Shark

リリース *0.1*

YST

2022 年 12 月 01 日

目次

第 1 章　Contents 3

1.1 Usage 3

Shark Timer is a Python based timer app that can calculate time for test and play anouncements.

Check out the *Usage* section for further information, including how to installation the project.

注釈: This project is under active development.

第 1 章

Contents

1.1 Usage

1.1.1 Quick Start

The quickest start is to use an exe file built for windows with mainly `python3.11`, `streamlit1.14.0` and `pyinstaller`. Creating a shortcut of the exe in *../dist/OSCE_timer/OSCE_timer.exe* may help us to find the exe in a shorter time.

To start the app, just click the shortcut.

1.1.2 Calibration

Please notify the timer should be calibrated before real run.

The method of calibration is shown below.

Turn the `Debug Mode` to `on`.

Turn the `Setting` to `calibration`.

Setting other necessary parameters.

Click `Set`.

Click `Start`.

Once the timer has been calibrated, the timer can work.

1.1.3 Assessment

To assess whether the timer is correct while a whole run. We have a `Debug Mode`.

Set `Debug Mode` to on.

Set `Setting` to `Default`.

Setting other necessary parameters.

Click `Set`.

Click `Start`.

Once the timer finishes its run, the elapsed time and standarized elapsed time will be shown in the main window.

注釈: Here we get the standarized time from [WorldtimebApi](http://worldtimeapi.org/). Therefor, for both `Calibration` and `Assessment`, internet acess is needed.

```
import requests
from bs4 import BeautifulSoup
import datetime
def get_wt_time(URL = "http://worldtimeapi.org/api/timezone/Asia/Tokyo"):
  r = requests.get(URL)
      soup = BeautifulSoup(r.text, "html.parser")
      time_at_now = re.findall(r'.*\"datetime\":"(.*)",\"day_of_week\"', soup.
↪contents[0])[0]
      return(datetime.datetime.strptime(time_at_now, '%Y-%m-%dT%H:%M:%S.%f%z'))
```

1.1.4 Start from defined time

To start the timer from a previously defined time, we can directly to input the time to the `Start time in plan` cell. If the inputted time is behind the current time, after we click the `Start` button, we will see a clock running until the inputted time. Otherwise, the timer will start immediately.

1.1.5 Changing the voice of anouncement

Please directly modify the setting file named `define_sound_files.txt` in `../dist/OSCE_timer/`.

Replace the audio file names if needed and meanwhile put those newly added files in the same folder.

1.1.6 Build from source

The exe file of this app was built using `pyinstaller`. Although other exe building software may also work, we did not check.

Flowing the [instruction of streamlit on building exe file](https://discuss.streamlit.io/t/streamlit-deployment-as-an-executable-file-exe-for-windows-macos-and-android/6812/42?page=2), we used the following commands and import all packages used in this tool.

```
import time
import base64
import streamlit
import requests
import datetime
import re
from bs4 import BeautifulSoup
import streamlit.web.cli as stcli
import sys

if __name__ == "__main__":
  sys.argv=["streamlit", "run", "OSCE_timer_ver8.0.py", "--global.developmentMode=false"]
  sys.exit(stcli.main())
```

```
(.venv) $ pyinstaller.exe --noconsole --copy-metadata streamlit --collect-data streamlit␣
↪.\OSCE_timer.py --icon=favicon.ico --clean
```