

日々是Oracle APEX

Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

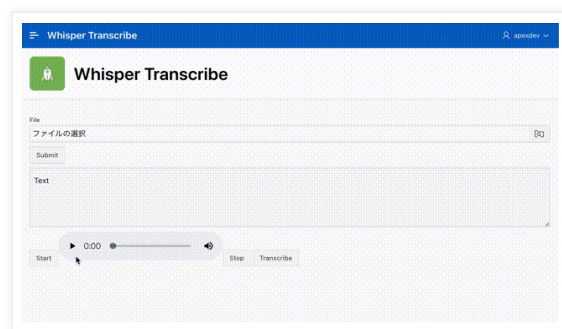
2023年1月24日 火曜日

OpenAI Whisperを使った文字起こしアプリの作成(5) - ブラウザからの音声入力

ブラウザで音声を録音し、そのデータを使って直接WhisperのAPIを呼び出して文字起こしをした結果を、ページ・アイテムに設定する実装を追加してみました。

以下のような動作になります。

Startボタンを押してからStopボタンを押すまでに、結果としてTextに文字起こしされる言葉を録音しています。



Whisperを実行するコードに、CORSの対応を追加します。

Whisper APIを呼び出すページはAPEXのサーバーが生成しています。OriginがAPEXのサーバーであれば、Whisper APIを呼び出せるように許可を与えています。

```
import os
from flask import Flask, request
import uuid
import whisper
model = whisper.load_model("small")

app = Flask(__name__)
@app.route('/transcribe', methods=['POST'])
def transcribe():
    if request.method == 'POST':
        if 'file' not in request.files:
            return 'No file part'

        tempfile = os.path.join('./audio', str(uuid.uuid1()))
        file = request.files['file']
        file.save(tempfile)
```

```
    result = model.transcribe(tempfile)
    os.remove(tempfile)
    return result
```

CORSの対応を追加する。

```
@app.after_request
```

```
def after_request(response):
```

```
    response.headers['Access-Control-Allow-Origin'] = "https://APEXアプリが動作しているホスト名"
```

```
    return response
```

```
if __name__ == "__main__":
```

```
    app.run(host='0.0.0.0', port=8443, ssl_context=('./certs/fullchain.pem', './certs/privkey.p
```

whisper-server.py hosted with ❤ by GitHub

[view raw](#)

次に共有コンポーネントの静的アプリケーション・ファイルを作成します。

ファイル名はpage-actions.jsとします。内容は以下になります。今回は記述するJavaScriptのコードを、ひとつのファイルにまとめています。

```
var mediaRecorder = null;
var mimeType = '';
var chunks = [];
var recording = null; // 録音のBlobデータ

const CONSTRAINTS = {"video": false, "audio": true};

/*
 * 録音を開始する。
 */
const START_AUDIO_RECORDING = {
    name: "start-audio-recording",
    action: function(event, element, args) {
        if (mediaRecorder.state == "inactive") {
            console.log("start audio recording");
            mediaRecorder.start();
        }
    }
};

/*
 * 録音を停止する。
 */
const STOP_AUDIO_RECORDING = {
    name: "stop-audio-recording",
    action: function(event, element, args) {
        if (mediaRecorder.state == "recording") {
            console.log("stop audio recording");
```

```

        mediaRecorder.stop();
    }
}

};

/*
 * Whisper APIを呼び出して、文字起こしを実行する。
 * 文字列はargs.targetとして指定されたページ・アイテムに保存する。
 */
const WHISPER_TRANSCRIBE = {
    name: "whisper-transcribe",
    action: function(event, element, args) {
        if (recording !== null) {
            let formData = new FormData();
            formData.append("file", recording);
            let request = new XMLHttpRequest();

            request.onreadystatechange = () => {
                if (request.readyState === 4) {
                    let response_json = JSON.parse(request.response);
                    $s(args.target, response_json.text);
                }
            };
            request.open("POST", args.url);
            request.send(formData);
        }
    }
}

/*
 * ページ・ロード時に実行する。
 */
window.onload = () => {

    /* アクションの初期化 */
    apex.actions.add([START_AUDIO_RECORDING, STOP_AUDIO_RECORDING, WHISPER_TRANSCRIBE]);

    /*
     * 音声レコーダーの初期化。
     */
    navigator.mediaDevices
        .getUserMedia(CONSTRAINTS)
        .then((stream) => {
            mediaRecorder = new MediaRecorder(stream);
            /* 再生の準備ができたときに呼ばれる */
            mediaRecorder.ondataavailable = (e) => {
                mimeType = e.data.type;
            }
        })
    .catch((error) => {
        console.log("Error: " + error.message);
    });
}

```

```

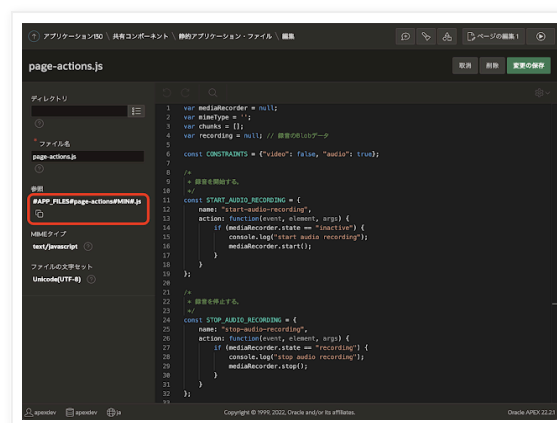
        chunks.push(e.data);
        console.log("data available", e.data);
    };
    /* 録音を停止したときに呼ばれる。 */
    mediaRecorder.onstop = () => {
        recording = new Blob(chunks, {'type': mimeType});
        chunks = []; // 今までの録音は削除。
        player.src = window.URL.createObjectURL(recording);
        console.log("audio recorder stopped.");
    };
    })
    .catch(function (e) {
        alert(e);
    });
    /* onload終了 */
}

```

page-actions.js hosted with ❤️ by GitHub

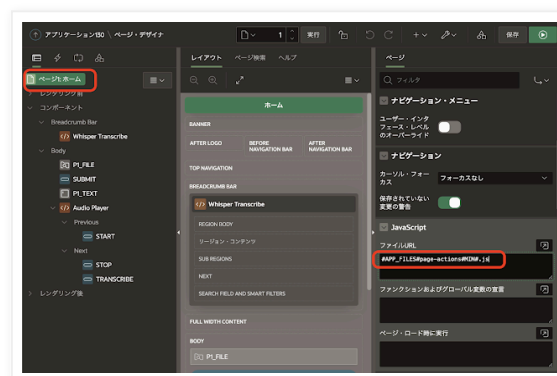
[view raw](#)

参照はコピーしておきます。



ページ・デザイナーにて、ホーム・ページを開きます。

JavaScriptのファイルURLを設定します。



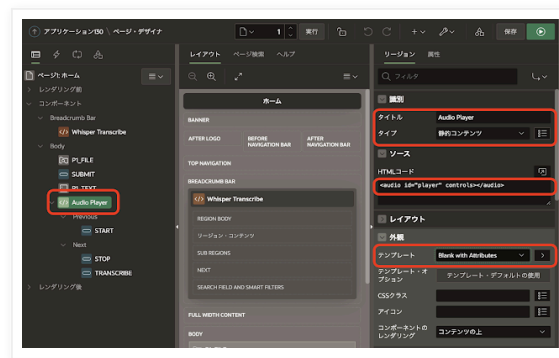
音声レコーダーとなるリージョンと、START、STOP、TRANSCRIBEの3つボタンを作成します。

リージョンを作成します。

識別の**タイトル**は**Audio Player**とします。**タイプ**に**静的コンテンツ**を選択し、**ソース**の**HTMLコード**として以下を記述します。

```
<audio id="player" controls></audio>
```

リージョンの修飾を最小限にするため、**外観**の**テンプレート**として**Blank with Attributes**を選択します。

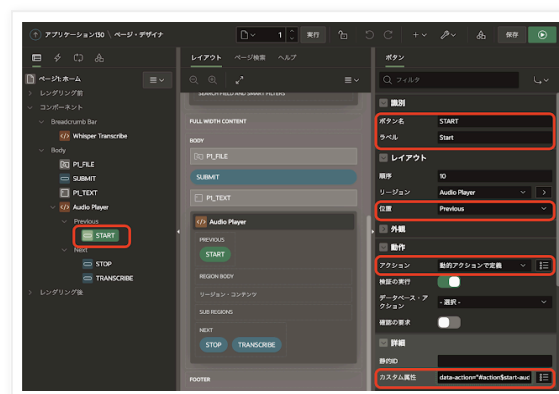


録音を開始するボタン**START**を作成します。

識別の**ボタン名**は**START**、**ラベル**は**Start**とします。**レイアウト**の位置として、レコーダーの左に配置されるよう、**Previous**を選択します。

動作の**アクション**に**動的アクション**で定義を選択し、**詳細のカスタム属性**にて実行する**APEXアクション**を指定します。

```
data-action="#action$start-audio-recording"
```

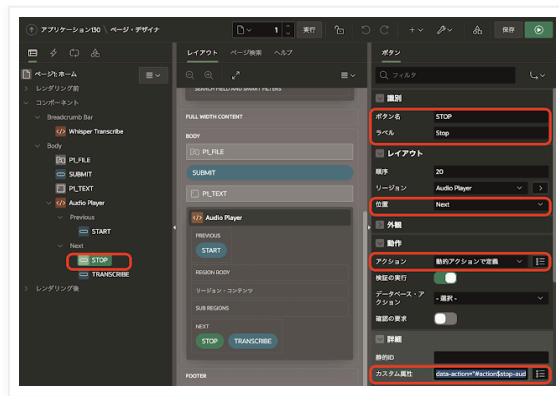


録音を停止するボタン**STOP**を作成します。

識別の**ボタン名**は**STOP**、**ラベル**は**Stop**とします。**レイアウト**の位置として、レコーダーの右に配置されるよう、**Next**を選択します。

動作の**アクション**に**動的アクション**で定義を選択し、**詳細のカスタム属性**にて実行する**APEXアクション**を指定します。

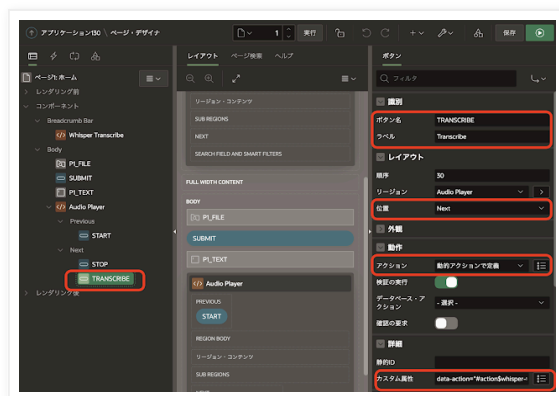
```
data-action="#action$stop-audio-recording"
```



同様に、Whisper APIを呼び出し文字起こしを実行するボタンTRANSCRIBEを作成します。

動作の指定である詳細のカスタム属性は、以下の記述になります。

`data-action="#action$whisper-transcribe?target=P1_TEXT&url=&G_TRANSCRIBE_URL."`



以上で実装は完了です。アプリケーションを実行すると、最初のGIF動画のように動作します。

例えば日報を音声入力するとしても、いちいちファイルに保存するのは現実的ではないと感じたので、JavaScriptで実装してみました。以前に紹介したアプリケーションのエクスポートは置き換えています。

Oracle APEXのアプリケーション作成の参考になれば幸いです。

完

Yuji N. 時刻: 17:05

共有

<

ホーム

>

ウェブ バージョンを表示

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。こちらの記事につきましては、免責事項の参照をお願いいたします。

[詳細プロフィールを表示](#)