

RTSystemBuilder マニュアル

Ver.0.1.0

2023/XX/XX



株式会社 グローバルアシスト

目 次

1	概要	3
1.1	ツール概要.....	3
2	インストールと設定	4
2.1	サーバ側の設定	4
2.1.1	データベースの準備	4
2.1.2	Python 実行用仮想環境の構築.....	5
2.1.3	サーバ実行用バッチファイルの修正	5
2.2	クライアント側の設定	6

1 概要

1.1 ツール概要

XXX は、

2 インストールと設定

2.1 サーバ側の設定

2.1.1 データベースの準備

1.1.1.1 データベースのインストール

リレーショナルデータベース管理システム(RDBMS)として、PostgreSQL を使用します。インストーラは、以下のサイトなどからダウンロードする事が可能です。

<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>

※こちらの環境では、Windows 版 Ver.13.5 で動作確認を行っております。

インストーラを使用してインストールを行う場合には、基本的には全てデフォルト設定のままでインストールしてください。なお、途中で管理者パスワードを設定する部分がありますが、こちらは任意のパスワードを設定して頂いて構いません。

インストールを行いましたら、コマンドプロンプトなどで、

`<PostgreSQL インストール先ディレクトリ>/bin/psql.exe -version`

を実行し、バージョン番号が表示される事を確認してください。

1.1.1.2 データベースの構築

「スタート」メニューから表示されるプログラム一覧から、「PostgreSQL X.X」→「SQL Shell(psql)」を選択して起動してください。「Server」や「Database」など、幾つかの情報を入力されるように求められますが、全てデフォルト値のままで構いません。ただし、パスワードにつきましては、インストールした際に設定したパスワードを入力してください。

起動した SQL Shell で以下の SQL コマンドを順番に実行します。

- ・ユーザ(ロール)の作成

```
CREATE ROLE rsdlab LOGIN PASSWORD 'rsdlab' ;
```

- ・データベースの作成

```
CREATE DATABASE componentdb;
```

- ・権限の付与

```
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO rsdlab;
```

一度、SQL Shell を終了した後、再度、SQL Shell を起動してください。ただし、今回は以下を指定してください。

Database : componentdb

ユーザ名 : rsdlab

SQL Shell 内に Server/doc/TranTables.sql の中身を貼り付けて各テーブルを作成してください。なお、テーブルを作成する際には、一度に全てのテーブル作成を行うのではなく、1つずつ作成し、エラーが発生していないか確認してください。

2.1.2 Python 実行用仮想環境の構築

まず Python 本体をインストールしてください。

※こちらの環境では、Windows 版 Ver.3.6.8 を使用して動作確認を行っております。

サーバ実行用 Python 仮想環境を保持するディレクトリを作成し、コマンドプロンプトなどで作成したディレクトリに移動してください。

以下のコマンドを実行し、仮想環境を作成してください。

```
<Python インストールパス>/python -m venv <仮想環境名>
```

※<仮想環境名>には任意の名称を付けてください。

コマンドプロンプトで、以下のコマンドを実行し、作成した仮想環境に入ってください。

```
<仮想環境構築ディレクトリ>/Scripts/activate.bat
```

作成した仮想環境ディレクトリに、Server/doc/requirement.txt をコピーした後、以下のコマンドで仮想環境の構築を行ってください。

```
pip install -r requirements.txt
```

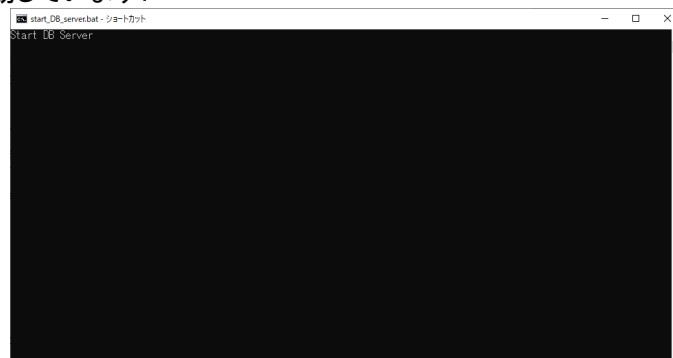
2.1.3 サーバ実行用バッチファイルの修正

DB サーバ起動用バッチファイル(Server/ComponentDatabaseService/start_DB_server.bat)の中身を構築した環境に応じて修正してください。修正する箇所は、「APP_ROOT」と「APP_Python」の2箇所となります。

```
set APP_ROOT=" <DB サーバのソースを配置したディレクトリ名> "
```

```
set APP_Python=<Python 仮想環境を構築したディレクトリ名>
```

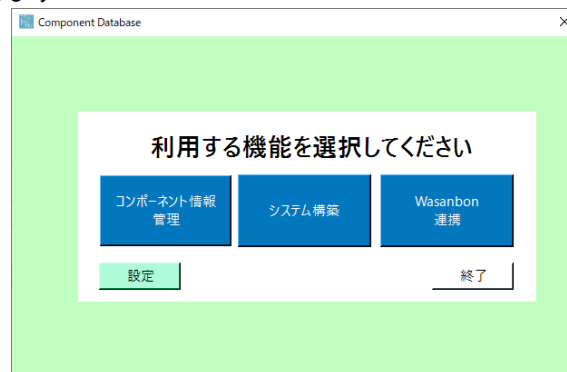
上記を修正した後、DB サーバ起動用バッチファイルを実行してください。以下のような画面が表示されれば DB サーバが正常に起動しています。



サーバ起動用バッチファイルは任意のディレクトリから実行可能です。

2.2 クライアント側の設定

クライアントアプリを起動します。



画面左下の「設定」ボタンを選択し、【設定画面】を表示します。

各項目の設定内容は以下の通りです。

◎データベース設定

・IP アドレス

サーバ側ソフトを実行している PC の IP アドレスを設定します。クライアントアプリと同じ PC 上で起動している場合には、「localhost」と設定しても構いません。

・ポート番号

データベース管理システム(PostgreSQL)が使用しているポート番号を設定します。PostgreSQL のインストールを行う際に、ポート番号を変更していなければ、デフォルト設定の「50201」を設定します。

◎GitHub 設定

・ユーザ Id

クライアントアプリを使用するユーザの GitHub アカウント名を設定します。

・アクセストークン

「ユーザ Id」で設定した GitHub アカウントで取得したアクセストークンを設定します。

・システム構築時ベースリポジトリ URI

構築したシステムを登録するユーザ/組織のリポジトリの URI を設定します。

→大原研のリポジトリを使用する場合には、<https://github.com/rsdlab> を指定します。

・Wasanbon binder レポジトリ URI

使用する Wasanbon binder のリポジトリの URI を設定します。

→大原研のリポジトリを使用する場合には、https://github.com/rsdlab/wasanbon_binder を指定します。