Web-Based-Tug-Boat-Scheduling-System

Introduction

This is a demo of tugboat scheduling system. It is a web-based automatical scheduling system.

This Demo only can run on LAN.

Requirement

This sytem is based on python enviornment.

The environments and dependencies' version of demo overview:

```
OS: Windows
Python == 3.12.2
Django == 5.0.2
pandas == 2.2.1
Requests == 2.31.0
node.js == 20.11.1
npm == 10.2.4
pytz == 2024.1
Vue == 3.4.21
```

System Environment

```
Windows
```

Python Environment

```
Python version >=3.10
```

If have ever installed Python, use the command following in command prompt to check the version.

```
>python

Python 3.11.4 (tags/v3.11.4:d2340ef, Jun  7 2023, 05:45:37) [MSC v.1934 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

If have not ever installed Python, please go to Python official website to download and install.

Back-End: Django

Make sure the Python has been installed successfully before install the back-end.

- Install the **Django**:

Run the command in command prompt:

```
python -m pip install Django
```

The last version of Django is `v5.0.2`.

- Install **pandas**:

Run the command in command prompt:

```
pip install pandas
```

Pandas is an open-source, BSD licensed library that provides high-performance, easy-to-use data structures and analysis tools

- Install **requests**:

Run the command in command prompt:

```
pip install requests
```

Requests is a commonly used HTTP request library that facilitates sending HTTP requests to websites and obtaining response results.

- Install **pytz**:

Run the command in command prompt:

```
pip install pytz
```

Pytz is a library for time and date.

Front-End: Vue.js and Node.js

- Install node.js

The vue is depending on node.js. If have ever installed `node.js` and `npm`, please check the version:

Run the command in command prompt:

```
node -v
npm -v
```

If have not ever installed, please go to node.js official website to download and install.

- Install Vue.js

Run the command in command prompt:

```
npm install vue -g
```

Check the version of `vue.js`:

Run the command in command prompt:

```
npm list vue
```

- Install `vue-cli` Open the folder `website\NingboHarbour` and run command in command prompt:

```
npm install
```

Start Server

At Back-End

- Run server

Open folder `web\WBTBSsystem` and run command in command prompt:

```
python manage.py runserver
```

Default run at http://127.0.0.1:8000/.

Successful running example:

```
System check identified no issues (0 silenced).
March 26, 2024 - 22:54:55
Django version 4.2.7, using settings 'WBTBSsystem.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

- Access Database through super-user

Admin Super-user:

```
Username: ADMIN001
Password: 12345678

Username: Team202310
Password: Team202310
```

Access database through http://127.0.0.1:8000/admin

NO # BETWEEN 8000 and ADMIN Otherwise, **CAN NOT** go into back-end admin page.

- Run on LAN

If want to run this system on a LAN, please open folder `website\NingboHarbour` and run command in command prompt:

```
python manage.py runserver 0.0.0.0:8000
```

then check the IPv4 of the server device by following command in command prompt:

```
ipconfig
```

Then find `IPv4 Address`:

For example:

```
IPv4 Address . . . . . . . . . . . . : 172.19.39.208
```

Then other devices should access the website through the Ipv4:

For example, based on the IPv4 above, other devices should visit:

```
172.19.39.208:8000
```

At Front-End (Optional)

- Run server

Open folder `website\NingboHarbour` and run command in command prompt:

```
  npm run serve
```

Successful running example:

```
 DONE  Compiled successfully in 175ms
23:00:55


  App running at:
  - Local:   http://localhost:8080/
  - Network: http://172.19.39.208:8080/
```

Default run at http://127.0.0.1:8080/ and network url is based on device's IPv4.

Default Accounts

Access serve as **ADMIN**:

```
  Username: ADMIN001
  Password: 12345678
```

Access serve as **SCHEDULER**:

```
  Username: SC0001
  Password: 12345678

  Username: SC0002
  Password: 12345678
```

Access serve as **CAPTAIN** (CP0001-CP0040):

```
  Username: CP0001
  Password: 12345678

  Username: CP0002
  Password: 12345678


  ...


  Username: CP0040
  Password: 12345678
```

Test using the preset dataset

If want to test the system using the preset dataset `test_container.xlsx` in `web\WBTBSsystem`, the dates in data set need to be changed.

For example, if want to test on 4/10/2024:

```
CN0001  1,516   China   4/3/2024  9:00   INBOUND    1
```

Should be changed to

```
CN0001  1,516   China   4/10/2024  9:00   INBOUND  1
```

Q & A

- **`vue-cli-service` is not a command**

If there is error when run `npm run serve`:

```
'vue-cli-service' 不是内部或外部命令，也不是可运行的程序
或批处理文件。
```

please open folder `website\NingboHarbour` in command prompt and run command:

```
npm install
```

- **Proxy error**

If there is error when access the server from front-end and prompt show that:

```
Proxy error: Could not proxy request /api/login/ from localhost:8080 to
http://127.0.0.1:8000.
See https://nodejs.org/api/errors.html#errors_common_system_errors for more
information (ECONNREFUSED).
```

Please make sure that **back-end has been started**.

Open folder `web\WBTBSsystem` in command prompt and run command:

```
python manage.py runserver
```

- **Web page is blank page**

If open the server and the web page is blank page, please first close the back end and open folder
`web\WBTBSsystem` in conmmand prompt and run command:

```
python manage.py collectstatic
```

then restart back-end.