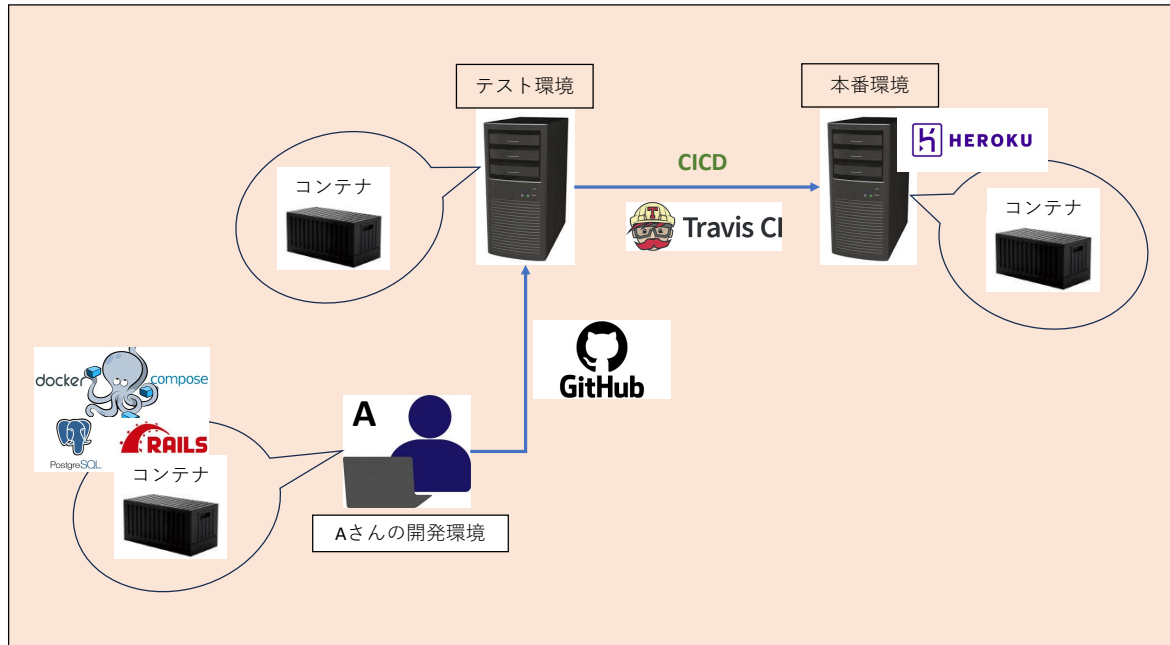


## 【自己学習-Docker環境構築】



### 【構成内容】

Dockerを利用しての「開発」/「テスト」/「本番」環境の統一化を図りました。

全体の流れとしては、Aさんの開発環境下にある各コンテナ(PostgreSQL・RAILS)をDockerComposeにて管理し、成果物をGitHub上のテスト環境へとpushします。そして連携させたTravis CIにてテストを実施し合格した内容が本番環境へと反映されます。

```
Dockerfile
~/Desktop/product-register

1 FROM ruby:2.7.8
2 RUN apt-get update && apt-get install -y \
3     build-essential \
4     libpq-dev \
5     nodejs \
6     postgresql-client \
7     yarn
8 WORKDIR /product-register
9 COPY Gemfile Gemfile.lock /product-register/
10 RUN bundle install
```

```
docker-compose.yml
~/Desktop/product-register

1 version: '3'
2
3 volumes:
4   db-data:
5
6 services:
7   web:
8     build: .
9     ports:
10      - '3000:3000'
11     volumes:
12      - './product-register'
13     environment:
14       DATABASE_PASSWORD: 'postgres'
15     tty: true
16     stdin_open: true
17     depends_on:
18       - db
19     links:
20       - db
21
22   db:
23     image: postgres
24     volumes:
25       - 'db-data:/var/lib/postgresql/data'
26     environment:
27       - 'POSTGRES_USER=postgres'
28       - 'POSTGRES_PASSWORD=postgres'
```