

ООО „КРЮГЕР ХАУС“

Документация по блоку обмена между Artix и УНФ в части кассовых смен
Конфигурация УНФ 3.0



Новосибирск, 2023 г.

Оглавление

	Стр
1 Установка и подготовка к использованию	1
2 Настройка запуска по расписанию	2
3 Описание алгоритма	3
3.1 Алгоритм	3

Пт.
27
Янв.
2023

1 Установка и подготовка к использованию

- На сервере должен быть установлен Python версии ≥ 3.11
- Копируем каталог «Workshift_load» с программой на рабочий сервер
- Заходим в каталог «src» и выполняем команду

```
chmod +x wsh_load.py
```

Для того, что бы сделать файл скрипта исполняемым

- Если отсутствуют, то создаем два файла «last_date_open.txt» и «last_date.txt». В каждый файл записываем любую дату ранее текущей в формате «2023-01-25 20:31:01».
- В корневом каталоге выполняем команду для создания виртуального окружения

```
python3.11 -m venv .venv
```

- Выполняем команду для установки нужных пакетов

```
pip install -r requirements.txt
```

2 Настройка запуска по расписанию

- Выполняем команду

```
crontab -e
```

- В открывшемся редакторе в конец файла добавить строку

```
* * * * * /home/administrator/Workshift_load/src/wsh_load.py $HOME/command.log 2>&1
```

Сохранить файл. Теперь при таких настройках скрипт будет запускаться каждую минуту. Это необходимо для отладки. Для работы нужно выставить нужный интервал запуска.

3 Описание алгоритма

- Структура проекта

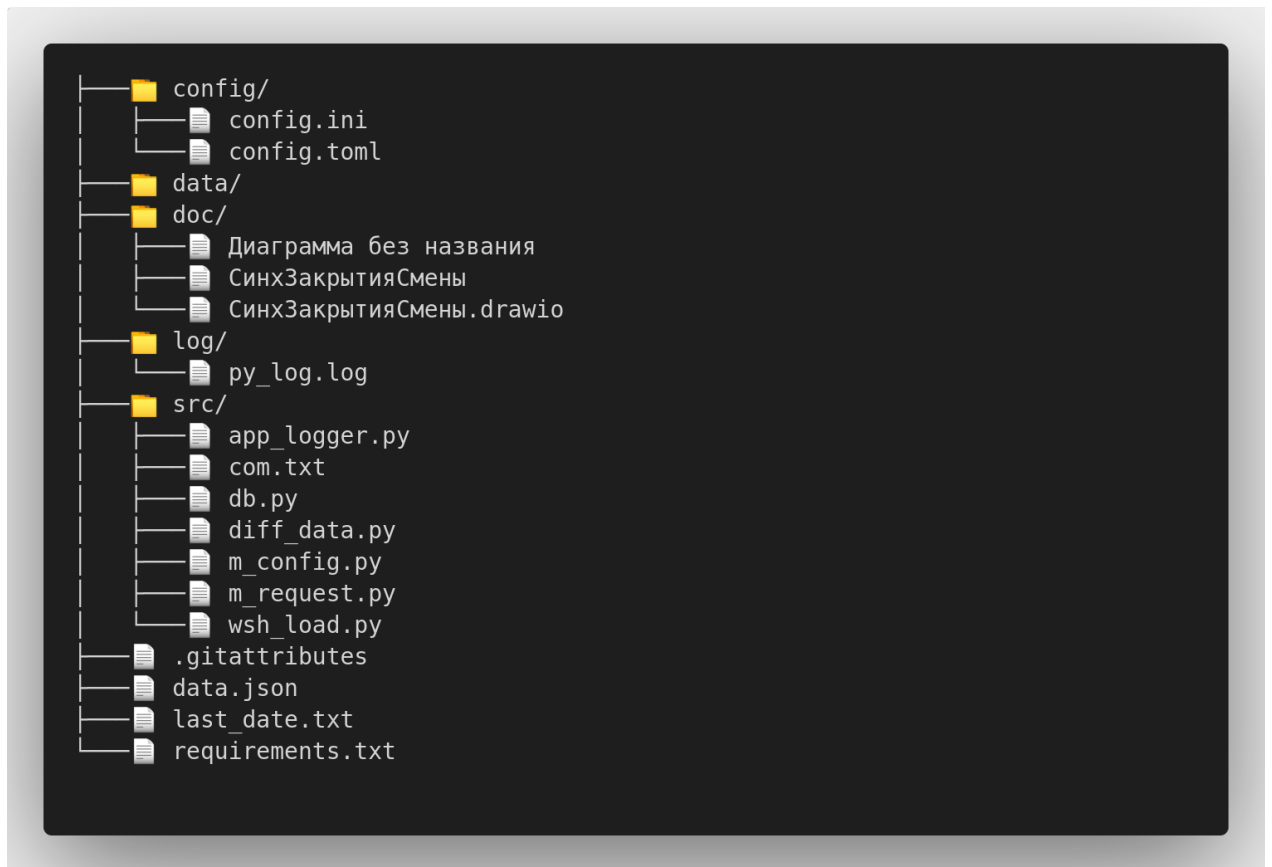


Рисунок 3.1 – «Структура проекта».

3.1 Алгоритм

- При запуске программы происходит чтение настроек. Если чтение успешно, запускается функция «main()», в противном случае программа завершает свою работу.

```
if __name__ == "__main__":

    # Чтение настроек
    m_conf = m_config.m_Config()
    rc = m_conf.loadConfig()
    if not rc == None:
        main()
    else:
        logger.info(u'Программа завершила работу')
```

- При запуске функции «main()» происходит соединение с базой данных и создается объект для работы с ней, так же создается объект для работы с Http запросами.

```
tData = db.workDb(rc)
rec_con = m_request.req1C(rc)
```

- Получаем список смен которые были открыты с последней зафиксированной даты, если появились новые открытые смены, тогда формируем и отправляем Http запрос в 1С. Если код возврата был успешным (200), тогда меняем дату в файле, на дату открытия последней смены.

```
# Список открытых смен от последнего зафиксированного времени
l_workshift_open = tData.get_last_workshift_open()
# Если нечего отправлять, то не отправляем
if len(l_workshift_open) > 0:
    status_code = rec_con.post_workshift_open(l_workshift_open)
# Меняем дату в файле только в случае успешного результата работы 1С
if status_code == 200:
    tData.save_new_date_open()
else:
    logger.info(u'status_code_open - ' + str(status_code ))
```

- Получаем список смен которые были закрыты с последней зафиксированной даты, если появились новые закрытые смены, тогда формируем и отправляем Http запрос в 1С. Если код возврата был успешным (200), тогда меняем дату в файле, на дату закрытия последней смены.

```
# Список закрытых смен от последнего зафиксированного времени
l_workshift = tData.get_last_workshift()
# Если нечего отправлять, то и не отправляем
if len(l_workshift) > 0:
    status_code = rec_con.post_workshift(l_workshift)
# Меняем дату в файле только в случае успешного результата работы 1С
if status_code == 200:
    tData.save_new_date()
else:
    logger.info(u'status_code - ' + str(status_code ))
```

- Завершаем работу программы.

Список иллюстраций

3.1	«Структура проекта».	3
-----	----------------------	---

Журнал изменений

Версия	Дата	Автор(ы)	Изменения
0.01	27.01.23	РК	Начало работы