHMED files. For this, first use the editor to create the desired plan. Then access a hidden export functionality: select the EXPORT button, then use the Shortcut “Ctrl-e” to download the CHMED file of the current medication plan.

While Documedis currently only support the CHMED16A JSON variant, we plan to extend support to the FHIR-based CHMED16AF in the future.

Patient risk data

The most critical extension needed in your existing primary system is the ability to properly assign the various risks to a patient and to encode the medication.

To enable risk encoding, you should add various user controls to your patient data user interface (UI), one for each of the relevant CDS check / risk type. The detailed implementation depends on your needs and the number of different risks available to encode a certain check type:

- Some checks / risks are simple “binary” value lists with just a single item (has/hasNot, true/false) where a checkbox or switch UI control might be most appropriate (e.g. doping, driving).
- Other checks / risks consist of a small number of possible values, where a dropdown-list or a few radio buttons might make the most sense (e.g. liverInsufficiency, renalInsufficiency, reproduction).
- Finally, there are a few checks / risks where the selection comes from such a large list of available risk that some sort of search/add/remove functionality might be needed (allergyExcipient, allergySubstance).

The implementation in the Documedis Medication module might be handy to get an idea on how an integration could look (https://documedis.hcisolutions.ch/2018-01/app/medication). In addition, Documedis includes API methods for easy risk selection, e.g. by supporting allergy searches.

The implementation in the Documedis Medication module might be handy to get an idea on how an integration could look (https://documedis.hcisolutions.ch/2018-01/app/medication). In addition, Documedis includes API methods for easy risk selection, e.g. by supporting allergy searches.
For the following CDS check types, patient risk information is needed by the CDS.CE module and must therefore be assigned to the patient and be encoded in his CHMED medication object:

<table>
<thead>
<tr>
<th>Check Type</th>
<th>Necessary risks / risk IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>allergyExcipient</td>
<td>0…n excipient allergies and intolerances of the patient by ID, as available through INDEX/CDSCODE or the Documedis API as risks of this check type.</td>
</tr>
<tr>
<td>allergySubstance</td>
<td>0…n substance allergies and intolerances of the patient by ID, as available through INDEX/CDSCODE or the Documedis API as risks of this check type.</td>
</tr>
<tr>
<td>doping</td>
<td>For athletes, the (single) risk ID 580 must be included.</td>
</tr>
<tr>
<td>driving</td>
<td>For drivers or machine operators, the (single) ID 615 must be included.</td>
</tr>
<tr>
<td>liverInsufficiency</td>
<td>The level of liver insufficiency of the patient by ID, as available through INDEX/CDSCODE or the Documedis API. One of three possible values:</td>
</tr>
<tr>
<td></td>
<td>- 574 Leberinsuffizienz, leichte (Child-Pugh A)</td>
</tr>
<tr>
<td></td>
<td>- 573 Leberinsuffizienz, mittelschwere (Child-Pugh B)</td>
</tr>
<tr>
<td></td>
<td>- 572 Leberinsuffizienz, schwere (Child-Pugh C)</td>
</tr>
<tr>
<td></td>
<td>+ <em>Patient health data (see below), only available if older than 18 years</em></td>
</tr>
<tr>
<td>renallInsufficiency</td>
<td>The level of renal insufficiency of the patient by ID, as available through INDEX/CDSCODE or the Documedis API. One of four possible values:</td>
</tr>
<tr>
<td></td>
<td>- 577 Niereninsuffizienz, leichte (Clcr ≥60–89 ml/min)</td>
</tr>
<tr>
<td></td>
<td>- 576 Niereninsuffizienz, mittelschwere (Clcr ≥30–59 ml/min)</td>
</tr>
<tr>
<td></td>
<td>- 575 Niereninsuffizienz, schwere (Clcr ≥15–29 ml/min)</td>
</tr>
<tr>
<td></td>
<td>- 579 Niereninsuffizienz, terminale (Clcr &lt;15 ml/min)</td>
</tr>
<tr>
<td></td>
<td>+ <em>Patient health data (see below), only available if older than 18 years</em></td>
</tr>
<tr>
<td>reproduction</td>
<td>The level of reproduction risk group of the patient by ID, as available through INDEX/CDSCODE (CHR) or the Documedis API. One of three possible values supported:</td>
</tr>
<tr>
<td></td>
<td>- 612: Frauen im gebärfähigen Alter</td>
</tr>
<tr>
<td></td>
<td>- 78: Schwangerschaft (additionally, the first day of the last menstruation must be included in CHMED16A:Medication/Patient/ Med/DLstMen, using ISO 86013 date format, yyyy-mm-dd)</td>
</tr>
<tr>
<td></td>
<td>- 77: Stillzeit</td>
</tr>
</tbody>
</table>

All risks are available either in INDEX/CDSCODE or through the general (non-CDS) Documedis API (/cds/risks).

The behaviour of the CDS-Check depends on provided patient risks and requested checks:

<table>
<thead>
<tr>
<th>Request</th>
<th>Patient data RiskCategory</th>
<th>Documedis behaviour</th>
<th>RLV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Checktype included</td>
<td>RiskCategory not included</td>
<td>-1: Not enough input data The system assumes that the check was requested (should be done) but that there was no relevant information about this risk provided with the patient data (which points to data missing on the patient)</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>RiskCategory is included, but empty (with no individual risks listed)</td>
<td>Protocol: Risk excluded The system assumes that the check was requested but that the patient does not have any such risks, because the request explicitly lists the RiskCategory, but does not include any such risk.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RiskCategory is included and risks are listed</td>
<td>Relevant Risks are searched for and evaluated.</td>
<td>Any possible result.</td>
</tr>
<tr>
<td>Checktype not included</td>
<td>Doesn’t matter</td>
<td>The system does not run that check, as it wasn’t requested</td>
<td></td>
</tr>
</tbody>
</table>
Patient risk encoding is not relevant or needed for the following checks, as these rely only on the medication and/or patient health data such as birthday, weight and height:

<table>
<thead>
<tr>
<th>Check Type</th>
<th>Necessary patient data</th>
</tr>
</thead>
<tbody>
<tr>
<td>elderly</td>
<td>Birthdate (check is only based on patient age, “older than 65 years”)</td>
</tr>
<tr>
<td>doubleMedication</td>
<td>Medication (ID, dosage/unit, ROA)</td>
</tr>
<tr>
<td>interaction</td>
<td>Medication (ID, dosage/unit, ROA)</td>
</tr>
<tr>
<td>interactionFlycicleCH</td>
<td></td>
</tr>
<tr>
<td>nutrition</td>
<td>Medication (ID, dosage/unit, ROA)</td>
</tr>
<tr>
<td>posology</td>
<td>Patient weight in kg and height in cm</td>
</tr>
</tbody>
</table>

**Patient health data**

For the following CDS check types, patient basic information is needed by the CDS.CE module and must therefore be assigned to the patient and be encoded in his CHMED medication object:

<table>
<thead>
<tr>
<th>Check Type</th>
<th>Necessary patient information</th>
</tr>
</thead>
<tbody>
<tr>
<td>elderly</td>
<td>Birthdate (check is only based on patient age, “older than 65 years”)</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/BDt (ISO 8601, yyyy-mm-dd)</td>
</tr>
<tr>
<td>liverInsufficiency</td>
<td>Birthdate (check is only done when patient is older than 18 years)</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/BDt (ISO 8601, yyyy-mm-dd)</td>
</tr>
<tr>
<td></td>
<td>+Patient risk data (see above)</td>
</tr>
<tr>
<td></td>
<td>+Patient medication data (see below)</td>
</tr>
<tr>
<td>posology</td>
<td>Weight in kg (check is only done when patient &gt;40kg and &lt;120kg):</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/Med/Meas/Type=1 (=weight)</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/Med/Meas/Val={weightInKg, e.g. 80}</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/Med/Meas/Unit=2 (=kg)</td>
</tr>
<tr>
<td></td>
<td>Height in cm:</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/Med/Meas/Type=2 (=height)</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/Med/Meas/Val={heightInCm, e.g. 175}</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/Med/Meas/Unit=1 (=cm)</td>
</tr>
<tr>
<td></td>
<td>Birthdate (check is only done when patient is older than 18 years)</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/BDt (ISO 8601, yyyy-mm-dd)</td>
</tr>
<tr>
<td></td>
<td>+Patient medication data (see below)</td>
</tr>
<tr>
<td>renalInsufficiency</td>
<td>Birthdate (check is only done when patient is older than 18 years)</td>
</tr>
<tr>
<td></td>
<td>• CHMED16A:Medication/Patient/BDt (ISO 8601, yyyy-mm-dd)</td>
</tr>
<tr>
<td></td>
<td>+Patient risk data (see above)</td>
</tr>
<tr>
<td></td>
<td>+Patient medication data (see below)</td>
</tr>
</tbody>
</table>
No patient risk or health data needed

For the following CDS check types, no individual patient information is needed by the CDS.CE module. Instead, the focus is on the medication of the patient.

<table>
<thead>
<tr>
<th>doubleMedication</th>
<th>Only based on medication data (article/product)</th>
</tr>
</thead>
<tbody>
<tr>
<td>interaction</td>
<td>Only based on medication data (article/product)</td>
</tr>
<tr>
<td>interactionFlycicleCH</td>
<td>Only based on medication data (article/product, posology, ROA)</td>
</tr>
</tbody>
</table>

This advanced variant of an interaction check does not only check interactions between two medications, but can also check for triplet and quadruplet interactions (between 3 or 4 drugs). In addition, the dosage, quantity unit and route of administration data are used in the interaction calculation to eliminate unnecessary alerts, minimizing overalerting. Therefore, the CHMED of the request must include full patient medication data (taking times, dosage value and unit, route of administration).

| nutrition | N/A (the nutrition-related risks are interactions between foods and drugs. These are not to be encoded on the patient, because they depend on what he eats on a certain day. Instead, the CDS check simply returns the list of all known food interactions for each product/medicine checked and the doctor then informs the patient about the food risks associated with this medication. Values returned correspond to INDEX/CDSCODE (CHN)) |

Patient medication data

Most CDS checks rely on the proper encoding of the patient medication in the CHMED object. As mentioned, drugs must be encoded using an established identifier such as the GTIN from GS1 or the Pharmacode or Productnumber from HCI INDEX databases such as hospINDEX or medINDEX.

For each identified drug, dosage / dosage unit and route of administration (ROA) must be encoded using appropriate INDEX data. The most important things to know:

- The relevant information is found on the PRODUCT level and the related schemas.
- Therefore, for an article, first find its product number in INDEX/ARTICLE/ART/PRDNO.
- The list of possible ROA of an article or product can then be found in INDEX/PRD/CPT/CPTROA/ROA.
- The possible dosage units of an article or product can be found as the sum of the following data sources:
  - The base quantity unit of a product, from INDEX/PRD/CPT/PQTYU
  - Various additional quantity units from PRODUCT_PROPRIETARY_QUANTITY
  - An additional substance unit from PRODUCT_SUBSTANCE_ALTERNATIVE_QUANTITY (the one entry where NSFLAG=1)

In addition, please consult the eMediplan_CHMED16A specification for details about the proper handling of the Medication/Medicament object nodes.

The following examples show extracts with the most relevant data for each check (the extracts themselves are not valid CHMED, as they have been shortened for documentation).
A. interactionFlycicleCH and posology (patient medication with posology, units and ROA)

CHMED16A0{
  "Patient":{
    "FName": "Dora",
    "LName": "Graber",
    "BDt": "1976-01-01",
    "Gender": 2,
    "Medicaments": [
      {
        "Id": "5331339",  // Pharmacode of this medication
        "IdType": 3,  // IdType 3 = Pharmacode (as defined in CHMED spec)
        "Pos": [
          {
            "DtFrom": "2016-11-09",
            "D": [  // Dosage is 1 unit in the morning and 1 unit in the evening in this example.
              1.0,  // Here, the unit is STK ("Stück"), as described in the next field below
              0.0,
              1.0,
              0.0
            ]
          }
        ],
        "Unit": "STK",  // From INDEX/PRD/CPT/PQTYU or PRODUCT_PROPRIETARY_QUANTITY
                          // or PRODUCT_SUBSTANCE_ALTERNATIVE_QUANTITY (where NSFLAG=1)
        "Roa": "PO"  // From INDEX/PRD/CPT/CPTROA/ROA
      }
    ]
  }
}
B. AllergyExcipient / AllergySubstance (patient has penicillin allergy and lactose intolerance)

CHMED16A0{
  "Patient":{
    "FName": "Dora",
    "LName": "Graber",
    "BDt": "1945-11-23",
    "Gender": 2,
    "Rc": [
      {
        "Id": 6,
        "R": [
          571, RiskCategory.R 571 = CCHKEY of penicillin allergy (from CDSCODE)
          235 RiskCategory.R 235 = CCHKEY of lactose intolerance (from CDSCODE)
        ]
      }
    ]
  },
}
C. Posology of reserve drug

In general, only the drugs currently taken by the patient should be transferred to the CDS-Check. If any relevant input is missing or empty, the check will return a relevance of "-1: Not enough input data". As defined in the CHMED spec, for a reserve drug, the quantity and the maximum quantity are mandatory fields.

```
CHMED16A0{
  "Patient":{  
    "FName":"Dora",
    "LName":"Graber",
    "BDt":"1980-01-01",
    "Gender":1,
    "Meas":[
      {
        "Type":1,
        "Val":"72",
        "Unit":2
      },
      {
        "Type":2,
        "Val":"160",
        "Unit":1
      }
    ],
  },
  "Medicaments":[
    {
      "Id":"13709",
      "IdType":4,
      "Pos":[
        {
          "DtFrom":"2017-11-27",
          "InRes":1,  
          "TT":[
            {
              "A":2.0,  
              "MA":5.0
            }
          ]
        }
      ],
      "Unit":"STK",
      "Roa":"PO"
    }
  ]
```

Amount of the medication in units as defined below

Max Amount per cycle (here: one day / default duration (86400 sec.))

From INDEX/PRD/CPT/PQTYU or PRODUCT_PROPRIETARY_QUANTITY or PRODUCT_SUBSTANCE_ALTERNATIVE_QUANTITY (where NSFLAG=1)
6.2.4. Release management, CE-certified modules and APP and API routes

The base URL https://documedis.hcisolutions.ch of the Documedis suite is a moving target, always providing the latest software of the platform. It is therefore not suitable for deep API integration. While once published simple APP routes will continue to work, the API itself is prone to change. Therefore, Documedis also provides release-bound URLs to stable, long term API and APP functions.

The CE-certified modules are NEVER available directly, but are always tied to an individual release to guarantee long-term interface stability. Their base URL is https://ce.documedis.hcisolutions.ch

Each release is named according to a year-month pattern similar to the INDEX releases, e.g. 2018-01 for the initial release. In addition, all CE-certified modules are only available through the separate ce.* domain.

- Therefore, the initial version of the CDS.CE API is available through https://ce.documedis.hcisolutions.ch/cds/2018-01/api
  Using a browser, this will forward you toe /docs, the Swagger-based interactive API documentation.
  As the API is JSON/REST-based, the routes are ordered per-resource (not per module).

- The CDS.CE app is available through https://ce.documedis.hcisolutions.ch/cds/2018-01/app
  Nothing will be shown on this URL, instead you must call this URL using the specified inputs to then receive the CDS view of the check result.

6.2.5. System requirements for the APP

The viewer function of the CDS check result is a pure HTML based webpage. Therefore, it has some system requirements:

- The browser (or embedded browser component) must support HTML5 and JavaScript. For each release, browser support is decided based on Desktop Browser Market Share Switzerland as published by StatCounter. The CE modules support the latest version of the top 3 browsers by marketshare. For enterprise market and software integration usage, the check will always also support the two browsers from Microsoft (IE11, Edge). In release 2018-01, these are Chrome, Firefox, Safari, IE and Edge.

- The minimum screen size to display the HTML page must be 1280x400 pixel. Optimum performance is achieved with FullHD resolution (1920x1080 pixel). Screen resolution support is based on Desktop Screen Resolution Stats Switzerland as published by StatCounter. The CE modules support the smallest variant of the 6 topmost resolutions.
6.2.6. **APP and API for "ClinicalDecisionSupport CDS.CE" (5/6/7)**

For the CE-certified CDS, you can call the interactive check viewer as a webapp and generate a PDF or use the check services to verify the medication risks in the background. Therefore, the CDS.CE check must be integrated in an external third-party system or be used through the Documedis medication module. When calling the CDS.CE module, you must post a CHMED object to the check, so that the user can then view the result of the CDS check as it is returned by our system.

**APP**

Display the CDS check in a modern, interactive HTML view offering a wide array of functionalities (or as a PDF).

URL: [https://ce.documedis.hcisolutions.ch/cds/2018-01/app](https://ce.documedis.hcisolutions.ch/cds/2018-01/app)

<table>
<thead>
<tr>
<th>HTTP HEADER fields</th>
<th>Type</th>
<th>Values</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept:</td>
<td>string</td>
<td>text/html or application/pdf</td>
<td>The format of the CDS check response, HTML or PDF</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HTTP POST JSON object</th>
<th>Type</th>
<th>Values</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>medication*</td>
<td>string</td>
<td>CHMED16A...</td>
<td>CHMED-string of the patient medication</td>
</tr>
<tr>
<td>checks</td>
<td>List of 0…n check</td>
<td>The desired CDS checks and the display level threshold.</td>
<td>Default / if empty or missing: all risks / all levels</td>
</tr>
<tr>
<td>check</td>
<td>string</td>
<td>e.g. allergySubstance, interaction</td>
<td>The name of the check (see chapter 5). Default / if empty or missing: error</td>
</tr>
<tr>
<td>hideAbove</td>
<td>int</td>
<td>e.g. 400</td>
<td>The highest risk display level to display (see chapter 6). Use this to minimize overalerting. If the value is set, the check returns a relevance &quot;rlv&quot; result of null/empty for the selected check type; it will therefore hide all risks lower than the threshold defined by this value. Allowed values: &gt;= 400, not permitted to be lower. &gt;= 500 for liver and renal insufficiencies. Default / if empty or missing: all levels</td>
</tr>
<tr>
<td>targetOrigin</td>
<td>string</td>
<td>&quot;**&quot; for desktop applications &quot;yoursite.com&quot; for iframe integration in web applications</td>
<td>Value used by the CDS viewer to decide on how to transmit back the current height of the HTML view to the parent window using JavaScript: &quot;**&quot;: js window.external.documedisCdsCeHeightChanged &quot;yoursite.com&quot;: js postMessage</td>
</tr>
<tr>
<td>printModes</td>
<td>List of 0…n strings</td>
<td>check, product</td>
<td>If one or both of the allowed values are included in the request, the HTML UI will show a &quot;Print&quot;-Button that allows the printout of the CDS result in different layouts.</td>
</tr>
<tr>
<td>helpUrl</td>
<td>string</td>
<td>&quot;default&quot; &quot;<a href="https://acme.com/cdshelp">https://acme.com/cdshelp</a>&quot;</td>
<td>If a helpUrl value is provided, a &quot;Help&quot; button is displayed in the top right corner of the app. With a value of &quot;default&quot;, this button will point to the generic online documentation of the Documedis CDS. To link to your own docs instead, just provide the desired URL as the helpUrl value.</td>
</tr>
</tbody>
</table>
If unable to do a proper REST call and resorting to using a FORM POST request instead, remember to put this JSON into a field called documedisJSON as part of your key-value pairs inside the form.

Additionally, Documedis provides two PDF printout variants of the CDS check. To use this, simply change the “Accept” header field from text/html (for the webpage) to application/pdf (for PDF) and add a special route segment to the end of your URL, either /check or /product

(resulting in a final URL of e.g. https://ce.documedis.hcisolutions.ch/cds/2018-01/app/medication/cds/check)

You can also make these printout variants available in the HTML view: just extend your input JSON with the element “printModes”, then add one or both of the printout variants (“check”, “product”) to enable PDF generation directly from the UI in the selected layout.
API

Get a summarized or detailed CDS check result, depending on your exact needs. Both methods use exactly the same check algorithm, only the resulting JSON response is a bit different: The summarized check is simply a leaner/smaller version of the detailed version. The summarized check is useful if you just want to display a single icon as a check result summary, perhaps with some additional tooltip information. The detailed check is useful if you want to display your own full CDS result detail; it offers all the details used by the APP to generate the interactive HTML view of the CDS check results as mentioned above.

URL CDS-Check Summarized: https://ce.documedis.hcisolutions.ch/cds/2018-01/api/checks/summarized
URL CDS-Check Detailed: https://ce.documedis.hcisolutions.ch/cds/2018-01/api/checks/detailed

<table>
<thead>
<tr>
<th>HTTP HEADER fields</th>
<th>Type</th>
<th>Values</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accept:</td>
<td>string</td>
<td>application/json</td>
<td>The format of the CDS check response, a JSON object</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HTTP POST JSON object</th>
<th>Type</th>
<th>Values</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>medication*</td>
<td>string</td>
<td>CHMED16A...</td>
<td>CHMED-string of the patient medication</td>
</tr>
<tr>
<td>checks</td>
<td>List of 0...n checks</td>
<td>The desired CDS checks and the display level threshold.</td>
<td>Default / if empty or missing: all risks / all levels</td>
</tr>
<tr>
<td>check</td>
<td>string</td>
<td>e.g. allergySubstance, interaction</td>
<td>The name of the check (see chapter 5). Default / if empty or missing: error</td>
</tr>
<tr>
<td>hideAbove</td>
<td>int</td>
<td>e.g. 400</td>
<td>The highest risk display level to display (see chapter 6). Use this to minimize overalerting. If this value is set, the check returns a relevance &quot;rlv&quot; result of null/empty for the selected check type; it will therefore hide all risks lower than the threshold defined by this value. Allowed values: &gt;= 400, not permitted to be lower. &gt;= 500 for liver and renal insufficiencies. Default / if empty or missing: all levels</td>
</tr>
</tbody>
</table>

Input: JSON

```json
{
  "medication": "CHMED16...",
  "checks": [
    {
      "check": "interaction",
      "hideAbove": 400
    },
    {
      "check": "posology",
      "hideAbove": 400
    }
  ]
}
```

Output: JSON or PDF.

For the JSON, see detailed schema and example in the appendix of this document:
- Global Relevance ("total" worst case)
- Result for each medication (Worst case and per check)
- Result for each check (Worst case)

In addition, the /cds/risks/... methods of the API allow you to access the individual risks as JSON objects, by id, complete or per checktype, to complete your patient data so that it can be used for the CDS checks. List all known risk types or provide a CheckTypes array, e.g. https://ce.documedis.hcisolutions.ch/cds/2018-01/api/risks?CheckTypes=RenalInsufficiency to get just the renal insufficiency risks.
If you integrate these risk lists into your own system, we suggest to cache them locally. The "id" of each risks will remain fixed over its lifetime and serves as primary key (independent of the checktype). See “6.2.3. The medication plan object – the core of the transmitted data” for more information about the patient risk encoding.

**API responses for the CDS-Check**

The result of the detailed CDS check is a complex JSON object that allows you to build a GUI similar to the one offered by the APP. To get a fully documented view of the object with all its properties, use the swagger-based documentation available through [https://ce.documedis.hcisolutions.ch/cds/2018-01/api/docs](https://ce.documedis.hcisolutions.ch/cds/2018-01/api/docs)

In addition, the following diagram describes the result object in a more graphical way which might make it easier to understand. It must be read from left to right, drilling deeper and deeper into the object tree. Large colored points mark a class type that is repeatedly used in different parts of the response; these are the Relevance, the Medicament and the Substance objects.

On the following pages the diagram will be explained using a left and a right half of it, split between the main objects and the MedicationItemCheck details as denoted by the large left brace in the middle of the diagram.

Please remember that it is strongly suggested to only use the summarized check, then take that response to display the CDS check result in general and to proceed to the APP to display the check result detail! This way, you can take full advantage of the CE-certified CDS.CE module of Documedis. If you chose to implement your own GUI using the detailed check, you must ensure the quality of the implementation yourself.

Basically, the “ClinicalDecisionSupportCheckDetailed” result has three main properties:

- The summarized relevance of the check, as described in “4.2 The CDS relevancies”.
- A result perspective based on each MedicationItem of the patient. A list of medications and their risks.
- A result perspective based on each Check type. A list of check and their risks.
Going into the details of the left part of the diagram:

Each MedicationItem consists of

- The Relevance of this MedicationItem
- The Medicament [ ] used in this MedicationItem. The id and description of the drug product itself.
- The various MedicationItemChecks that were performed for this MedicationItem. This is a list of all the checks that were done for this item based on the data available for this product and the risks of the patient.

The availability of some of these properties depends on the type of the check: The FlyCicleChUrl string is only available for this check, and the same goes for Posology, Interactions and DoubleMedications. In all other cases, the details are in the generic list of risks. Details about these specific properties can be found on the next page.

Each Check consists of

- The Relevance of this Check
- The various CheckMedicationItems used in this Check. This is a list of all Medicaments and their Relevance relevant to this check.
- Various details about the check.

The Relevance consists of

- a result code based on the hideAbove filter provided,
- a nonfiltered “true” result code and
- the description of the relevance in the language of the Accept-Language header field of the request.

The Relevance object is also re-used in many of the check details [ ].
Going into the details of the right part of the diagram:

As described above, each MedicationItemCheck can have slightly different properties, depending on the exact type of the check itself.

For the Posology check result:
- The Posology with a list of all Dosages checked
  - With a list of the relevant individual DosageSubstances
    - As defined by the Substance according to a list of all DosageSubstanceMedicaments
      - which consists of each Medicament and its dosage.

For the Interaction check result
- A list of all Interactions, with the details for each:
  - The causing and the reacting substance mechanisms as two lists of InteractionSides
    - For each InteractionSide, the Substance and the list of Medicaments with that substance
      - For each Substance, its ID and description.

For a DoubleMedication check result:
- A list of all DoubleMedications, with the details of each:
  - The Relevance for each Medicament with that risk.

For any other type of Risk check result:
- A list of all Risks, with the details of each.
6.2.7. Access control

Each INDEX customer with a professional level subscription (or a software company contract) must use his INDEX login to generate access tokens for his organization on the INDEX website. It is a good practice to generate separate tokens for each instance using Documedis and use proper descriptions. The token description will be shown in the footer of HTML and PDF views in the APP.

Depending on your current contract details, HCI might also provide you directly with a token.

If you call Documedis and provide a token, it must either match with the one stored on our backend for the HCI-SoftwareOrgId or the HCI-CustomerId field provided in the request.

HCI-CustomerId must contain the identifier of the INDEX subscriber using this software instance:
- If the end user has an INDEX login, this is the numeric part of his INDEX username (e.g. “12345” for a login such as epn12345@hcisolutions.ch). This applies to e.g. hospINDEX or careINDEX or to demo systems of software companies. The token is validated against this ID.
- If the end user does not have an INDEX login, but only has a subscription, either directly (pharmINDEX) or through a software company as reseller (medINDEX), this is the GLN of the pharmacy or the doctor/practice. The token is validated against HCI-SoftwareOrgID.

If the token is valid, we do completely trust your software. This makes you responsible for the contents of all header fields, especially for the HCI-UserAccessType which defines access to critical functionalities such as Rx mode and extended information access.

The content of the field "HCI-UserAccessType" is a three-element code such as e.g. "A-SOFT-dent":
- The first element denotes the general professional level of the user in the medical field. It can be either:
  - A: Academic medical user, permitted by law to prescribe or dispense medication (doctor, pharmacist, dentist, vet)
  - B: Other medical professional, not allowed to prescribe medication (e.g. nurse, physio etc.). Default.
  - C: Service or group account / company or organization "machine" user
  - D: Public user / Anonymous
- The second element (HCI-UserAuthenticationLevel) denotes the authentication level of the user in your system, the level of trust that you have into the authenticity of the user information that you provide to Documedis. This allows us to clearly separate functionalities for more general access (e.g. access to additional documents for a drug) from others that require a very high level of trust (e.g. sending a prescription to a pharmacy).
  - SIGN: The user provided a qualified digital signature according to federal Swiss law (SR 943.03)
  - HARD: The user identity was verified using physical means (HPC Smartcard, RFID or HIN VPN)
  - COMP: The user identity was issued using a company process (e.g. Windows Account / AD)
  - SOFT: The user identity was verified a user/password-based identity provider
  - NONE: The user is anonymous
- For the third (and optional) element (HCI-UserHealthProfession), the [IHP] element, please provide the health profession of the user using the Refdata IHP (Index of Health Professions), using the refdatabase code as published on refdata.ch

Examples (UserAccessType/UserAuthenticationLevel/UserHealthProfession):
- B/SOFT/nurse: A nurse logged into your system using a username/password
- C/SOFT/nurse: Multiple station personnel that log into your KIS with a single shared Windows account
- A/SOFT: A pharmacist logged into your system using Swiss-Rx-Login, but you aren't providing IHP info
- A/HARD/doctmed: A doctor that access your system through a HIN gateway
- A/SIGN: An academic professional that you verified using a qualified digital signature.
6.2.8. Request error codes

Request may fail due to a number of reasons. In such a case, the various HTTP error codes (401, 403 etc.) are used to give a precise error information. In addition, in some cases the response includes additional error details. The online API docs feature a detailed description of the possible errors and the error object model: https://ce.documedis.hcisolutions.ch/cds/2018-01/api/docs/

6.2.9. Error handling / Offline behavior

There can be cases that the APP or the API are not available due to downtime or a technical issue. Therefore, if your software can not connect to one of our URL endpoints or if the endpoint returns a HTTP status other than 200 (OK), you MUST use the provided icon “Service not available” to tell to the end user that the CDS check is currently unavailable. We suggest using short timeouts, e.g. 2 seconds, when connecting to our service (ideally in an async mode). This assures that availability issues are quickly visible and that the issuing system isn't blocked.

6.2.10. SLA / Availability / Performance

Through the SLA that is part of the INDEX contract giving access to Documedis, we will guarantee an availability of 99.5%. Based on our platform experience, an uptime of 99.9% can be expected, but is not guaranteed. Service time is 7x24h. Maintenance windows are in the nights from Saturday to Sunday.

To check current CDS.CE module availability, please verify current uptime status using our external monitoring available through http://stats.pingdom.com/t7myjtazclq9/4192941 (this service tests service availability 1x/minute and is also used for internal alerting and monthly uptime reporting).

Incident response time is 1 hour during office hours (0800h-1700h) on workdays (Monday-Friday), except on public holidays in the Canton of Berne. Response time is the amount of time between your incident report arriving by e-mail (hotline@hcisolutions.ch) or phone (Tel. 058 851 2600) in our Hotline and their acknowledgement.

Incident resolution time is 8 hours during office hours (0800h-1700h) on workdays (Monday-Friday), except on public holidays in the Canton of Berne. Resolution time is the amount of time between your incident report arriving by e-mail (hotline@hcisolutions.ch) or phone (Tel. 058 851 2600) in our Hotline and the resolution of the incident.

Module Performance: Due to the connected nature of the internet, we are unable to guarantee any end-to-end performance values. We do however target the following performance values as measured using the Example Winforms application “DID” (provided in the “Additional tools” section below) running on a machine in our local intranet:

On the API:
- 200 ms to do a detailed or summarized CDS check, using 12/13 checks (not including FlycicleCH)
- 2000 ms when including FlycicleCH

On the APP:
- 200 ms to get a CDS check view, using 12/13 checks (not including FlycicleCH)
- 2000 ms when including FlycicleCH
- Times above + 2000 ms to generate a CDS check PDF
6.2.11. Acceptance Criteria for verification of installation

As Documedis CDS.CE is a certified medical device, we need to be able to verify your system integration. To do this, please proceed as follows:

- Use this documentation to develop a new version of your product that supports CDS integration
- When that new version is ready for testing, contact us through our hotline
- Our product owner will contact you and provide a test plan including a set of test cases (data files) and a test protocol. Do the test according to the test plan and protocol. Fix your product if needed.
- Contact our product owner once more and arrange for a remote session (using Teamviewer, Webex etc.) to go through the test cases together with the product owner, who will fill out a test protocol. Once the test is accepted, the protocol is signed by the product owner and you are good to go. If not, fix and repeat.

This is only relevant for the initial integration, to make sure that you got the basics right. Further updates of your products don’t need to be re-validated, as the correct integration is your own responsibility.

6.2.12. Support organization

HCI Solutions AG does NOT provide end-user support for the Documedis modules.

HCI Solutions AG does only provide customer support to the technical contacts mentioned in the INDEX contracts or the internal CRM tool. Typically, these contacts are the operational SPOC of the software house and/or the INDEX end user such as a hospital, care home, pharmacy or medical practice.

If a customer or a software provider encounters a bug in the software that is attributable to the Documedis module or the data that it is based on, a bug report must be supplied directly to hotline@hcisolutions.ch
6.2.13. Additional tools

We provide a number of supporting software tools to help developers to quickly setup a solution.

Icons and Translations

Please use our free resources to design and label any CDS elements in your solution! If you integrate the CDS webservice, you MUST use our icons and texts, so that the end users switching between solutions always encounter the identical terminologies when working with CDS. This includes the icons for the check types and the relevance icons plus the German and French translations of the associated terms.

Swagger API documentation

The REST API of Documedis provides JSON responses to your requests. The complete API is documented using a state-of-the-art auto generated web interface, based on the popular Swagger style. This allows you to play around with the API, enter example data, execute requests, get real data and see curl examples of your requests. Using 3rd-party add-ons, you can even generate client POCOs directly from this.

Example Winforms application “DID / Documedis Integration Demo”

The free source code for a Windows Forms based “Pseudo-KIS” (including C# source code) shows a typical integration of the CDS services into a 3rd party desktop application. You can select between three example patient records (hardcoded in the app.config), while the app does a background CDS check for the respective patient and displays the check result as a large colorful relevance icon based on our free icon set (included in the solution). Clicking this relevance icon will open the standalone CDS check viewer of Documedis. Clicking the PDF icon will download the CDS report as a PDF.

In addition, there is the possibility to edit a record using the medication editor (and you can even click the "send" button in the Documedis medication editor to send the edited eMediplan back into the example application).

Example Web application

A small standalone "KIS" web application that can be run as a SPA. It works similar to the Winforms application, but the example patient records are hardcoded in the HTML. After selecting a patient, an AJAX call does a summarized CDS check and displays the result. Clicking the result will then open a separate webpage that shows the CDS check viewer provided by Documedis. You can also edit the patient record in the Medication editor and send the modified version back to the SPA. Clicking the PDF icon will download the CDS report as a PDF.

Do not forget to change the example headers to your own values!

The SPA is hosted under our legacy e-mediat domain to be able to demonstrate cross-domain communication.

Postman project

Postman is a handy tool to explore REST APIs such as Documedis.
7. Appendix

7.1. APP UI integration: Receiving notification from the viewer

The APP viewer uses JavaScript to send feedback to the calling system. If the viewer is integrated in an application (either as an iFrame of a web application or an embedded browser control in a desktop application), the application can therefore receive data from the Documedis module.

Use the parameter “targetOrigin” to decide how you want to receive the feedback:

- If you have a desktop application with Documedis in an embedded browser, use a star/asterisk (“*”). The JavaScript code of the viewer will then issue a window.external.METHODNAME(object) call. Depending on the embedded browser control, you can then add a listener to it for this method.

- If you have a web application with Documedis in an embedded iframe, use the origin of your webpage (e.g. “myapp.mycompany.com”). The JavaScript code of the viewer will then issue a HTML5 parent.postMessage (object) call. You can then add an Event listener to your webapplication and parse the messages.

The transmitted object is a very simple JSON object, including message type and value.

7.1.1. For CDS.CE

This module posts back the current height of the page. This is handy for iFrame integration, to e.g. handle scrollbars properly (but probably not so relevant for the desktop application). The event fires every time the height changes (e.g. because of a user opening or closing a detail).

For desktop applications, the JavaScript calls window.external.documedisCdsCeHeightChanged

7.2. Example eMediplan

If you need some initial example data, you can find it in the demo applications or you can use the fictive test example below. It includes a large number of medications with various risks.

CHMED16A1H4sIAAAAAAAC72VTY/aMBCG/8rk10l3NvmA3EoqtqwjQjdQysOlZkUYidO4livLfxdwD3dMIKE LCYvJ6PH7mVWLvSRrlHLGi/p5MnuMIcJ8EQsaKr2aHxcyZr0GiMA4wjdCR7Qp7bMSBk/AU5zzWY9ESgLoBj R/5ZUJuvc26I32CQAoZhJrgjuOEcUr1zMKvVHLjRQglb4Is9skil/3tPphp transmitted 0TICIj6IJndZSYkRzPPkn01i28kzQ 4Bu4xna7aIa626b6a8CbuVLXCLISL3k+fYEdNZTq77IPXOC3TdtW2XS95gpZxQRqzclplTGzGlqCtjYnzDzg ZplgZ0SziFh32L4w8RALYXyYcoXUdGBJhvfbZ6IFif+xGxm+1ItPyO6ctis6ihZFlx14Rz+xVRGjFP7QIVRdtah Zm7LP6zBOCs1/5iZQiw3drr/P5n3HeoY3gEpCxficit6BPaDWFwnHzAyMykWSpbJLi8Rvlv1e4PBwOb4pNz2J9P+ SsU6hqPqUvjt3r3dTMmMZZHzWvCwxesoEp8QURYAX5v9yboL/D1m5/gD5v6gLKPyDrrKua8wWPvcsdvBK39P o6P/X5BXCWq1ytxHXbT9+M8r7uZLZT2bapLwemzPPcOXNHf5/KWCuc5Fcmmkw/REkGW73S3JynG6vrCrvUv eDSZQlUp/3WwJhPqc8GznCw9NGz2OIWOHosvs3F3RbdvfmCmRgwiaAA=
### 8. Version History

<table>
<thead>
<tr>
<th>Version</th>
<th>Changes</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-01.V1.4</td>
<td>Added detailed requirement documentation for the CHMED medication plan object</td>
<td>12.06.2019</td>
<td>twa</td>
</tr>
<tr>
<td>2018-01.V1.2</td>
<td>Added more details to “patient risk data”</td>
<td>24.05.2018</td>
<td>twa</td>
</tr>
<tr>
<td>2018-01.V1.1</td>
<td>Added “Patient risk data” to 6.2.3, added API response for CheckDetailed</td>
<td>20.04.2018</td>
<td>twa</td>
</tr>
<tr>
<td>2018-01</td>
<td>Release version</td>
<td>09.02.2018</td>
<td>twa</td>
</tr>
<tr>
<td>X4.0</td>
<td>Major rewrite, split into two separate modules (Medication / CDS.CE)</td>
<td>13.12.2017</td>
<td>twa</td>
</tr>
<tr>
<td></td>
<td>Updated relevance icon for 6 (No Information)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Added “driving” CDS check type</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Added error handling, system requirements</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Threshold/hideAbove rule: Allowed value: &gt;= 400, not permitted to be lower!</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X3.3</td>
<td>Allow base64 encoding on selected header fields</td>
<td>22.09.2017</td>
<td>twa</td>
</tr>
<tr>
<td></td>
<td>Added optional logo/organization fields for mediplan PDF view</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X3.2</td>
<td>Critical change: “Potential risks” added to relevancies, renumbered display levels!</td>
<td>25.08.2017</td>
<td>twa</td>
</tr>
<tr>
<td></td>
<td>Print Mediplan as PDF</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>New tabs prescription and polymedicationcheck</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Changed naming from “partner” to “serviceprovider”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X3.1</td>
<td>Some minor changes in header fields (*GLN &gt; *Id), document restructured</td>
<td>30.06.2017</td>
<td>twa</td>
</tr>
<tr>
<td>X3.0</td>
<td>Major rewrite for new UI and documedis.hcisolutions.ch architecture</td>
<td>22.06.2017</td>
<td>twa</td>
</tr>
<tr>
<td>X2.4</td>
<td>The individual CDS &quot;checks&quot; are now &quot;check types&quot;. Added icon for &quot;Check not Available&quot;. Minor corrections.</td>
<td>21.02.2017</td>
<td>twa</td>
</tr>
<tr>
<td>X2.3</td>
<td>Redesign of CDS request and response.</td>
<td>16.12.2016</td>
<td>twa</td>
</tr>
<tr>
<td></td>
<td>NULL relevance for check results above threshold</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Renamed REL to &quot;RLV&quot; and risk ID to &quot;check&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Example request</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X2.2</td>
<td>Added relevance 70 and 500, added display order; updated check names</td>
<td>21.11.2016</td>
<td>twa</td>
</tr>
<tr>
<td>X2.1</td>
<td>Initial Draft</td>
<td>05.10.2016</td>
<td>twa</td>
</tr>
</tbody>
</table>