

## Веб-сервер

Веб-сервер, размещается на firstbyte.ru.

Пользователь подключается к веб-серверу через веб-браузер. Задает уставки, включает, отключает холодильную установку, отображает температуру.

<https://github.com/serereg/nic_srv>

## Компьютер-клиент с функцией OPC-сервера

Компьютер-клиент расположен в здании на производстве. Компьютер-клиент опрашивает веб-сервер для получения команд от пользователя и предоставления ему текущей информации.

<https://github.com/serereg/nic_cl>

На компьютере-клиенте работает OPC-сервер для управления ПЛК-контроллером, приложение подключается к OPC-серверу через библиотеку openopc.

*OPC-сервер и связь с ПЛК налажена.*

# Отдельные пояснения

*https://github.com/serereg/nic\_srv/blob/master/pargolovo\_server/\_\_main\_\_.py*

*# async def \_simulate\_commands(self):*

*# import random*

*# while True:*

*# await asyncio.sleep(5)*

*# if not self.\_opc\_clients:*

*# continue*

*# cmd = dict(type="command", action=random.choice(["on", "off", "set"]),*

*# item="node1", value=random.randint(-20, 20))*

*# logger.info("sending command {} to {} clients", cmd, len(self.\_opc\_clients))*

*# await asyncio.gather(\*[ws.send\_json(cmd) for ws in self.\_opc\_clients])*

*async def ws\_browser\_handler(self, request):*

*pass*

Закомментированная часть – имитация случайных команд "on", "off", "set, value=[ -20, 20 ].  
Но эти команды должны передаваться с веб-клиента (javascript, например).

*Внешний вид веб-страницы*

Вкл

Откл

Текущая

Задание