分野間連携データ基盤

詳細設計書

認証・認可



変更来歴

|  |  |  |  |
| --- | --- | --- | --- |
| # | 版数 | 発行年月日 | 変更内容 |
| 1 | 1 | 2022/05/27 | 新規作成 |

目次

[１． はじめに １](#_Toc107335225)

[１．１ ドキュメント体系 １](#_Toc107335226)

[１．２ 適用範囲 １](#_Toc107335227)

[２． 共通内部仕様 ２](#_Toc107335228)

[２．１ 使用プログラム言語、フレームワーク ２](#_Toc107335229)

[３． 認証サーバ ３](#_Toc107335230)

[３．１ 構成 ３](#_Toc107335231)

[３．２ 初期設定 ３](#_Toc107335232)

[３．３ 内部IF ３](#_Toc107335233)

[３．３．１ 内部IF定義 ３](#_Toc107335234)

[３．３．２ 内部IF詳細 ３](#_Toc107335235)

[３．４ Keycloak設定 ３](#_Toc107335236)

[４． 認可サーバ ５](#_Toc107335237)

[４．１ 構成 ５](#_Toc107335238)

[４．２ 初期設定 ５](#_Toc107335239)

[４．３ 内部IF ５](#_Toc107335240)

[４．３．１ 内部IF定義 ５](#_Toc107335241)

[４．３．２ 内部IF詳細 ５](#_Toc107335242)

[４．４ Keycloak設定 ５](#_Toc107335243)

[５． 認可GW ７](#_Toc107335244)

[５．１ 構成 ７](#_Toc107335245)

# はじめに

## ドキュメント体系

本書はシステムの内部設計を記載する。構成ファイル一覧を表 １-1に記載する。

表 １-1詳細設計書構成ファイル一覧

|  |  |
| --- | --- |
| # | ドキュメント名 |
|  | 詳細設計書\_認証・認可.docx（本書） |
|  | 詳細設計書\_認証・認可\_別紙1\_認証サーバ\_内部IF定義.html |
|  | 詳細設計書\_認証・認可\_別紙2\_認証サーバ\_内部IF詳細.xlsx |
|  | 詳細設計書\_認証・認可\_別紙3\_認証サーバ\_Keycloak設定.xlsx |
|  | 詳細設計書\_認証・認可\_別紙4\_認可サーバ\_内部IF定義.html |
|  | 詳細設計書\_認証・認可\_別紙5\_認可サーバ\_内部IF詳細.xlsx |
|  | 詳細設計書\_認証・認可\_別紙6\_認可サーバ\_Keycloak設定.xlsx |

## 適用範囲

本書は2022年10月版の詳細設計を対象とする。

# 共通内部仕様

## 使用プログラム言語、フレームワーク

本システムで使用するプログラム言語、フレームワークを下表に記載する。

表 ２-1 プログラム言語・フレームワーク、文字コード

|  |  |  |
| --- | --- | --- |
| # | 対象 | ソフトウェア(規格) |
|  | プログラム言語 | Python3.8 64bit |
|  | Webフレームワーク | FastAPI |
|  | WebAPI定義 | OpenAPI 3.0 |
|  | 認証・認可サーバ | Keycloak 18.0.1 |
|  | 文字コード | UTF-8 |

# 認証サーバ

## 構成

Docker Composeファイルにコンテナ定義を記載し、それによって以下表に示すコンテナ群を起動する。

|  |  |  |
| --- | --- | --- |
| # | コンテナ | 概要 |
|  | Nginxコンテナ | Webサーバ |
|  | FastAPIコンテナ | REST API |
|  | Keycloakコンテナ | 認証機能 |

## 初期設定

認証サーバを起動する際にシェルスクリプトによって初期設定をする。

DockerfileにおけるENTRYPOINTなどを想定。

初期設定値を以下に示す。

|  |  |
| --- | --- |
| # | 設定値 |
|  | レルム名 |
|  | アクセストークンの生存期間 |
|  | データカタログ作成ツールクライアント画面（支援サービス群）のクライアントID |
|  | データカタログ作成ツールクライアント画面（支援サービス群）の有効なリダイレクトURI |
|  | データカタログ作成ツールAPIサーバ（支援サービス群）のクライアントID |
|  | 来歴管理のクライアントID |
|  | 契約管理のクライアントID |

## 内部IF

ここで内部IFとは、認証サーバクライアント画面から呼ばれるREST APIのことを指す。

画面と内部IFによって行う業務は以下の通り。

|  |  |  |  |
| --- | --- | --- | --- |
| # | 業務 | アクター | 概要 |
|  | 認証サーバの管理 | CADDE運用管理者 | レルムの設定や共通的なクライアントの管理をする |
|  | CADDEユーザの管理 | CADDE運用管理者 | 申請に伴ってCADDEユーザを管理する |
|  | 外部IdPの管理 | CADDE運用管理者 | 外部IdPを管理する |

### 内部IF定義

「詳細設計書\_認証・認可\_別紙1\_認証サーバ\_内部IF定義.html」を参照のこと。

### 内部IF詳細

「詳細設計書\_認証・認可\_別紙2\_認証サーバ\_内部IF詳細.xlsx」を参照のこと。

## Keycloak設定

「詳細設計書\_認証・認可\_別紙3\_認証サーバ\_Keycloak設定.xlsx」を参照のこと。

# 認可サーバ

## 構成

Docker Composeファイルにコンテナ定義を記載し、それによって以下表に示すコンテナ群を起動する。

|  |  |  |
| --- | --- | --- |
| # | コンテナ | 概要 |
|  | Nginxコンテナ | Webサーバ |
|  | FastAPIコンテナ | REST API |
|  | Keycloakコンテナ | 認可機能 |

## 初期設定

認証サーバを起動する際にシェルスクリプトによって初期設定をする。

DockerfileにおけるENTRYPOINTなどを想定。

初期設定値を以下に示す。

|  |  |  |
| --- | --- | --- |
| # | 設定値 | 備考 |
|  | レルム名 | ＿ |
|  | 提供者コネクタクライアントID | ＿ |
|  | 認証サーバのエイリアス | ＿ |
|  | 認証サーバの認可URL | CADDE運用管理者から通知された後 |
|  | 認証サーバのトークンURL | CADDE運用管理者から通知された後 |
|  | 認証サーバのUserInfo URL | CADDE運用管理者から通知された後 |
|  | 連携用クレイアントのクライアントID | CADDE運用管理者から通知された後 |
|  | 連携用クライアントのクライアントシークレット | CADDE運用管理者から通知された後 |

## 内部IF

ここで内部IFとは、認可サーバクライアント画面から呼ばれるREST APIのことを指す。

画面と内部IFによって行う業務は以下の通り。

|  |  |  |  |
| --- | --- | --- | --- |
| # | 業務 | アクター | 概要 |
|  | 認可レルムの管理 | データ提供者 | 認可サーバにはデータ提供者ごとにレルムがあり、各データ提供者はそれぞれの認可レルムを管理する |
|  | 認可の管理 | データ提供者 | 契約なしの認可を管理する |

### 内部IF定義

「詳細設計書\_認証・認可\_別紙4\_認可サーバ\_内部IF定義.html」を参照のこと。

### 内部IF詳細

「詳細設計書\_認証・認可\_別紙5\_認可サーバ\_内部IF詳細.xlsx」を参照のこと。

## Keycloak設定

「詳細設計書\_認証・認可\_別紙6\_認可サーバ\_Keycloak設定.xlsx」を参照のこと。

# 認可GW

## 構成

|  |  |  |
| --- | --- | --- |
| # | コンテナ | 概要 |
|  | Nginxコンテナ | Webサーバ |
|  | FastAPIコンテナ | REST API |