|  |
| --- |
| 富士ソフト 株式会社 |
| 【AIVoiceAnalytics】  API仕様書 |
| 音声解析機能 |
| 第4.01版 |
|  |
| **2018/6/4** |

改版履歴

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版数 | 年月日 | 改定箇所 | 改定内容 | 担当 |
| 1.0 | 2017/12/1 | 全般 | 新規作成 | FSI |
| 2.0 | 2018/2/1 | 全般 | Phase2 改訂 | FSI |
| 3.0 | 2018/4/1 | 全般 | Phase3 改訂 | FSI |
| 3.01 | 2018/4/4 | 音声認識開始、音声認識、音声認識終了API | 履歴、フィラー文字除去対応を追記 | FSI |
| 3.02 | 2018/4/6 | 音声ファイル認識API | 履歴、フィラー文字除去対応を追記 | FSI |
| 3.03 | 2018/4/25 | 音声ファイル認識API | 戻り値に利用時間追加 | FSI |
| 3.04 | 2018/5/8 | 全般 | 「録音」、「音声認識」を「音声解析」に文言統一  （リカイアス関連は音声認識） | FSI |
| 3.05 | 2018/5/29 | 全般 | 赤文字、青文字を黒に変更 | FSI |
| 4.0 | 2018/6/1 | 全般 | Phase4 改訂 | FSI |
| 4.01 | 2018/6/4 | 音声認識再開API | 新規作成 | FSI |

目次

[1. 音声解析開始API 4](#_Toc515892389)

[1.1. 処理概要 4](#_Toc515892390)

[1.2. 処理詳細 5](#_Toc515892391)

[1.3. リカイアスHTTPリクエスト 6](#_Toc515892392)

[1.4. DBアクセス 9](#_Toc515892393)

[1.4.1. テーブル登録 9](#_Toc515892394)

[2. 音声解析更新API 11](#_Toc515892395)

[2.1. 処理概要 11](#_Toc515892396)

[2.2. 処理詳細 11](#_Toc515892397)

[2.3. リカイアスHTTPリクエスト 12](#_Toc515892398)

[2.4. DBアクセス 13](#_Toc515892399)

[2.4.1. テーブル更新 13](#_Toc515892400)

[2.4.2. テーブル登録 14](#_Toc515892401)

[3. 音声解析終了API 15](#_Toc515892402)

[3.1. 処理概要 15](#_Toc515892403)

[3.2. 処理詳細 16](#_Toc515892404)

[3.3. リカイアスHTTPリクエスト 17](#_Toc515892405)

[3.4. DBアクセス 18](#_Toc515892406)

[3.4.1. 検索 18](#_Toc515892407)

[3.4.2. テーブル更新 18](#_Toc515892408)

[3.4.3. テーブル登録 19](#_Toc515892409)

[4. 音声ファイル解析API 21](#_Toc515892410)

[4.1. 処理概要 21](#_Toc515892411)

[4.2. 処理詳細 22](#_Toc515892412)

[4.3. リカイアスHTTPリクエスト 24](#_Toc515892413)

[4.4. DBアクセス 25](#_Toc515892414)

[4.4.1. テーブル登録 25](#_Toc515892415)

[5. リカイアスユーザ辞書登録API 26](#_Toc515892416)

[5.1. 処理概要 26](#_Toc515892417)

[5.2. 処理詳細 27](#_Toc515892418)

[5.3. リカイアスHTTPリクエスト 28](#_Toc515892419)

[5.4. 形態素解析ユーザ辞書ファイルフォーマット 28](#_Toc515892420)

[6. リカイアスユーザ辞書CSV出力API 29](#_Toc515892421)

[6.1. 処理概要 29](#_Toc515892422)

[6.2. 処理詳細 30](#_Toc515892423)

[7. フィラー情報登録API 31](#_Toc515892424)

[7.1. 処理概要 31](#_Toc515892425)

[7.2. 処理詳細 32](#_Toc515892426)

[7.3. 形態素解析フィラー情報ファイルフォーマット 32](#_Toc515892427)

[8. フィラー情報CSV出力API 33](#_Toc515892428)

[8.1. 処理概要 33](#_Toc515892429)

[8.2. 処理詳細 34](#_Toc515892430)

[9. 音声ファイル解析進捗率取得API 35](#_Toc515892431)

[9.1. 処理概要 35](#_Toc515892432)

[9.2. 処理詳細 36](#_Toc515892433)

[10. 音声解析再開API 37](#_Toc515892434)

[10.1. 処理概要 37](#_Toc515892435)

[10.2. 処理詳細 38](#_Toc515892436)

[10.3. DBアクセス 39](#_Toc515892437)

[10.3.1. ~~通話者ログ取得~~ 39](#_Toc515892438)

[10.3.2. ~~通話ログテーブル更新~~ 39](#_Toc515892439)

[10.3.1. 音声解析ログ取得 40](#_Toc515892440)

# 音声解析開始API

## 処理概要

リカイアスの認証処理を行い、音声認識セッションを開始する

表 1‑1　I/F概要：音声解析開始API

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| インターフェース名 | | | | | | | | |
| /{コンテキストルート}/api/speech/start | | | | | | | | |
| **入力値** | | | | | | | | |
| **パラメータ一覧** | | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **型** | | **入力規則** | | **備考** |
|  |  |  | |  | |  | |  |
| **返却値** | | | | | | | | |
| **No** | **項目名** | | **フィールド名** | | **型** | | **備考** | |
|  | 処理結果 | | resultList | | RestResult[] | |  | |
|  | 結果コード | | code | | String | |  | |
|  | メッセージ | | message | | String | |  | |
|  | 詳細 | | detailList | | String[] | |  | |
|  | トークン | | token | | String | |  | |
|  | 音声認識ID | | uuid | | String | |  | |
|  | 音声解析ログID | | speechLogId | | Long | | 音声解析処理終了後に音声解析ログ取得のため | |

## 処理詳細

1. セッションの利用時間をクリアする。
2. ユーザ情報の「リカイアスライセンスID」、「リカイアスモデルID」、「サービス利用ID」と「パスワード」を取得する。以上各項目がヌルまたは空白の場合エラーコードを設定しレスポンスを返す。
3. リカイアス認証を行い、認証トークンを取得する（表 1‑2　リカイアス認証リクエスト内容　参照）。認証に失敗した場合は結果コードにリカイアス認証エラーのコードを設定しレスポンスを返す。
4. リカイアス音声認識の開始を行い、音声認識用のIDを取得する（表 1‑3　リカイアス音声認識開始リクエスト内容　参照）。音声解析開始に失敗した場合はリカイアスの認証を切断し、結果コードに音声解析開始エラーのコードを設定してレスポンスを返す（表 1‑4　リカイアス認証切断リクエスト内容　参照）。
5. 音声解析ログテーブル、利用時間テーブルの登録を行う（表 1‑6　音声解析ログテーブル登録、表 1‑7　利用時間テーブル登録　参照）。
6. 利用時間IDをセッションに設定する。
7. 5.～6.の処理でエラーが発生した場合リカイアスの音声認識終了し認証を切断する（表 1‑5　リカイアス音声解析終了リクエスト内容、表 1‑4　リカイアス認証切断リクエスト内容　参照）。結果コードにエラーコードを設定してレスポンスを返す。
8. ユーザ情報設定の音声保存が有効の場合、以下の処理を行う。
   1. セッション情報に音声データ一時ファイル名を設定する(ファイル名：音声解析ログID\_トークン\_音声認識ID)
   2. 音声ファイル保存用のディレクトリを作成する({音声保存ディレクトリ}/企業ID/音声解析ログID)

※音声保存ディレクトリは設定ファイルに定義する

1. レスポンスにトークン、音声認識ID、音声解析ログIDを設定し、結果コードに正常終了コードを設定してレスポンスを返す。

## リカイアスHTTPリクエスト

表 1‑2　リカイアス認証リクエスト内容

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| URL | | | | | | HTTPメソッド | |
| https://api.recaius.jp/auth/v2/tokens | | | | | | POST | |
| **リクエストヘッダー** | | | | | | | |
| **No** | **キー名** | | **設定値** | | **備考** | | |
|  | Content-Type | | application/json | |  | | |
| **リクエストボディ** | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **設定値** | | **型** | **備考** |
|  | サービス情報 | speech\_recog\_jaJP | |  | |  |  |
|  | 認証ID | service\_id | | 認証用サービスID | | String | リカイアスライセンステーブル |
|  | パスワード | password | | 認証用パスワード | | String | リカイアスライセンステーブル |
|  | セッション有効時間 | expiry\_sec | | 600（音声認識時） | | int | 設定ファイルに定義 |

※リクエストボディはJSON形式

表 1‑3　リカイアス音声認識開始リクエスト内容

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| URL | | | | | | | | | HTTPメソッド |
| https://api.recaius.jp/asr/v2/voices | | | | | | | | | POST |
| **リクエストヘッダー** | | | | | | | | | |
| **No** | | **キー名** | | | **設定値** | | | **備考** | |
|  | | Content-Type | | | application/json | | |  | |
|  | | X-Token | | | リカイアス認証で取得したトークン | | |  | |
| **リクエストボディ** | | | | | | | | | |
| **No** | **項目名** | | **フィールド名** | **設定値** | | **型** | **備考** | | |
|  |  | | energy\_threshold | リクエスト．エネルギー閾値 | | int | 音声と判断する音量のレベルを指定します。0～1000まで指定できます。指定範囲外の値を指定すると無効になります。 | | |
|  |  | | result\_type | nbest | | String | 認識結果の型を指定します。  ”nbest”、”one\_best”、”confnet”を指定できます。  不正な値を指定した場合は  HTTPレスポンス400が返ります。 | | |
|  | モデルID | | model\_id | モデルID | | int | 音声認識に使用するモデルIDを指定します。  ベースモデルまたはユーザ辞書付きモデルのいずれかを指定します。  不正な値を指定した場合はHTTPレスポンス400が返ります。 | | |
|  |  | | result\_count | 1 | | String | 認識結果の候補が取得できる数を指定します。  本値はresult\_typeが”one\_best”の場合は無効です。1～10まで指定できます。1より小さい値を指定した場合は1に、10より大きい値を指定した場合は10に設定されます。 | | |
|  | 音声ログ  情報 | | data\_log | 1 | | String | 認識する音声データを保存するか否かを指定します。0の時は保存せず、1の時に保存します。  ※本値は、契約時に音声データを保存する契約を交わした場合のみ有効です。0より小さい値を指定した場合は0に、1より大きい値を指定した場合は1に設定されます。 | | |

※リクエストボディはJSON形式

表 1‑4　リカイアス認証切断リクエスト内容

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | | | | HTTPメソッド |
| https://api.recaius.jp/auth/v2/tokens | | | | DELETE |
| **リクエストヘッダー** | | | | |
| **No** | **キー名** | **設定値** | **備考** | |
|  | X-Token | リカイアス認証で取得したトークン |  | |
| **リクエストボディ** | | | | |
| なし | | | | |

表 1‑5　リカイアス音声解析終了リクエスト内容

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | | | | HTTPメソッド |
| https://api.recaius.jp/asr/v2/voices/リカイアス音声認識開始で取得した音声認識ID | | | | DELETE |
| **リクエストヘッダー** | | | | |
| **No** | **キー名** | **設定値** | **備考** | |
|  | X-Token | リカイアス認証で取得したトークン |  | |
| **リクエストボディ** | | | | |
| なし | | | | |

## DBアクセス

### テーブル登録

表 1‑6　音声解析ログテーブル登録

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析ログ | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 音声解析ログID | 自動採番 |  |
|  | 更新日時 | システム日時 |  |
|  | 企業ID | ユーザ情報．企業ID |  |
|  | ユーザ ID | ユーザ情報．ユーザID |  |
|  | ユーザ名 | ユーザ情報．ユーザ名 |  |
|  | 音声解析ログ番号 | 企業内の音声解析ログ番号の最大値+１ |  |
|  | 種別 | 0：マイク | 0：マイク 1：ファイル |
|  | ファイル名 | (マイク使用の音声解析時は空欄) | 音声ファイル解析時のファイル名 |
|  | 開始日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 作成日時 | システム日時 |  |
|  | 作成者 | ユーザ情報．ユーザID |  |
|  | 作成者名 | ユーザ情報．ユーザ名 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

表 1‑7　利用時間テーブル登録

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 利用時間 | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 利用時間ID | 自動採番 |  |
|  | 更新日時 | システム日時 |  |
|  | 企業ID | ユーザ情報．企業ID |  |
|  | 機能種別 | 音声解析 |  |
|  | 開始日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 利用時間 | 0 |  |
|  | 作成日時 | システム日時 |  |
|  | 作成者 | ユーザ情報．ユーザID |  |
|  | 作成者名 | ユーザ情報．ユーザ名 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

# 音声解析更新API

## 処理概要

リカイアスに音声データを送信し解析結果を返却する。リカイアスの返却値に最終解析結果が含まれる場合は音声解析ログテーブル、利用時間テーブルを更新する。音声解析ログ詳細テーブルに登録する。

解析結果に対し、形態素解析（Kuromoji）を行い、フィラー文字を除去する。

表 2‑1　I/F概要：音声解析更新API

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| インターフェース名 | | | | | | | | |
| /{コンテキストルート}/api/speech/update/{token}/{uuid}/{voiceId} | | | | | | | | |
| **URLパラメータ一覧** | | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **型** | | **入力規則** | | **備考** |
|  | トークン | token | | String | | 必須 | |  |
|  | 音声認識ID | uuid | | String | | 必須 | |  |
|  | 音声ID | voiceId | | int | | 必須 | |  |
| **入力値** | | | | | | | | |
| 音声データ(音声形式：16bit リニアPCM、サンプリングレート：16000Hz、チャンネル数：1) | | | | | | | | |
| **返却値** | | | | | | | | |
| **No** | **項目名** | | **フィールド名** | | **型** | | **備考** | |
|  | 処理結果 | | resultList | | RestResult[] | |  | |
|  | 結果コード | | code | | String | |  | |
|  | メッセージ | | message | | String | |  | |
|  | 詳細 | | detailList | | String[] | |  | |
|  | リカイアス解析結果 | | analyzeResult | | AnalyzeResult[] | |  | |
|  | 解析結果種別 | | type | | String | |  | |
|  | 解析結果文字列 | | str | | String | |  | |
|  | 開始秒数 | | time | | Integer | | ミリ秒 | |

## 処理詳細

1. リカイアスに音声データを送信し解析結果を取得する。
2. 音声ファイルの長さ（秒数）をセッションの利用時間に加算する。
3. ユーザ情報の音声保存設定が有効な場合、リクエストの音声データを一時ファイルに追加保存する(ファイル名：{一時ファイルディレクトリ/セッション．音声データ一時ファイル名})

※一時ファイルディレクトリは設定ファイルに定義する。

1. リカイアスの解析結果に最終結果(type=RESULT)が含まれる場合は以下の処理を行う。
   1. 解析結果に対し、フィラー文字除去を行う。

* 解析結果の文節に形態素解析（Kuromoji）を行い、フィラー文字の判定を行う。
* フィラー文字と判定された文字を除去し、解析結果に再格納する。
* フィラー除去用の形態素解析インスタンスは全企業共通のものを１つ使う。
  1. 音声解析ログテーブル、利用時間テーブルを更新する。（表 2‑3　音声解析ログテーブル更新、表 2‑4　利用時間テーブル更新　参照）
  2. 音声解析ログ詳細テーブルに解析結果件数分登録を行う。（表 2‑5　音声解析ログ詳細テーブル登録　参照）音声解析ログテーブルの更新に失敗した場合はセッションに解析結果を保存する。音声解析ログテーブルの更新に成功した場合はセッションの解析結果をクリアする。

1. レスポンスにリカイアスの解析結果を設定し、3．の更新処理でエラーが発生した場合は結果コードにエラーコード、それ以外の場合は正常終了コードを設定してレスポンスを返す。

## リカイアスHTTPリクエスト

表 2‑2　リカイアス認証切断リクエスト内容

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| URL | | | | HTTPメソッド |
| https://api.recaius.jp/asr/v2/voices/リクエスト．音声認識ID | | | | PUT |
| **リクエストヘッダー** | | | | |
| **No** | **キー名** | **設定値** | **備考** | |
|  | Content-Type | multipart/form-data |  | |
|  | X-Token | リクエスト．トークン |  | |
| **リクエストボディ** | | | | |
| --バウンダリ文字列CRLF(改行)  Content-Disposition: form-data; name="voice\_id"CRLF(改行)  CRLF(改行)  リクエスト．音声IDCRLF(改行)  Content-Disposition: form-data; name="voice"CRLF(改行)  Content-Type: application/octet-streamCRLF(改行)  CRLF(改行)  --バウンダリ文字列CRLF(改行)  リクエスト．音声データ  --バウンダリ文字列CRLF(改行) | | | | |

## DBアクセス

### テーブル更新

表 2‑3　音声解析ログテーブル更新

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析ログ | | | |
| **更新条件** | | | |
| 音声解析ログID = リクエスト．音声解析ログID | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 更新日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

表 2‑4　利用時間テーブル更新

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 利用時間 | | | |
| **更新条件** | | | |
| 利用時間ID　= セッション 利用時間ID | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 更新日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 利用時間 | セッション．利用時間 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

### テーブル登録

表 2‑5　音声解析ログ詳細テーブル登録

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析ログ詳細 | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 音声解析ログ詳細ID | 自動採番 |  |
|  | 更新日時 | システム日時 |  |
|  | 企業ID | ユーザ情報．企業ID |  |
|  | 音声解析ログID | リクエスト．音声解析ログID |  |
|  | 内容 | 解析結果．解析文字列 | 暗号化 |
|  | 開始秒数 | 解析結果．開始秒数 |  |
|  | 終了秒数 | 解析結果．終了秒数 |  |
|  | 音声有無 | false：無 |  |
|  | 作成日時 | システム日時 |  |
|  | 作成者 | ユーザ情報．ユーザID |  |
|  | 作成者名 | ユーザ情報．ユーザ名 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

# 音声解析終了API

## 処理概要

リカイアスの音声認識を終了し解析結果を返却する。音声解析ログテーブル、利用時間テーブルの更新をする。音声解析ログ詳細テーブル、音声解析音声圧縮キューへ登録を行う。

解析結果に対し、形態素解析（Kuromoji）を行い、フィラー文字を除去する。

表 3‑1　I/F概要：音声解析終了API

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| インターフェース名 | | | | | | | | |
| /{コンテキストルート}/api/speech/end | | | | | | | | |
| **URLパラメータ一覧** | | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **型** | | **入力規則** | | **備考** |
|  | トークン | token | | String | | 必須 | |  |
|  | 音声認識ID | uuid | | String | | 必須 | |  |
|  | 音声ID | voiceId | | int | | 必須 | |  |
| **入力値** | | | | | | | | |
| **返却値** | | | | | | | | |
| **No** | **項目名** | | **フィールド名** | | **型** | | **備考** | |
|  | 処理結果 | | resultList | | RestResult[] | |  | |
|  | 結果コード | | code | | String | |  | |
|  | メッセージ | | message | | String | |  | |
|  | 詳細 | | detailList | | String[] | |  | |
|  | リカイアス解析結果 | | analyzeResult | | String[][] | |  | |
|  | 解析結果種別 | | type | | String | |  | |
|  | 解析結果文字列 | | str | | String | |  | |
|  | 開始秒数 | | time | | Integer | | ミリ秒 | |

## 処理詳細

1. リクエストにトークンと音声認識IDが設定されていないときは2.~4.の処理を行わず、5.から処理を行う。
2. リカイアスから最終結果が得られるまでリカイアスのフラッシュAPIを1秒おきに呼び続ける（表 3‑2　リカイアスフラッシュリクエスト内容　参照）。各通信にかかった時間をセッションの利用時間に加算する。
3. リカイアスの認証を切断する（表 1‑4　リカイアス認証切断リクエスト内容　）。
4. 上記2で取得した解析結果に対し、フィラー文字除去を行う。
   1. 解析結果の文節に形態素解析（Kuromoji）を行い、フィラー文字の判定を行う。
   2. フィラー文字と判定された文字を除去し、解析結果に再格納する。
   3. フィラー除去用の形態素解析インスタンスは全企業共通のものを１つ使う。
5. 音声解析ログテーブル、利用時間テーブルの更新を行う（表 3‑4　音声解析ログテーブル更新、表 3‑5　利用時間テーブル更新　参照）。
6. リカイアス解析結果を音声解析ログ詳細テーブルに登録する（表 2‑5　音声解析ログ詳細テーブル登録　参照）
7. ユーザ情報の音声保存設定が有効な場合、音声データ一時ファイルより音声解析ログ詳細テーブルのデータに対応する音声ファイル作成する。
   1. 音声解析ログ詳細テーブルより音声を生成していないレコードを取得する（表 3‑3　音声解析ログ詳細検索　参照）。
   2. (1)で取得した音声解析ログ詳細のレコード毎に、音声データ一時ファイルから開始秒数～終了秒数までのデータ抽出し音声ファイルとして「{音声保存ディレクトリ}/企業ID/音声解析ログID」に保存する。

ファイル名：音声解析ログ詳細ID.wav

ファイルフォーマット：16bit リニアPCM、サンプリングレート：16000Hz、チャンネル数：1

* 1. 音声解析ログ詳細の音声有無を「true:有」に更新する（表 3‑6　音声解析ログ詳細更新　参照）
  2. 音声データ一時ファイルを削除する
  3. 音声解析音声圧縮キューへ登録を行う（表 3‑7　音声解析音声圧縮キュー登録　参照）

1. レスポンスにリカイアスの解析結果を設定し、それ以外の場合は正常終了コードを設定してレスポンスを返す。

## リカイアスHTTPリクエスト

表 3‑2　リカイアスフラッシュリクエスト内容

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| URL | | | | | | HTTPメソッド | |
| https://api.recaius.jp/asr/v2/voices/リクエスト．音声認識ID/flush | | | | | | POST | |
| **リクエストヘッダー** | | | | | | | |
| **No** | **キー名** | | **設定値** | | **備考** | | |
|  | Content-Type | | application/json | |  | | |
|  | X-Token | | リカイアス認証で取得したトークン | |  | | |
| **リクエストボディ** | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **設定値** | | **型** | **備考** |
|  | 音声ID | voice\_id | | リクエスト．音声ID | | int |  |

※リクエストボディはJSON形式

## DBアクセス

### 検索

表 3‑3　音声解析ログ詳細検索

|  |  |
| --- | --- |
| テーブル名 | 取得項目 |
| 音声解析ログ詳細 | 全て |
| **取得クエリ** | |
| SELECT  \*  FROM  音声解析ログ詳細  WHERE  企業ID＝ユーザ情報．企業ID　AND  音声解析ログID＝リクエスト．音声解析ログID AND  開始秒数 IS NOT NULL AND  終了秒数 IS NOT NULL AND  音声有無＝false:無 | |

### テーブル更新

表 3‑4　音声解析ログテーブル更新

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析ログ | | | |
| **更新条件** | | | |
| 音声解析ログID = リクエスト．音声解析ログID | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 更新日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

表 3‑5　利用時間テーブル更新

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 利用時間 | | | |
| **更新条件** | | | |
| 利用時間ID = リクエスト．利用時間ID AND | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 更新日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 利用時間 | セッション情報．利用時間 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

表 3‑6　音声解析ログ詳細更新

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析ログ詳細 | | | |
| **更新条件** | | | |
| 音声解析ログ詳細ID = 音声を生成した音声解析ログ詳細．音声解析ログ詳細ID | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 更新日時 | システム日時 |  |
|  | 音声有無 | true:有り |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

### テーブル登録

表 3‑7　音声解析音声圧縮キュー登録

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析音声圧縮キュー | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 音声解析音声圧縮キューID | 自動採番 |  |
|  | 更新日時 | システム日時 |  |
|  | 企業ID | ユーザ情報．企業ID |  |
|  | 音声解析ログID | 音声を生成した音声解析ログ詳細．音声解析ログID |  |
|  | 音声解析ログ詳細ID | 音声を生成した音声解析ログ詳細．音声解析ログ詳細ID |  |
|  | 作成日時 | システム日時 |  |
|  | 作成者 | ユーザ情報．ユーザID |  |
|  | 作成者名 | ユーザ情報．ユーザ名 |  |

# 音声ファイル解析API

## 処理概要

アップロードされた音声ファイルの音声解析を行い、解析結果を返却する。

解析結果に対し、形態素解析（Kuromoji）を行い、フィラー文字を除去する。

音声解析ログテーブル、利用時間テーブル、音声解析音声圧縮キューへ登録を行う。

音声ファイルがWAVE形式以外で対応形式（拡張子がmp3、wma、m4a）の場合、WAVE変換処理を行い、解析を行う。

音声ファイル解析の進捗表示のため、セッション上に進捗率を保持する。

表 4‑1　I/F概要：音声ファイル解析API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| インターフェース名 | | | | |
| /{コンテキストルート}/api/speech/fileAnalyze | | | | |
| **入力値** | | | | |
| 音声データ(音声形式：16bit リニアPCM、サンプリングレート：16000Hz、チャンネル数：1) | | | | |
| **返却値** | | | | |
| **No** | **項目名** | **フィールド名** | **型** | **備考** |
|  | 処理結果 | resultList | RestResult[] |  |
|  | 結果コード | code | String |  |
|  | メッセージ | message | String |  |
|  | 詳細 | detailList | String[] |  |
|  | リカイアス解析結果 | analyzeResult | String[][] |  |
|  | 解析結果種別 | type | String |  |
|  | 解析結果文字列 | str | String |  |
|  | 開始秒数 | time | Integer | ミリ秒 |
|  | 音声解析ログID | speechLogId | Long |  |
|  | 利用時間 | time | Long | ミリ秒 |

## 処理詳細

1. セッション上の進捗率をクリアする。
2. アップロードされた音声ファイルを一時領域に保存する（ファイル保存場所は設定ファイルに定義、ファイル名は「システム日時(ミリ秒)\_セッションID」.拡張子）。
3. 音声ファイルの形式のチェックを行う。音声ファイルの形式が不正な場合は、結果コードに音声ファイル形式不正のエラーコードを設定しレスポンスを返す。

WAVE形式以外で対応形式（拡張子がmp3、wma、m4a）の場合、WAVE変換処理（FFMpegを利用）を行い、変換後のファイルを一時領域に保存する。変換前の音声ファイルは削除する。

1. 音声ファイルがWAVE形式で入力形式(音声形式：16bit リニアPCM、サンプリングレート：16000Hz、チャンネル数：1)に一致しない場合は入力形式のフォーマットのWAVEファイルに変換を行う。
2. リカイアス認証を行い、認証トークンを取得する（表 4‑2　リカイアス認証リクエスト内容　参照）。認証に失敗した場合は結果コードにリカイアス認証エラーのコードを設定しレスポンスを返す。
3. リカイアス音声認識の開始を行い、音声認識用のIDを取得する（表 1‑3　リカイアス音声認識開始リクエスト内容　参照）。音声認識開始に失敗した場合はリカイアスの認証を切断し、結果コードに音声解析開始エラーのコードを設定してレスポンスを返す（表 1‑4　リカイアス認証切断リクエスト内容　参照）。
4. 音声解析ログテーブル、利用時間テーブルの登録を行う（表 4‑3　音声解析ログテーブル登録、表 1‑7　利用時間テーブル登録　参照）
5. 一時領域に保存した音声ファイルから音声データを1秒分ずつ取得し、リカイアスに音声データを送信し解析結果を取得する（表 2‑2　リカイアス認証切断リクエスト内容　参照）。各通信にかかった利用時間を計上する。セッション上の進捗率を更新する。進捗率は、リカイアスへのデータの送信状況 「（送信済みデータサイズ　/　音声ファイルのデータサイズ）× 100」とサーバ上での処理状況を加味した0から100までの数値とする。
6. リカイアスの解析結果に最終結果(type=RESULT)が含まれる場合は以下の処理を行う。
   1. 解析結果に対し、フィラー文字除去を行う。

* 解析結果の文節に形態素解析（Kuromoji）を行い、フィラー文字の判定を行う。
* フィラー文字と判定された文字を除去し、解析結果に再格納する。
* フィラー除去用の形態素解析インスタンスは全企業共通のものを１つ使う。
  1. 音声解析ログテーブル、利用時間テーブルを更新する。（表 2‑3　音声解析ログテーブル更新、表 2‑4　利用時間テーブル更新　参照）
  2. 音声解析ログ詳細テーブルに解析結果件数分登録を行う。（表 2‑5　音声解析ログ詳細テーブル登録　参照）音声解析ログテーブルの更新に失敗した場合はセッションに解析結果を保存する。音声解析ログテーブルの更新に成功した場合はセッションの解析結果をクリアする。

1. リカイアスから最終結果が得られるまでリカイアスのフラッシュAPIを1秒おきに呼び続ける（表 3‑2　リカイアスフラッシュリクエスト内容　参照）。各通信にかかった利用時間を計上する。
2. 上記10で取得した解析結果に対し、上記9の処理を行う。
3. セッション上の進捗率を更新する。
4. リカイアスの音声解析を終了する（表 1‑5　リカイアス音声解析終了リクエスト内容　参照）。
5. リカイアスの認証を切断する（表 1‑4　リカイアス認証切断リクエスト内容　参照）。
6. ユーザ情報の音声保存設定が有効な場合、音声データ一時ファイルより音声解析ログ詳細テーブルのデータに対応する音声ファイル作成する。
   1. 音声解析ログ詳細テーブルより音声を生成していないレコードを取得する（表 3‑3　音声解析ログ詳細検索　参照）。
   2. (1)で取得した音声解析ログ詳細のレコード毎に、音声データ一時ファイルから開始秒数～終了秒数までのデータ抽出し音声ファイルとして「{音声保存ディレクトリ}/企業ID/音声解析ログID」に保存する。

ファイル名：音声解析ログ詳細ID.wav

ファイルフォーマット：16bit リニアPCM、サンプリングレート：16000Hz、チャンネル数：1

* 1. 音声解析ログ詳細の音声有無を「true:有」に更新する（表 3‑6　音声解析ログ詳細更新　参照）
  2. 音声データ一時ファイルを削除する
  3. 音声解析音声圧縮キューへ登録を行う（表 3‑7　音声解析音声圧縮キュー登録　参照）

1. レスポンスにリカイアスの解析結果を設定し、正常終了コードを設定してレスポンスを返す。

## リカイアスHTTPリクエスト

表 4‑2　リカイアス認証リクエスト内容

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| URL | | | | | | HTTPメソッド | |
| https://api.recaius.jp/auth/v2/tokens | | | | | | POST | |
| **リクエストヘッダー** | | | | | | | |
| **No** | **キー名** | | **設定値** | | **備考** | | |
|  | Content-Type | | application/json | |  | | |
| **リクエストボディ** | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **設定値** | | **型** | **備考** |
|  | サービス情報 | speech\_recog\_jaJP | |  | |  |  |
|  | 認証ID | service\_id | | 認証用サービスID | | String | リカイアスライセンステーブル |
|  | パスワード | password | | 認証用パスワード | | String | リカイアスライセンステーブル |
|  | セッション有効時間 | expiry\_sec | | 3600（音声ファイル認識時） | | int | 設定ファイルに定義 |

※リクエストボディはJSON形式

## DBアクセス

### テーブル登録

表 4‑3　音声解析ログテーブル登録

|  |  |  |  |
| --- | --- | --- | --- |
| テーブル名 | | | |
| 音声解析ログ | | | |
| **設定値** | | | |
| **No** | **項目名** | **設定値** | **備考** |
|  | 音声解析ログID | 自動採番 |  |
|  | 更新日時 | システム日時 |  |
|  | 企業ID | ユーザ情報．企業ID |  |
|  | ユーザ ID | ユーザ情報．ユーザID |  |
|  | ユーザ名 | ユーザ情報．ユーザ名 |  |
|  | 音声解析ログ番号 | 企業内の音声解析ログ番号の最大値+１ |  |
|  | 種別 | 1：ファイル | 0：マイク 1：ファイル |
|  | ファイル名 | リクエスト．ファイル名 | 音声ファイル解析時のファイル名 |
|  | 開始日時 | システム日時 |  |
|  | 終了日時 | システム日時 |  |
|  | 作成日時 | システム日時 |  |
|  | 作成者 | ユーザ情報．ユーザID |  |
|  | 作成者名 | ユーザ情報．ユーザ名 |  |
|  | 更新者 | ユーザ情報．ユーザID |  |
|  | 更新者名 | ユーザ情報．ユーザ名 |  |

# リカイアスユーザ辞書登録API

## 処理概要

CSVファイルよりリカイアス音声認識用ユーザ辞書の登録を行う。

表5‑1　I/F概要：リカイアスユーザ辞書登録API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| インターフェース名 | | | | |
| /{コンテキストルート}/api/lexicon/update | | | | |
| **入力値** | | | | |
| CSVファイル | | | | |
| **返却値** | | | | |
| **No** | **項目名** | **フィールド名** | **型** | **備考** |
|  | 処理結果 | resultList | RestResult[] |  |
|  | 結果コード | code | String |  |
|  | メッセージ | message | String |  |
|  | 詳細 | detailList | String[] |  |
|  | 一括処理結果 | bulkResultList | BulkResult[] |  |
|  | データ番号 | number | Integer |  |
|  | 処理結果 | resultList | RestResult[] |  |
|  | 結果コード | code | String |  |
|  | メッセージ | message | String |  |
|  | 詳細 | detailList | String[] |  |
|  | データ件数 | dataCount | Integer |  |
|  | 総エラー件数 | errorCount | Integer |  |

表5‑2　インポートファイル入力項目

|  |  |  |  |
| --- | --- | --- | --- |
| ファイル形式 | | CSV形式(1行目をヘッダ行とし、各データを「”」で囲みカンマで区切る) | |
| **文字コード** | | Window-31J | |
| **内容** | | | |
| **No** | **項目名** | | **備考** |
|  | 表記 | | 必須 |
|  | 読み | | 必須 |

## 処理詳細

1. ユーザ情報の「企業ID」、「リカイアスモデルID」、「サービス利用ID」と「パスワード」を取得する。以上各項目がヌルまたは空白の場合エラーコードを設定しレスポンスを返す。
2. CSVファイルを読み込む。エラーがある場合は結果コードにエラーコードを設定してレスポンスを返す。
3. リカイアス認証を行い、認証トークンを取得する（表 1‑2　リカイアス認証リクエスト内容　参照）。認証に失敗した場合は結果コードにリカイアス認証エラーのコードを設定しレスポンスを返す。
4. リカイアスユーザ辞書一括登録APIを呼ぶ（表 5‑3　リカイアスユーザ辞書一括登録リクエスト内容　参照）。エラーのある場合は認証を切断し、結果コードにエラーコードを設定してレスポンスを返す（表 1‑4　リカイアス認証切断リクエスト内容　参照）。
5. リカイアスの認証を切断する（表 1‑4　リカイアス認証切断リクエスト内容　参照）
6. ~~「黒文字(kuromoji)」について、更新した辞書情報でインスタンスを再生成する。~~
7. 形態素解析用のユーザ辞書ファイルを出力する（表5‑4　形態素解析ユーザ辞書ファイルフォーマット）。エラーが発生した場合は、結果コードにエラーコードを設定してレスポンスを返す。
8. 更新件数をレスポンスに設定する。
9. 結果コードに正常終了コードを設定してレスポンスを返す。

## リカイアスHTTPリクエスト

表 5‑3　リカイアスユーザ辞書一括登録リクエスト内容

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| URL | | | | | | HTTPメソッド | |
| https://api.recaius.jp/asr/v2/lexicon/リクエスト．モデルID/update | | | | | | PUT | |
| **リクエストヘッダー** | | | | | | | |
| **No** | **キー名** | | **設定値** | | **備考** | | |
|  | Content-Type | | application/json | |  | | |
| **リクエストボディ** | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **設定値** | | **型** | **備考** |
|  | ユーザ辞書配列 | ulex | |  | | 配列 |  |
|  | 表記 | surface | | リクエスト．表記 | | String |  |
|  | 読み | pron | | リクエスト．読み | | String |  |
|  | 品詞ID | class\_id | | リクエスト．品詞ID | | int | 6固定 |

※リクエストボディはJSON形式

## 形態素解析ユーザ辞書ファイルフォーマット

表5‑4　形態素解析ユーザ辞書ファイルフォーマット

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ファイル形式 | | CSV形式(ヘッダ行無し、各データは「”」で囲まずにカンマで区切る) | | |
| **文字コード** | | UTF-8 | | |
| **内容** | | | | |
| **No** | **項目名** | | **設定値** | **備考** |
|  | 表記 | | リクエスト．表記 |  |
|  | セグメント値 | | リクエスト．表記 |  |
|  | 読み | | リクエスト．読み |  |
|  | 品詞 | | 名詞 | 名詞固定 |

# リカイアスユーザ辞書CSV出力API

## 処理概要

ユーザ辞書を読み込んで、CSVファイル形式で出力を行う。

表6‑1　I/F概要：リカイアスユーザ辞書登録API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| インターフェース名 | | | | |
| /{コンテキストルート}/api/lexicon/file/{ext}/{id} | | | | |
| **パラメータ一覧** | | | | |
| ext ファイル種類(eg:CSV) | | | | |
| id ファイルID(eg:USER\_DICTIONARY) | | | | |
| **返却値** | | | | |
| **No** | **項目名** | **フィールド名** | **型** | **備考** |
| 1 | 処理結果 | resultList | RestResult[] |  |
| 2 | 結果コード | code | String |  |
| 3 | 出力ファイル | USER\_DICTIONARY | StreamingOutput |  |

表6‑2　出力CSVファイル列項目

|  |  |  |  |
| --- | --- | --- | --- |
| ファイル形式 | | CSV形式(1行目をヘッダ行とし、各データを「”」で囲みカンマで区切る) | |
| **文字コード** | | Window-31J | |
| **内容** | | | |
| **No** | **項目名** | | **備考** |
|  | 表記 | | 必須 |
|  | 読み | | 必須 |

## 処理詳細

1. ユーザ情報の「リカイアスモデルID」、「サービス利用ID」と「パスワード」を取得する。以上各項目がヌルまたは空白の場合エラーコードを設定しレスポンスを返す。
2. リカイアス認証を行い、認証トークンを取得する（表 1‑2　リカイアス認証リクエスト内容　参照）。認証に失敗した場合は結果コードにリカイアス認証エラーのコードを設定しレスポンスを返す。
3. リカイアスユーザ辞書登録単語一覧取得する。
4. ユーザ辞書の全てのキーと値を辿り、CSVファイル内容を作成する。エラーが発生した場合は、エラーコードを設定しレスポンスを返す。
5. 結果コードに正常終了コードを設定してレスポンスを返す。

# フィラー情報登録API

## 処理概要

CSVファイルより音声解析用フィラー情報の登録を行う。

表‑1　I/F概要： フィラー情報登録API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| インターフェース名 | | | | |
| /{コンテキストルート}/api/filler/update | | | | |
| **入力値** | | | | |
| CSVファイル | | | | |
| **返却値** | | | | |
| **No** | **項目名** | **フィールド名** | **型** | **備考** |
|  | 処理結果 | resultList | RestResult[] |  |
|  | 結果コード | code | String |  |
|  | メッセージ | message | String |  |
|  | 詳細 | detailList | String[] |  |
|  | 一括処理結果 | bulkResultList | BulkResult[] |  |
|  | データ番号 | number | Integer |  |
|  | 処理結果 | resultList | RestResult[] |  |
|  | 結果コード | code | String |  |
|  | メッセージ | message | String |  |
|  | 詳細 | detailList | String[] |  |
|  | データ件数 | dataCount | Integer |  |
|  | 総エラー件数 | errorCount | Integer |  |

表‑2　インポートファイル入力項目

|  |  |  |  |
| --- | --- | --- | --- |
| ファイル形式 | | CSV形式(1行目をヘッダ行とし、各データを「”」で囲む) | |
| **文字コード** | | Window-31J | |
| **内容** | | | |
| **No** | **項目名** | | **備考** |
|  | フィラー文字列 | | 必須 |

## 処理詳細

1. ~~ユーザ情報の「企業ID」、「リカイアスモデルID」、「サービス利用ID」と「パスワード」を取得する。以上各項目がヌルまたは空白の場合エラーコードを設定しレスポンスを返す。~~
2. CSVファイルを読み込む。バリデーションを実施して、エラーがある場合は結果コードにエラーコードを設定してレスポンスを返す。
3. 形態素解析用のフィラー情報ファイルとして上書き保存をする（表‑3　形態素解析フィラー情報ファイルフォーマット）。エラーが発生した場合は、結果コードにエラーコードを設定してレスポンスを返す。
4. 「黒文字(kuromoji)」について、更新したフィラー情報でインスタンスを再生成する。
5. 更新件数をレスポンスに設定する。
6. 結果コードに正常終了コードを設定してレスポンスを返す。

## 形態素解析フィラー情報ファイルフォーマット

表‑3　形態素解析フィラー情報ファイルフォーマット

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ファイル形式 | | CSV形式(ヘッダ行無し、各データは「”」で囲まない) | | |
| **文字コード** | | UTF-8 | | |
| **内容** | | | | |
| **No** | **項目名** | | **設定値** | **備考** |
|  | フィラー文字列 | | リクエスト．フィラー文字列 |  |

# フィラー情報CSV出力API

## 処理概要

フィラー情報を読み込んで、CSVファイル形式で出力を行う。

表‑1　I/F概要：フィラー情報CSV出力API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| インターフェース名 | | | | |
| /{コンテキストルート}/api/filler/file/{ext}/{id} | | | | |
| **パラメータ一覧** | | | | |
| ext ファイル種類(eg:CSV) | | | | |
| id ファイルID(eg:FILLER\_DICTIONARY) | | | | |
| **返却値** | | | | |
| **No** | **項目名** | **フィールド名** | **型** | **備考** |
| 1 | 処理結果 | resultList | RestResult[] |  |
| 2 | 結果コード | code | String |  |
| 3 | 出力ファイル | FILLER\_DICTIONARY | StreamingOutput |  |

表‑2　出力CSVファイル列項目

|  |  |  |  |
| --- | --- | --- | --- |
| ファイル形式 | | CSV形式(1行目をヘッダ行とし、各データを「”」で囲む) | |
| **文字コード** | | Window-31J | |
| **内容** | | | |
| **No** | **項目名** | | **備考** |
|  | フィラー文字列 | | 必須 |

## 処理詳細

1. ~~ユーザ情報の「リカイアスモデルID」、「サービス利用ID」と「パスワード」を取得する。以上各項目がヌルまたは空白の場合エラーコードを設定しレスポンスを返す。~~
2. フィラー情報登録フィラー文字列一覧取得する。エラーが発生した場合は、エラーコードを設定しレスポンスを返す。
3. フィラー情報の全てのフィラー文字列を辿り、CSVファイル内容を作成する。エラーが発生した場合は、エラーコードを設定しレスポンスを返す。
4. 結果コードに正常終了コードを設定してレスポンスを返す。

# 音声ファイル解析進捗率取得API

## 処理概要

音声ファイル解析途中、処理の進捗率を取得する。

表‑1　I/F概要：音声ファイル解析進捗率取得API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| インターフェース名 | | | | |
| /{コンテキストルート}/api/ getProgressRate | | | | |
| **パラメータ一覧** | | | | |
| なし | | | | |
| **返却値** | | | | |
| **No** | **項目名** | **フィールド名** | **型** | **備考** |
| 1 | 処理結果 | resultList | RestResult[] |  |
|  | 結果コード | code | String |  |
|  | メッセージ | message | String |  |
|  | 詳細 | detailList | String[] |  |
| 5 | 進捗率 | progressRate | Float | 値範囲：0~100 |
|  |  |  |  |  |
|  |  |  |  |  |

## 処理詳細

1. セッション上の進捗率を取得する。エラーが発生した場合は、エラーコードを設定しレスポンスを返す。
2. 結果コードに正常終了コードを設定してレスポンスを返す。

# 音声解析再開API

## 処理概要

リカイアスの認証処理を行い、音声認識セッションを開始する

表 10‑1　I/F概要：音声解析再開API

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| インターフェース名 | | | | | | | | |
| /{コンテキストルート}/api/speech/resume | | | | | | | | |
| **入力値** | | | | | | | | |
| **パラメータ一覧** | | | | | | | | |
| **No** | **項目名** | **フィールド名** | | **型** | | **入力規則** | | **備考** |
| 1 | 音声解析ログID | speechLogId | | Long | | 必須 | |  |
| **返却値** | | | | | | | | |
| **No** | **項目名** | | **フィールド名** | | **型** | | **備考** | |
| 2 | 処理結果 | | resultList | | RestResult[] | |  | |
| 3 | 結果コード | | code | | String | |  | |
| 4 | メッセージ | | message | | String | |  | |
|  | 詳細 | | detailList | | String[] | |  | |
|  | トークン | | token | | String | |  | |
|  | 音声認識ID | | uuid | | String | |  | |

## 処理詳細

1. リクエストの音声解析ログIDより音声解析ログ情報を取得する（表 10‑2　音声解析ログ取得　参照）。
2. 音声解析ログ情報が取得できない場合は、結果コードに存在しないエラーのコードを設定しレスポンスを返す。
3. 取得した音声解析ログの企業IDがユーザ情報．企業IDと異なる場合は、結果コードに存在しないエラーのコードを設定しレスポンスを返す。
4. 取得した音声解析ログのユーザIDがユーザ情報．ユーザIDと異なる場合は、結果コードに存在しないエラーのコードを設定しレスポンスを返す。
5. ユーザ情報の「リカイアスライセンスID」、「リカイアスモデルID」、「サービス利用ID」と「パスワード」を取得する。以上各項目がヌルまたは空白の場合エラーコードを設定しレスポンスを返す。
6. リカイアス認証を行い、認証トークンを取得する（表 1‑2　リカイアス認証リクエスト内容　参照）。認証に失敗した場合は結果コードにリカイアス認証エラーのコードを設定しレスポンスを返す。
7. リカイアス音声認識の開始を行い、音声認識用のIDを取得する（表 1‑3　リカイアス音声認識開始リクエスト内容　参照）。音声認識開始に失敗した場合はリカイアスの認証を切断し、結果コードに音声認識開始エラーのコードを設定してレスポンスを返す（表 1‑4　リカイアス認証切断リクエスト内容　参照、表 1‑5　リカイアス音声解析終了リクエスト内容）。
8. ユーザ情報の音声保存設定が有効の場合、セッション情報に音声データ一時ファイル名を設定する(ファイル名：音声解析ログID\_トークン\_音声認識ID)。
9. レスポンスにトークン、音声認識IDを設定し、結果コードに正常終了コードを設定してレスポンスを返す。

## DBアクセス

### 音声解析ログ取得

表 10-2　音声解析ログ取得

|  |  |
| --- | --- |
| テーブル名 | 取得項目 |
| 音声解析ログ | （全て） |
| **取得クエリ** | |
| SELECT  \*  FROM  音声解析ログ  WHERE  音声解析ログID＝リクエスト．音声解析ログID AND  削除日時 IS NULL | |

**保護用紙**