Development guide for news sentiment BI

1. Environment requirement
2. Sample data initialize
3. ETL
4. Client
5. News (Reporting server for Client UI)
6. ORM and database structure
7. Endpoints
8. Unit Test
9. Client
10. Order + queueing system
11. Environment requirement

|  |  |
| --- | --- |
| IDE | Visual Studio Code or other IDE for code editing |
| Application Server | Build in Flask + URL mapping in python3 |
| Python / PIP3 version | >= 3.7.4 |
| NPM version | >= 12.18.0 |
| Docker | Make sure docker is installed on the development computer |
| AWS | AWS account and the policy right are set (You can use files in /Setup/03\_AWS\_IAM\_POLICY.json) |
| Git | Git command or Git application |
| Testing database | MySQL server 5.7.30 or above |
| Source | <https://github>.com/brianlaihkhk/sample-news-sentiment |

1. Sample data initialize

Inside the /Setup folder, you can initialize the DB

|  |  |
| --- | --- |
| 00\_DATABASE\_INIT.sql | Initialize the database, create user and set according permissions |
| 01\_CREATE\_TABLE.sql | Create table / schema script (Here we don’t use hibernate.hbm2ddl.auto to generate) |

If you want to use AWS solution, upload the Sample data to S3, obtain the wheel files (.whl) using

* python3 -m pip download --only-binary :all: --dest . -r requirements.txt

Upload the wheel files to S3 folder. The files are required in AWS Glue calling python script.

1. ETL

(a) ETL using local folder:

Obtain the module using

* Python3 install -r requirements.txt

You can start importing data below command for data import

* python3 ./local\_server.py –env-file=<environment file>

(b) ETL using AWS S3:

In the .env.XXXX file, update the corresponding API keys and folders for the process job

Text

Description automatically generated

Move the files into corresponding S3 folders

Graphical user interface, application

Description automatically generated

Execute python3 ./s3\_start, it will load the files from the s3 buckets

Graphical user interface, text, email

Description automatically generated

1. Client

First obtain the package using npm install . It will load the package.json

Inside /Client/App.js, you can config the HOST in line 7 for the backend url (Currently is pointing to localhost 8000)

Text

Description automatically generated with medium confidence

You can run local development via npm run start

1. News (Reporting server for Client UI)

Obtain the module using

* Python3 install -r requirements.txt

You can start running below command for server start

* python3 ./server.py –env-file=<environment file>

You can change the listening port and host in .env.XXX

1. ORM and database structure

Files are defined in /News/orm.py

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Referenced Table | Association | Remarks |
| AggregateCategory | AGGREGATE\_CATEGORY | PK: NEWS\_DAY, CATEGORY |  |
| AggregateSentiment | AGGREGATE\_SENTIMENT | PK: NEWS\_DAY, SENTIMENT |  |
| AggregateTopic | AGGREGATE\_TOPIC | PK: NEWS\_DAY, TOPIC |  |
| AggregateTag | AGGREGATE\_TAG | PK: NEWS\_DAY, TAG |  |
| News | NEWS | PK: NEWS\_UUID | News article table |
| NewsMap | NEWS | PK: NEWS\_MAP\_UUID | Mapping table for news metadata to news |

1. Endpoints

Please refer to WS\_XXXXX.yml for more information

1. Unit Test
2. Client:

Run native-script + jest by using

* Npm test

The test file is located the same folder of the source code (inside /Client/src/components/XXXX.test.js)

b. Order + queueing system:

It is located inside /Unit Test folder

|  |  |
| --- | --- |
| Test\_etl.py | Unit test for ETL script  To run , execute python3 ./test\_etl.py |
| Test\_news.py | Unit test for News (Reporting server)  To run , execute python3 ./test\_news.py |