МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего

образования

«Санкт-Петербургский государственный университет аэрокосмического приборостроения»

Кафедра информационно-сетевых технологий

Дисциплина «Интернет вещей»

Автор – ассистент Д.О. Шевяков

Лабораторная работа № 4

ОРГАНИЗАЦИЯ ПЕРЕДАЧИ УПРАВЛЯЮЩИХ КОМАНД НА УДАЛЁННОЕ ОБОРУДОВАНИЕ

Цель работы – приобретение навыков организации передачи управляющих команд на удаленное оборудование.

Методические указания

Процесс передачи управляющих команд аналогичен получению данных мониторинга, только управляющие команды передаются, как запрос к оборудованию с командами управления.

В первую очередь модифицируем наш эмулятор таким образом, чтобы в нём появилось поле для ввода отправляемой команды. Вид модифицированного эмулятора представлен ниже, а его код в Приложении 1.

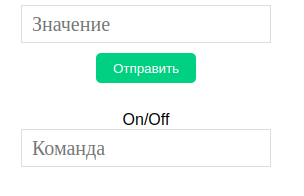


Рисунок 1 – Эмулятор с полем для отпрвки данных

Далее нам необходимо отправить данные на сервер. Сделать это можно при помощи все того же ajax-запроса. Пример представлен ниже.



Рисунок 2 – Код ajax – запроса для отправки данных

Далее необходимо модифицировать наш метод connect таким образом, чтобы он обрабатывал json-объект, содержащий данные, которые мы хотим передать на оборудование. Внутри этой функции вызовем метод connect нашей вещи и передадим ему наш request. Внутри нашего метода вытащим из нашего GET-запроса необходимую информацию и присвоим её нашей вещи. request – это объект, хранящий в себе данные GET – запроса, который был передан функции, чтобы получить к нему доступ, его также надо добавить в первую строку файла app.py. Код функции роутинга и метода вещи представлен ниже.

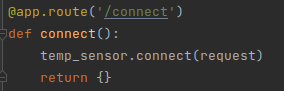


Рисунок 3 – Код функции connect

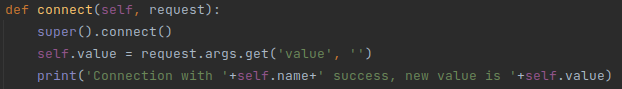


Рисунок 4 – Метод connect вещи Sensor

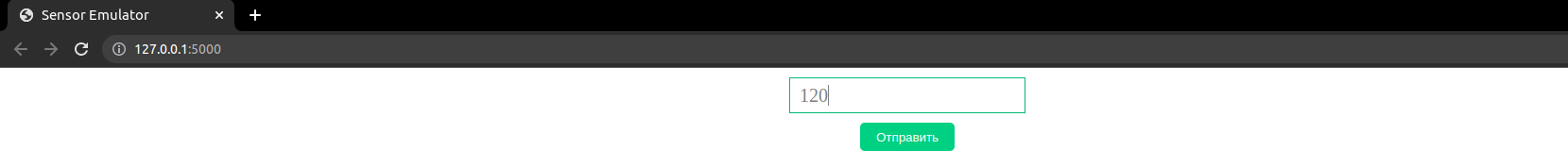


Рисунок 6 – Страница управления

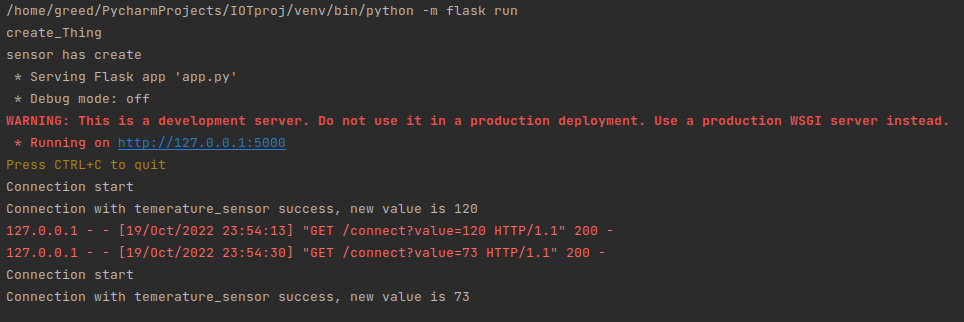


Рисунок 7 – Результат подключения

Задание

Организовать передачу управляющих команд из интерфейса на оборудование для всех вещей. Передачу управляющих команд можно реализовать как при помощи нескольких запросов, по одному для каждой вещи, так и одним запросом, который будет передавать все команды, а их обработкой будет заниматься отдельная функция на сервере. В рамках данной лабораторной работы необходимо реализовать передачу именно управляющих команд, переделывать запросы из прошлой лабораторной не нужно.

Содержание отчёта:

1. Титульный лист
2. Задание
3. Код классов и приложения
4. Скриншоты
5. Выводы

Приложение 1 sensor\_emulator.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Sensor Emulator</title>

<link rel="stylesheet" type="text/css" href="../static/css/main.css">

<script src="http://code.jquery.com/jquery-3.3.1.js"></script>

<script src="../static/js/index.js"></script>

</head>

<body>

<div id="monitoring\_data">

<input type="text" id="value" placeholder="Значение">

<button onclick="send\_data()">Отправить</button>

<br>

<labek>On/Off</labek>

<input type="text" id="command" disabled="disabled" placeholder="Команда">

</div>

</body>

</html>

Приложение 2 things.py

import abc

import json

class Thing(abc.ABC):

def \_\_init\_\_(self, name):

self.name = name

print('create\_Thing')

@abc.abstractmethod

def connect(self, \*args):

print('Connection start')

class Sensor(Thing):

def \_\_init\_\_(self, unit, name):

super().\_\_init\_\_(name)

self.unit = unit

self.value = 0

self.power = "on"

print('sensor has create')

def connect(self, request):

super().connect()

self.value = request.args.get('value', '')

print('Connection with '+self.name+' success, new value is '+self.value)

return json.dumps({"power": self.power})

Приложение 3 app.py

from flask import Flask, request, render\_template

import things

app = Flask(\_\_name\_\_)

temp\_sensor = things.Sensor('C', 'temerature\_sensor')

@app.route('/connect')

def connect():

return temp\_sensor.connect(request)

@app.route('/')

def hello\_world(): # put application's code here

return render\_template('sensor\_emulator.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run()

Приложение 4 index.js

function send\_data() {

$.ajax({

type: 'GET',

url: '/connect',

dataType: 'json',

contentType: 'application/json',

data: {

"value": document.getElementById("value").value,

},

success: function (response) {

document.getElementById("command").value = response["power"]

}

});

}