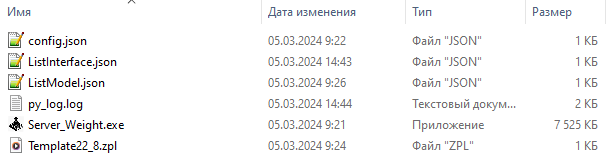
**Документация по весам.**

**Серверная часть весов (Основная).**

Это обычное консольное приложение, обеспечивающее связь между весами и принтерами, также поддерживает простой HTTP Сервер для связи с клиентом через средства API. В папке с этой программой обязательно должны находиться данные файлы: 

1. config.json – Файл конфигурации.
2. ListInterface.json – Файл сохраненных Интерфейсов.
3. ListModel.json – Файл сохраненных Моделей.
4. Template22\_8.zpl – Файл шаблона этикетки.
5. py\_log.log – Файл логирования.

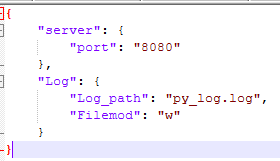
Если этих файлов не будет, ничего страшного, программа их создаст сама, только файлы Интерфейсов и Моделей будут без данных.

**ВАЖНО ПОМНИТЬ!!!!**

1. При запуске данной программы COM-порты должны **иметь возможность открываться.**
2. При этом также должна быть открыта программа **VirCOM.exe,** в которой нужные порты должны иметь статус **Connected**.
3. Если вы изменили данные сохраненных интерфейсов, моделей или конфигурации, то нужно **ОБЯЗАТЕЛЬНО перезапустить сервер**, чтобы изменения вступили в силу. Изменения также могут вступить в силу при работе с клиентским приложением, если изменения отправлять оттуда.

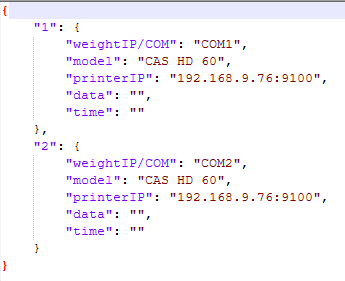
**Файл конфигурации сервера. Структура файла:**

1. Номер порта для связи сервера с клиентским приложением
2. Путь и название файла Логирования
3. Тип записи файла Логирования (w – записывает файл заново / a – дописывает в конец файла)



**Файл сохраненных интерфейсов. Структура файла:**

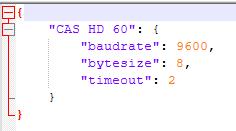
1. Технологический номер весов. (Является уникальным полем)
   1. COM-порт для связи интерфейса с весами. (Является уникальным полем) В будущем, возможно, еще добавится и общение по IP.
   2. Название модели весов.
   3. IP Принтера и порт, на котором должна будет печататься этикетка.
   4. Поле для Даты (имеет формат **дд.мм.гг**); может быть пустым, тогда дата будет подтягиваться с компьютера.
   5. Поле для времени (имеет формат **чч:мм**); может быть пустым, тогда дата будет подтягиваться с компьютера.

****

**Файл сохраненных моделей. Структура файла:**

1. Название модели (Является уникальным полем). В названии модели не должно быть русских букв *(косяк программы, неправильно происходит чтение и кодирование информации)*
   1. Baud rate
   2. Byte size
   3. Timeout

Обычно эти значения зависят от самого COM-порта.

****

**Запросы API (Не обязательно к прочтению):**

**GET:**

1. /Interfaces – возвращает словарь сохраненных Интерфейсов весов.
2. /Models – Возвращает словарь Моделей весов.

**POST:**

1. /AddInterface {data} – [формат data – Такой же, как и структура Интерфейса] Добавляет/Изменяет полученный Интерфейс. Если Интерфейс был изменен, то перезапускает его с новыми данными. Возвращает **код ответа**: 200 – если не удалось запустить интерфейс, 201 – при успехе, 400 – при возникновении ошибки. Также возвращает **текст**: *Not Added, Added, Error* соответственно.
2. /ChangeInterface {data} – Аналогично.
3. /AddModel {data} – [формат data – Такой же, как и структура Модели] Добавляет/Изменяет полученную Модель. Если Модель была изменена, то перезапускает Интерфейсы с этим типом Модели. Возвращает **код ответа**: 200 – если не удалось запустить интерфейс, 201 – при успехе, 400 – при возникновении ошибки. Также возвращает **текст**: *Not Added, Added, Error* соответственно.
4. /ChangeModel {data} – Аналогично.

**PUT:**

1. /ReloadInterfaces – Перезапускает все доступные из файла интерфейсы. Возвращает текст: *Reloaded* – при успехе(так всегда бывает)*, Not reloaded* – при возникновении ошибки.
2. /ChangeInterface {data} – См. POST.
3. /ChangeModel {data} – См. POST.

**DELETE:**

1. /Interfaces {data} - [формат data – словарь типа: {‘Технологический номер весов’ : True}] Удаляет доступные интерфейсы, если некоторые из них были запущены, то прекращает их работу. Возвращает **код ответа**: 201 – если удалось удалить интерфейс, 400 – при возникновении ошибки. Также возвращает словарь типа: {‘Технологический номер весов, которые не удалось удалить’ : False}.
2. /Models {data] - [формат data – словарь типа: {‘Название модели весов’ : True}] Удаляет доступные модели, если были Интерфейсы, которые использовали эти модели, то прекращает их работу. Возвращает **код ответа**: 201 – если удалось удалить модели, 400 – при возникновении ошибки. Также возвращает словарь типа: {‘Название моделей весов, которые удалось удалить’: True}.