**ニチガス都市ガスメーターPJのAIツール基本設計書**

システム開発部　韓 廷旻

内容

[1. はじめに 2](#_Toc76640735)

[1.1. 背景 2](#_Toc76640736)

[2. 機能概要 4](#_Toc76640737)

[2.1. 機能 4](#_Toc76640738)

[3. プログラムの構成 5](#_Toc76640739)

[3.1. ネットワーク構成 5](#_Toc76640740)

[3.2. サーバー側 6](#_Toc76640741)

[3.2.1. AIの動作 6](#_Toc76640742)

[3.3. クライアント側 8](#_Toc76640743)

[3.3.1. クライアント側のプログラム動作原理 9](#_Toc76640744)

[3.3.2. プログラムのフォルダ構成 10](#_Toc76640745)

[3.3.3. tool\_config\_【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍.jsonについて 11](#_Toc76640746)

[4. その他 13](#_Toc76640747)

# はじめに

本書は「【ﾆﾁｶﾞｽ】都市ガスメーター交換＋蛍」プロジェクト」用の新しいAIツールに関して述べる

## 背景

現行のAIはガスメーターからガス使用量、シリアルナンバー、モデル名などを読み取っている。図 1に記す矩形は、現行のAIで読み取ることができる領域である。各項目の内容を表 1.1に記す。

텍스트, 측정기, 실외, 주차장이(가) 표시된 사진

자동 생성된 설명

図 1 現行のAI

表 1 現行のAIで読み取れる項目一覧

|  |  |  |
| --- | --- | --- |
| # | 現行のAIで読み取れる項目 | 説明 |
|  | SK25MT1S | 機器の型。機器の形を見て判断する。 |
|  | type | 機器の型。機器に張られているシールから判断する |
|  | GasUsage | ガス使用量 |
|  | SerialNumber | 機器のシリアルナンバー |

しかし、「【ﾆﾁｶﾞｽ】都市ガスメーター交換＋蛍」プロジェクト」はAIで以下の内容を読み取る必要がある。

* 「図 2 取付前メーター写真」画像項目からは取外メーター指針値、取外メーター番号
* 「図 3 取付後メーターと蛍写真」画像項目からは取付メーター指針値、取付メーター番号
* 「図 4 完了のお知らせ」画像項目からは取外メーター指針値、取外メーター番号、取付メーター指針値、取付メーター番号



取外メーター番号

取外メーター指針値

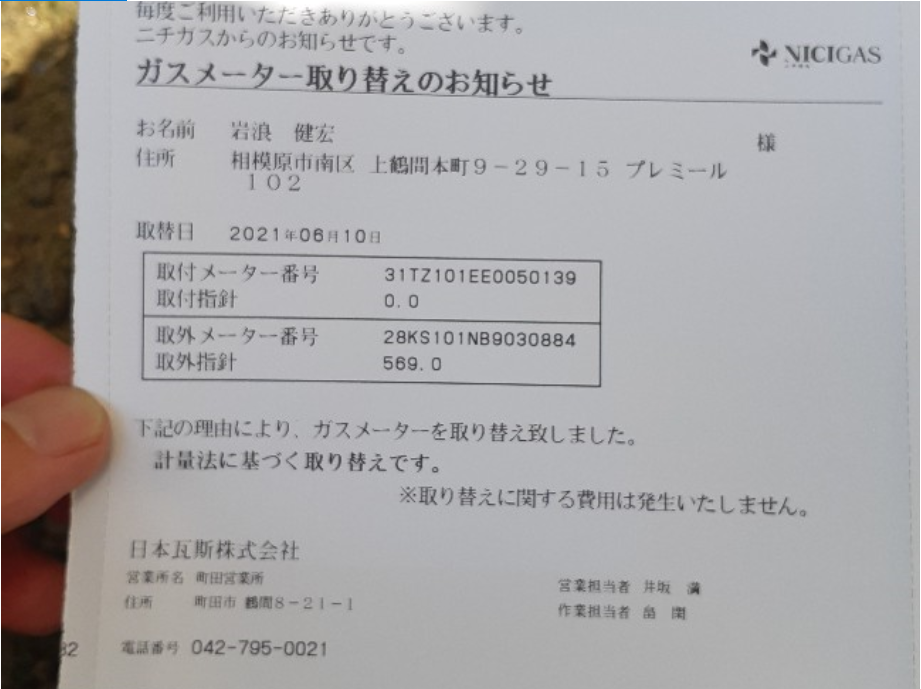
図 2 取付前メーター写真



取付メーター番号

取付メーター指針値

図 3 取付後メーターと蛍写真



取外メーター指針値

取外メーター番号

取付メーター指針値

取付メーター番号

図 4 完了のお知らせ

現在IOT推進部に使ってもらっている現行のAIでは上記の内容を読み取ることができないため、本開発で対応する。

# 用語の定義

本章で使用する用語の定義を記す。

表 2 用語の定義

|  |  |  |
| --- | --- | --- |
| # | 用語 | 説明 |
| 1 | ニチガス都市PJ | BLASの【ﾆﾁｶﾞｽ】都市ガスメーター交換＋蛍」プロジェクトの略記。 |

# 機能概要

本章はプログラムの機能に関して述べる。

## 機能

IoT推進部さんからの依頼機能は8つがある。

1. **取付前ガスメーターの指針値**

「図 2 取付前メーター写真」画像項目の画像から読み取った**取外メーター指針値**をAIで読み取って結果をBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付後ガスメーターの指針値**

「図 3 取付後メーターと蛍写真」画像項目の画像から読み取った**取付メーター指針値**をAIで読み取って結果をBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付前ガスメーター番号と作業者入力比較**

「図 2 取付前メーター写真」画像項目の画像から読み取った**取外メーター番号**と「【作業員入力】取外メーター社番QR読取り※リストと同じか注意」項目の作業者入力値と比較し、等しい場合はOK、異なる場合はNGをBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付後ガスメーター番号と作業者入力比較**

「図 3 取付後メーターと蛍写真」画像項目の画像から読み取った**取付メーター番号**と「【作業員入力】取付メーター社番QR読取り」項目の作業者入力値と比較し、等しい場合はOK、異なる場合はNGをBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付前ガスメーター指針値と完了のお知らせ比較**

「図 2 取付前メーター写真」画像項目の画像から読み取った**取外メーター指針値**と「図 4 完了のお知らせ」画像項目の画像から読み取った**取外メーター指針値**と比較し、等しい場合はOK、異なる場合はNGをBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付前ガスメーター番号と完了のお知らせ比較**

「図 2 取付前メーター写真」画像項目の画像から読み取った**取外メーター番号**と「図 4 完了のお知らせ」画像項目の画像から読み取った**取外メーター番号**と比較し、等しい場合はOK、異なる場合はNGをBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付後ガスメーター指針値と完了のお知らせ比較**

「図 3 取付後メーターと蛍写真」画像項目の画像から読み取った**取付メーター指針値**と「図 4 完了のお知らせ」画像項目の画像から読み取った**取付メーター指針値**と比較し、等しい場合はOK、異なる場合はNGをBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

1. **取付後ガスメーター番号と完了のお知らせ比較**

「図 3 取付後メーターと蛍写真」画像項目の画像から読み取った**取付メーター番号**と「図 4 完了のお知らせ」画像項目の画像から読み取った**取付メーター番号**と比較し、等しい場合はOK、異なる場合はNGをBLASの特定の項目に書き込む。書き込む項目は設定ファイルで指定できる。

# プログラムの構成

本章はプログラムの構成に関して述べる。

## ネットワーク構成

仕様は現行のガスメーターAIと同様である。（参照：図 5 ネットワーク構成）

AIサーバー：192.168.10.103：500XのDocker環境（500Xはポート番号）  
 ポート5000：ニチガス都市PJ用AIサーバー  
 ポート5003：既存のAIサーバー

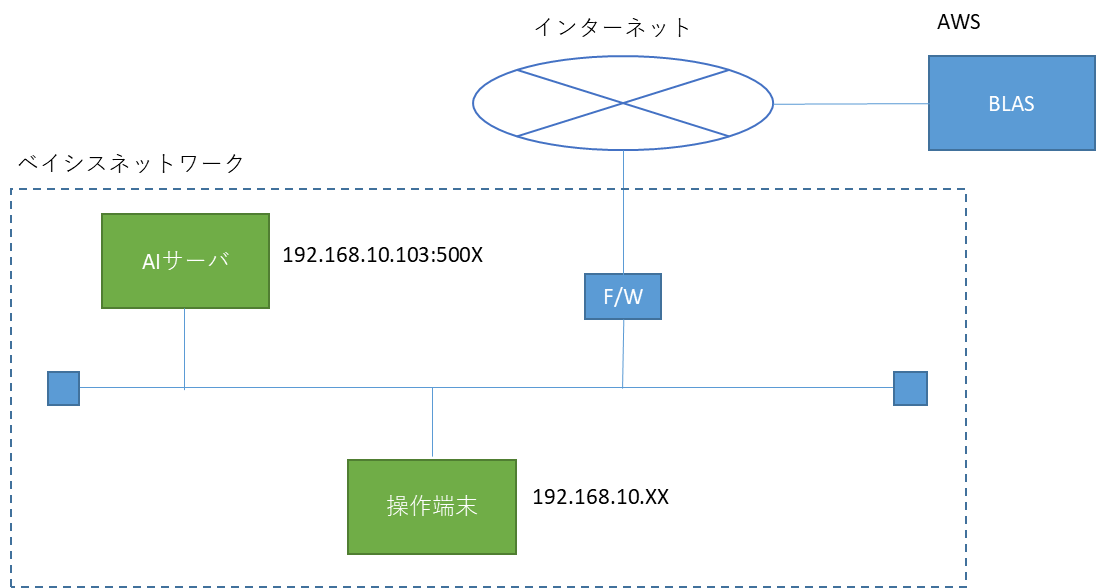


図 5 ネットワーク構成

## サーバー側

現行のガスメーターAIに学習データ「図 2 完了のお知らせ」追加し、学習すると現行のAIの精度に影響を与える可能性があるため、「【ﾆﾁｶﾞｽ】都市ガスメーター交換＋蛍」プロジェクト」専用のAIを新しく作成する。

### AIの動作

ニチガス都市PJ用のAIは３段階で値を読み取る。

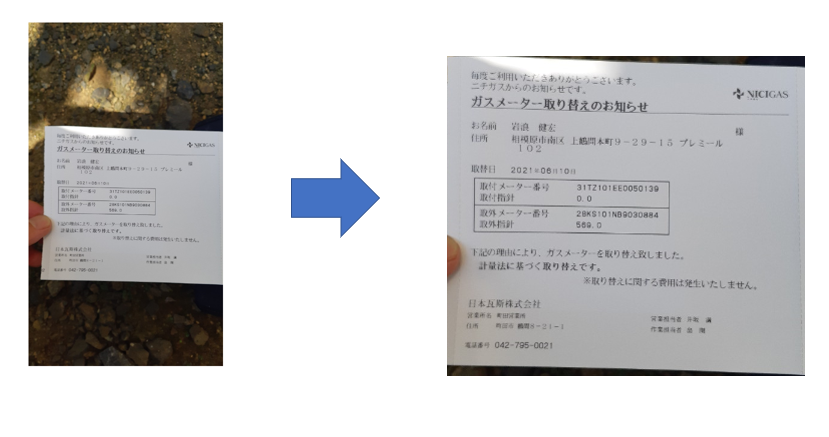
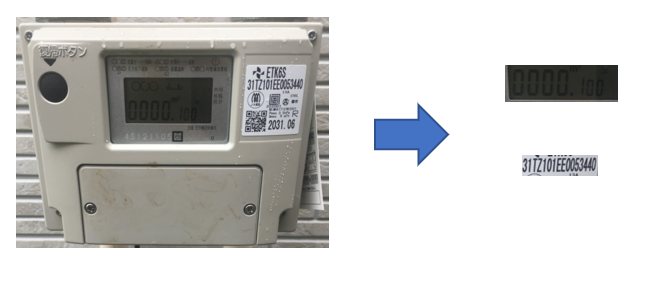
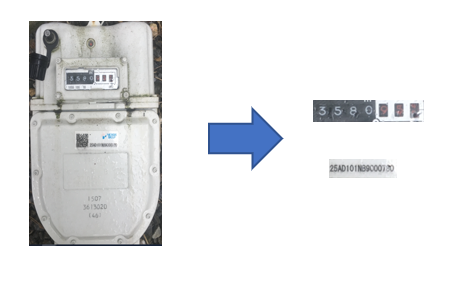
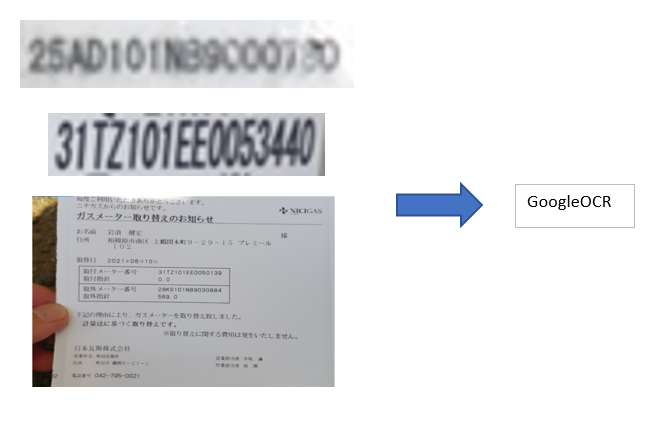
1. 領域検出AIを用いて読み取りたいテキスト領域だけを切り出す。（下の図参照）  
   

図 6 読み取りたいテキスト領域の切り出し

1. 切り出したテキスト領域をテキスト予測AI又はGoogle OCRで読み取る。  
     
   ・「図 2 取付前メーター写真」画像と「図 3 取付後メーターと蛍写真」画像の指針値はテキスト予測AIを使う。（下の図参照）  
     
   ・「図 2 取付前メーター写真」画像と「図 3 取付後メーターと蛍写真」画像のメーター番号と「完了のお知らせ」はGoogleOCRを使う（下の図参照）  
   
2. テキスト予測AI又はGoogleOCRで読み取った値を後処理で補正する。例えば、前4桁が正数、後ろ3桁が小数という規則があるにもかかわらず、小数点を読めなかった場合は「.」を追加する、メーター番号の場合は英数字だけにもかかわらず全角の文字が含まれていた場合は英数字に変換するなどである。  
   指針値の場合）0000100 → 0000.100  
   メーター番号の場合）英数字だけ

## AIサーバーのリクエストとレスポンス

本節では、AIサーバのリクエストとレスポンスについて説明する。

### リクエスト

**[リクエストURL]**

<http://192.168.10.103:5000/>

**[リクエストパラメーター]**

**メソッド**：POST

|  |  |  |
| --- | --- | --- |
| # | パラメーター | 説明 |
| 1. | file | 画像ファイルのバイナリデータ |

### レスポンス

レスポンスはjson形式で返却する。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | フィールド１ | ２ | ３ | 説明 |
|  | gasmeter | before | GasUsage | 取付前ガスメーターの指針値 |
|  | SerialNumber | 取付前ガスメーターのメーター番号 |
|  | after | GasUsage | 取付後ガスメーターの指針値 |
|  | SerialNumber | 取付後ガスメーターのメーター番号 |
|  | confirmation |  |  | 完了のお知らせ |

## クライアント側

この章では実際ユーザが設定及び起動するプログラムについて述べる。

### クライアント側のプログラム動作原理

クライアント側のプログラムは大きく4段階の処理を行う。(図 7 プログラムのシーケンス図 参照)

1. クライアント側のプログラムからBLASのRestAPIを用いて該当レコードの情報及び画像を取得する。
2. 取得した画像をrequestでAIサーバーに送信し、responseでAIサーバーが予測結果を取得する。
3. 「2.1機能」で定義された8つの値を求めるため、適切な処理を行う。
4. 求めた8つの値をBLASのRESTAPIを用いてBLASに書き込む。

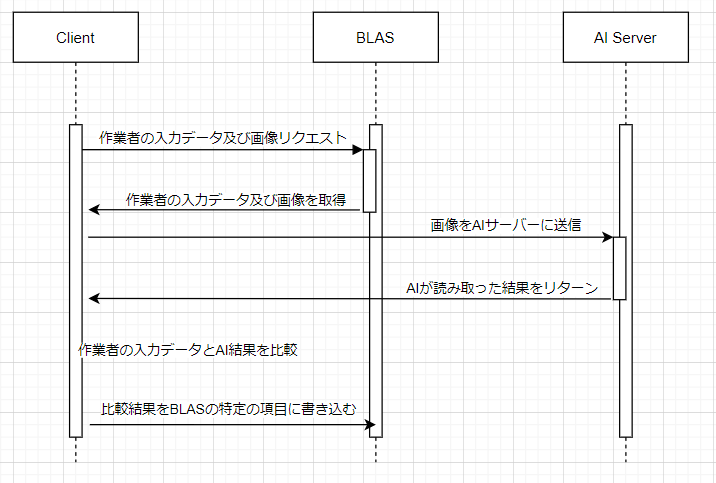


図 7 プログラムのシーケンス図

### プログラムのフォルダ構成

|  |
| --- |
| **nichigas\_gasmeter\_client** |
| **│ .gitignore** |
| **│ gas\_check\_start\_ニチガス.bat** |
| **│ README.md** |
| ├─ **gas** |
| │ │ ai\_client.py |
| │ │ app.py |
| │ │ blas\_client.py |
| │ ├─ blas\_rest |
| │ ├─ python-3.6.8-embed-amd64 |
| ├─ **setting\_history** |
| │ tool\_config\_【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍.json |
| └─**説明書** |

各ファイルやフォルダの説明は下記の通りである。

* **.gitignore ファイル**  
  githubにpushする時、push対象外のファイルを設定するためのファイル
* **gas\_check\_start\_ニチガス.bat ファイル**  
  ニチガス都市PJ用のAIツールを起動するためのバッチファイル
* **README.md ファイル**  
  プログラムに関しての簡単に説明するファイル（github用）
* **gas フォルダ**

プログラムの本体が格納されるフォルダ

* **gas/ai\_client.py**  
  AIサーバーと通信するためのモジュール
* **gas/app.py**プログラムの本体
* **gas/blas\_client.py**BLASと通信するためのモジュール
* **gas/blas\_rest フォルダ**BLASと通信するためのRestfulモジュール。gas/blas\_client.pyから呼び出す
* **gas/python-3.6.8-embed-amd64 フォルダ**組み込みpython。pythonコマンドやライブラリは本ディレクトリに格納する
* **setting\_history/ tool\_config\_【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍.json**プログラム起動時のニチガスPJ用のjson形式の設定ファイル
* **説明書 フォルダ**プログラムに関しての書類

### tool\_config\_【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍.jsonについて

プログラムを起動するため、tool\_config\_【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍.jsonに適切に設定する必要がある。

{

    "blas\_user\_name": "",

    "blas\_password": "",

    "blas\_project": "【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍",

    "blas\_url\_debug": "http://localhost/blas7/api/v1/",

    "blas\_url": "https://www.basis-service.com/blas70/api/v1/",

    "ai\_url\_debug": "http://192.168.10.103:5000",

    "ai\_url": "http://192.168.10.103:5000",

    "debug": false,

    "images": [

        "取付前ﾒｰﾀｰ写真",

        "取付後ﾒｰﾀｰと蛍写真",

        "完了のお知らせ"

    ],

    "compare\_images": [

        {

            "output": "（AI）取外メーター指針値と完了のお知らせチェック"

        },

        {

            "output": "（AI）取外メーター番号と完了のお知らせチェック"

        },

        {

            "output": "（AI）取付メーター指針値と完了のお知らせチェック"

        },

        {

            "output": "（AI）取付メーター番号と完了のお知らせチェック"

        }

    ],

    "output\_ai\_fields": {

        "before\_gasusage": {

            "result\_field": "（AI）取外メーター指針値"

        },

        "after\_gasusage": {

            "result\_field": "（AI）取付メーター指針値"

        }

    },

    "check\_field": {

        "column": "（AI）取外メーター番号と作業者入力チェック",

        "skip\_word": [

            "OK"

        ]

    },

    "compare\_blas\_fields": {

        "【作業員入力】取外メーター社番QR読取り※リストと同じか注意": {

            "result\_field": "（AI）取外メーター番号と作業者入力チェック"

        },

        "【作業員入力】取付メーター社番QR読取り": {

            "result\_field": "（AI）取付メーター番号と作業者入力チェック"

        }

    },

    "conditions": {},

    "date\_search": [{

        "field": "【作業員入力】作業日",

        "from": "2021-07-05"

    }]

}

各項目に関しては以下の通りである。

* blas\_user\_name：BLASのユーザID
* blas\_password：BLASのパスワード
* blas\_project：プロジェクト名  
  （デフォルト：【ﾆﾁｶﾞｽ】ﾆﾁｶﾞｽ\_都市ガスﾒｰﾀｰ交換＋蛍）
* blas\_url\_debug：デバッグ用のBLAS RestAPI URL  
  （デフォルト：http://localhost/blas7/api/v1/）
* blas\_url：本番BLAS用のBLAS RestAPI URL  
  （デフォルト：https://www.basis-service.com/blas70/api/v1/）
* ai\_url\_debug：デバッグ用のAIサーバー URL

（デフォルト：http://192.168.10.103:5000）

* ai\_url：本番BLAS用のAIサーバー URL  
  （デフォルト：http://192.168.10.103:5000）
* debug：デバッグモード設定  
  （デフォルト：False）
* images：AIに読ませたい画像項目を設定  
  （デフォルト：取付前ﾒｰﾀｰ写真、取付後ﾒｰﾀｰと蛍写真、完了のお知らせ）
* compare\_images：取付前後ガスメーター画像と完了のお知らせ画像を比較（2.1機能 5~8）。比較結果はoutputに設定したBLAS項目に書き込む

（デフォルト：（AI）取外メーター指針値と完了のお知らせチェック, （AI）取外メーター番号と完了のお知らせチェック, （AI）取付メーター指針値と完了のお知らせチェック, （AI）取付メーター番号と完了のお知らせチェック）

* output\_ai\_fields：取付前後ガスメーター画像の指針値をresult\_fieldに設定したBLAS項目に書き込む（2.1機能 1~2）

（デフォルト：（AI）取外メーター指針値、（AI）取付メーター指針値）

* check\_field：既にAIがチェックしたレコードをスキップするための設定項目。columnに設定したBLAS項目にskip\_workが既に入っているとスキップする。  
  （デフォルト： column: （AI）取外メーター番号と作業者入力チェック、 skip\_word: OK）
* compare\_blas\_fields：現場作業者の入力値と完了のお知らせ画像の値を比較（2.1機能 3~4）比較結果はresult\_fieldに設定したBLAS項目に書き込む。  
  （デフォルト：（AI）取外メーター番号と作業者入力チェック、（AI）取付メーター番号と作業者入力チェック）
* conditions：独立検索のための設定。検索に該当するレコードだけ
* date\_search：日付検索のための設定。日付検索に該当するレコードだけを

**注意事項**

ユーザはblas\_user\_name, blas\_password, check\_field, conditions, date\_search項目を設定する必要がある。上記以外の項目はデフォルトのまま使用してよい。

# 付録

## Dockerファイル

AIサーバーのDockerファイルを以下に記す。

|  |
| --- |
| FROM nvidia/cuda:9.2-devel-ubuntu18.04  RUN apt update  RUN apt-get install -yq wget build-essential gcc zlib1g-dev  WORKDIR /root/  RUN apt-get -y install software-properties-common  RUN apt-get update  RUN add-apt-repository ppa:deadsnakes/ppa  RUN apt-get update  RUN apt-get -y install python3 python3-pip  RUN wget https://bootstrap.pypa.io/get-pip.py  RUN python3.6 get-pip.py  RUN apt-get install -y libsm6 libxext6 libxrender-dev  RUN apt-get install -y libgl1-mesa-dev  RUN apt-get install -y libzbar0  RUN pip3 install torch==1.6.0+cu92 torchvision==0.7.0+cu92 -f https://download.pytorch.org/whl/torch\_stable.html  RUN pip3 install flask uwsgi opencv-python requests tqdm matplotlib  RUN pip3 install pillow==6.2.1  RUN pip3 install qrcode==6.1  RUN pip3 install pyzbar==0.1.8  #ポート5000を開けて5000にポートフォワードする  EXPOSE 5000:5000  ADD ./ai/ /root/ai  ADD ./client/ /root/client  ADD ./utils/ /root/utils  ADD ./yolo\_cfg/ /root/yolo\_cfg  ADD ./weights/ /root/weights  ADD ./config.py /root  ADD ./main.py /root  ADD ./models.py /root  RUN apt-get update \  && apt-get install -y locales \  && locale-gen ja\_JP.UTF-8 \  && echo "export LANG=ja\_JP.UTF-8" >> ~/.bashrc  CMD ["/usr/local/bin/uwsgi", "--socket", "0.0.0.0:5000", "--protocol=http", "-w", "main:app"] |