Stuart Challenge

Project Structure

This challenge is currently structured with the following specifications.

Path	Description
src	Contains all application regarding the pipeline
landing	Directory to hold coordinates **gzip** file
raw	Directory to uncompress coordinates file
tests	App unit tests

Filesystem

Inside the container you will be able to find the following paths.

Path	Description
/stuart	Root path to pipeline application
/stuart/challenge	Directory with application
/stuart/raw	Used as path to uncompressed files
/stuart/landing	Used as path to compressed files

Unit Tests

```
A few unit tests were made as a demonstration in some classes of the
project.

1. test_geohash_from_position : Function to get Geohash code from latitude
and logitude
2. test_position_from_geohash : Function to latitude and logitude from
Geohash code
3. test_get_geohash_prefixes : Function to generate all possible prefixes
to Geohash code.
4. test_get_shortest_prefix : Function to get the shortest prefix to
Geohash code.
```

All testes are being executed during Docker Build

```
RUN python3 -m unittest discover -s tests
```

if necessary is possible to run it inside the container with the command below

```
cd /stuart
$ python3 -m unittest discover -s tests
```

Build Image:

It's possible run the challenge on the fly with docker-compose or build the image and interact with the container.

Deploying Stuart challenge

```
$ git clone https://github.com/StuartHiring/python-test-Andre-Junior.git
$ cd python-test-Andre-Junior
```

Docker compose

```
$ docker-compose up -d
```

Docker Image

```
$ docker build -t stuart/challenge:1.0 .
$ docker run -d -t stuart/challenge:1.0
```

Run pipeline Manually:

During the Docker build process the pipeline is executed for the first time but if necessary is possible rerun using the approach below: Note: Don't foget to delete the previous file on path /stuart/raw

```
$ cd /stuart/challenge
$ python main.py
```