* 制作物名
  + 「InfoMaid」

外見

* + キャラクターの表示
    - キャラクターを表示しない
    - LCDにMMDの3Dキャラクターを表示
    - LCDにキャラクターの画像表示(表情を変える)

簡易的な機能

* + 時報機能
    - 機能
      * 現在時刻を10分ごとに読み上げる
    - 必要項目
      * 1分毎のスレッド処理
      * 現在時刻の取得
      * 時刻を読み上げるWAVファイルを読み込んで再生する
  + 特定時刻機能
    - 機能
      * 特定の時刻になると様々な動作をする
      * 朝起きる時間になると曲を流すなど
      * クリスマスなどのイベントが近づくと教えてくれる等を予定
    - 必須項目
      * 現在時刻の取得
      * 時刻を読み上げるWAVファイルを読み込んで再生する
  + つぶやき機能
    - 機能
      * たまに何かぼそっとつぶやく
    - 必要項目
      * 1分毎のスレッド処理
      * 一定確率でつぶやきのWAVファイルを再生する
  + センサーコミュニケーション
    - 機能
      * 朝起きたときや出かける時などに、声をかけたり曲を流したりすることで安らぎをもたらす
    - 必要項目
      * 朝起きたとき
        + 人感センサとタイマーを用いた目覚め検知
        + 声掛けようのWAVファイル再生(おはよう等)
        + 目覚まし用の曲の再生
      * お出かけ時（距離センサをブルートゥース接続したものを玄関に置いておく案）、後で追加するかもしれない程度
        + センサとタイマーを用いたお出かけ検知
        + お出かけ用の声掛けWAVファイル再生(いってらっしゃい等)
      * 帰宅時
        + センサを用いた帰宅検知
        + 帰宅時用の声掛けWAVファイル再生(おかえり等)
        + たまに癒やされる曲流す
  + 天気情報の読み上げ
    - 機能
      * 出かける時に天気予報を読み上げる
    - 必要項目
      * 出かける時検知
      * ウェザーハックから得たJSON形式の天気情報をビューティフルスープでテキストデータにする(入手する天気の地域は金沢で固定)
      * OpenJTalkによるテキスト読み上げ
  + ニュースタイトルの読み上げ
    - 機能
      * 朝起きてから少しすると、ニュース情報を読み上げる
    - 必要項目
      * 朝起きる検知時に、一定時間後に動作するスレッドをセットする
      * ビューティフルスープを用いてyahooニュース総合カテゴリのタイトルをテキストデータとして取得する
      * Jtalkによるテキスト読み上げ
  + ツイッター読み上げ
    - 機能
      * ツイッターのリプライが飛んできた時に読み上げる
      * 不在時にリプライが飛んできた場合は読み上げないでよい
    - 必要項目
      * TwitterAPIを用いてリプライが飛んできたことを検知
      * TwitterAPIを用いてリプライの内容をテキストデータとして取得
      * Jtalkによるテキスト読み上げ

発展的な機能

* + 音声認識
    - 音声認識の基本的な実装
      * しゃべった言葉を録音
      * 音声データをテキストデータに変換
      * 変換されたテキストデータに対して返答するテキストデータを選択
      * テキストデータを合成音声で音声データに変換して出力
    - 音声認識で実現する機能群
      * BGM機能
        + 機能

音声操作で、あらかじめ入れてある曲の中から該当する曲を探して流す

音声操作で、ユウチューブでそれらしい曲を探して流す

動画再生中は、他のスレッド処理を停止する

* + - * + 必要項目

音声認識の基本的な実装

スクレイピングで再生したい動画URLを取得

Web「自作Python Youtube検索プログラム」を参照

動画URLから動画を取得して再生する

Omxplayerを用いる

動画再生用のアプリケーション

コマンドラインやパイソンコードで操作可能

Web「RaspBerryPi3でyoutubeを自動再生する」を参照

* + - * タイマー機能
        + 機能

音声操作でタイマーのセット、リセット、取り消し

明日何時に起きるかのタイマー

* + - * + 必要項目

音声認識の基本的な実装

タイマーのセット、リセット、取り消しの実装

* + - * 検索機能
        + 機能

ニュースタイトル読み上げの後に、音声操作でニュースタイトルの中から気になる記事の本文を読む

音声操作で単語の意味をウィキペディアから検索する。

* + - * + 必要項目

音声認識の基本的な実装

スクレイピングで検索したい記事のURLを取得

URLの記事(テキストデータ)を取得

テキストデータを読ませる

* 必要なハードウェア
  + 無線接続可能なラズパイ
  + ラズパイ周辺機器一式
  + マイクロホン
    - サンワサプライ USBマイクロホン MM-MCU02BK
      * 非常にコンパクトで場所をとらないものを選択
      * ラズパイでの動作◯
  + スピーカー
    - LOGICOOL ステレオスピーカー Z120BW
      * ラズパイでの動作◯
      * USB給電
  + 人感センサ
    - Seeed Studio　101020060　PIRモーションセンサ(大型レンズタイプ)
      * ラズパイでの動作◯
      * 配線が簡単、場所をとらない





* 音声認識システムを実現するための方法思索
  + Julis
    - 概要
      * 汎用大語彙連続音声認識エンジン(らしい)
      * オープンソースソフトウェア
      * 自分で辞書を作ることで、優先的に拾ってほしい単語を指示できる
    - 利用規約
      * 商業利用制限なし
    - 必要なもの(ネットから拾ってくるもの)
      * 音響モデル(HMM)
        + 音声波形パターンをモデル化したもの
      * 単語辞書
      * 言語モデル
        + 単語と読みの辞書
  + Docomo音声認識API
    - 概要
      * 音声データを送信すると音声認識したデータをテキスト化してJSON形式で返すAPI
      * API利用申請を行えば使用できる
      * 最大発話時間15秒
      * 利用制限は緩い
    - 利用規約
      * 個人利用は無償提供
      * 商用利用は個別相談
    - 必要なもの
      * ネットワーク環境
  + GoogleSpeechAPI
    - 概要
      * 音声データを送信すると音声認識したデータをテキスト化してJSON形式で返すAPI
        + テキストデータの信用度を付けてくれる

例：おはよう→「おはよう」、「オハヨウ」、「おはおう」

* + - * 実行時間が長い
        + 10秒の音声ファイルに対して実行時間は約10秒かかる
      * 主に10-15秒の音声に対応しており、短い音声データはうまく認識できない
      * 精度は高く、聞き取りづらい声でも拾ってくれる
    - 利用規約
      * 無料版では１日50回程度の実行に限られる
* ラズパイ上でボイスロイド(結月ゆかり2)を動かす方法の模索
  + 前提
    - ボイスロイドはlinux上では動かない
  + 以下、動作環境

■OS

Windows 10

Windows 8.1

Windows 7 SP1

※32bit/64bit対応

■CPU

Intel / AMD Dual Core以上のプロセッサ(Intel Core i3以上推奨)

→　2.2GHz

■必要なハードディスク空き容量

インストールに3GB以上の空き容量が必要

※システムドライブに1GBの空き容量が必要

■RAMメモリ

2GB以上(4GB以上推奨)

■ディスプレイ解像度

XGA(1024x768)以上の解像度

■その他

DVD-ROMドライブ

DirectX 9.0c以降に対応したサウンドカード

インターネット接続環境

※64bitOSでは32bitモードで動作します。

※日本語以外のOSは動作保証外です。

※Virtual PC、VMware等の仮想環境を除きます。

* + ではどうするのか？？？
    - ~~ラズパイ上にdockerでWindowsイメージをのせて動かす~~
      * そもそもdockerではCUIが使えるようになるがディスプレイ情報とかないから、docker上ではボイスロイド使えない。ボツ。
      * 動作は重くないのか？
        + ラズパイがメモリ1GBなのに対して、要求は最低2GBの推奨4GB。アウトでは？
        + ~~Windows10IOTなる、Iotを意識した軽量なOSがあるらしい~~

Windows10IOTはそもそもデスクトップが存在しないとかで、ボイスロイドの動作環境にはWin10Iotはないしまず動かないと考えていい。

* + - * Arm7では64\_84CPUのイメージは動かないからラズパイ対応のイメージを探すことになるが、数は少ないとのこと
      * マイクロSDでは書き込み回数の制限のため、外付けSSDにDockerを入れておくほうがいい
    - HTTPリクエストで動作するサーバーを立てておいて、テキスト渡したらWindowsのサーバー上でボイスロイドが動作してWAVファイルを返す
      * [参考]<https://narusejun.com/archives/18/>
      * 学生ならサーバー立てるのは無料らしい
      * 手間かかるよねこれ。ネットワークの知識が必要になるし。コマンドラインからボイロは動かせないから、Win32APIを使ったマウス操作でボイロを動かす筋肉ソリューションが必要になるし。
      * ラズパイからしてみれば即応性が無い。タイムラグなく対話するのは難しい。
        + じゃあそもそもWindowsで動かせば？？？
    - ~~ラズパイにメモリを増設して、仮想環境とかボイロとか全部ラズパイに乗っける。~~
      * そもそもメモリだけでなく、CPUの能力がが足りない。ボツ。
    - ラズパイをやめる
      * ラズパイ見たいに、USBで周辺機器さしこんですぐに使える奴をシングルボードコンピュータ(SBC)という
      * Arduinoみたいなのをワンボードマイコン(OBM)という
        + ボイスロイドの要求を満たすようなものだと、3万以上かかる
        + ワンボードマイコンでWindowsが初期で入らないものだと問題あり
      * NUC(ナック)とかの小型PC使う
        + セッティング難しそう
      * 用語調べ
        + DDR

読み込みと書き込みを同時にできるので処理効率は実質2倍速？

* + - * + EMMC

SSDよりも遅いが小さい