

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Trabalho final 1

IOT – Casa Smart

Ana Júlia Santana

Getúlio Martins Resende

Guilherme Ferreira

Henrique Santos de Lima

Uberlândia

06/05/2021

| | |
|------------------------------|----------|
| Requisitos do projeto | 3 |
| Diagrama lógico do projeto | 3 |
| Comandos | 3 |
| Interface Web | 4 |
| Servidor | 4 |
| Arduino | 4 |
| Dispositivos | 4 |
| Desenvolvimento | 5 |
| Servidor | 5 |
| Interface web | 5 |
| Gerenciador | 5 |
| Comunicação | 5 |
| Microcontrolador | 6 |
| Referências | 7 |

Requisitos do projeto

O projeto consiste em uma aplicação do conceito de IOT, do inglês Internet das Coisas, que se traduz em uma aplicação conjunta entre Python, Flask em um ambiente Linux que se comunica com o microcontrolador Arduino, nos permitindo mudar seus estados de saída que podem interpretados como alimentadores de relés que, por fim, permite que dispositivos elétricos de uma casa inteligente possa ser retirados da energia elétrica com comandos provenientes de um website, com interface homem-máquina de fácil compreensão e uso.

As funcionalidades desta aplicação incluem as seguintes:

- Ativa ou desativa um dispositivo.
- Adiciona ou remove novos dispositivos.
- Renomeia um dispositivo.

Diagrama lógico do projeto

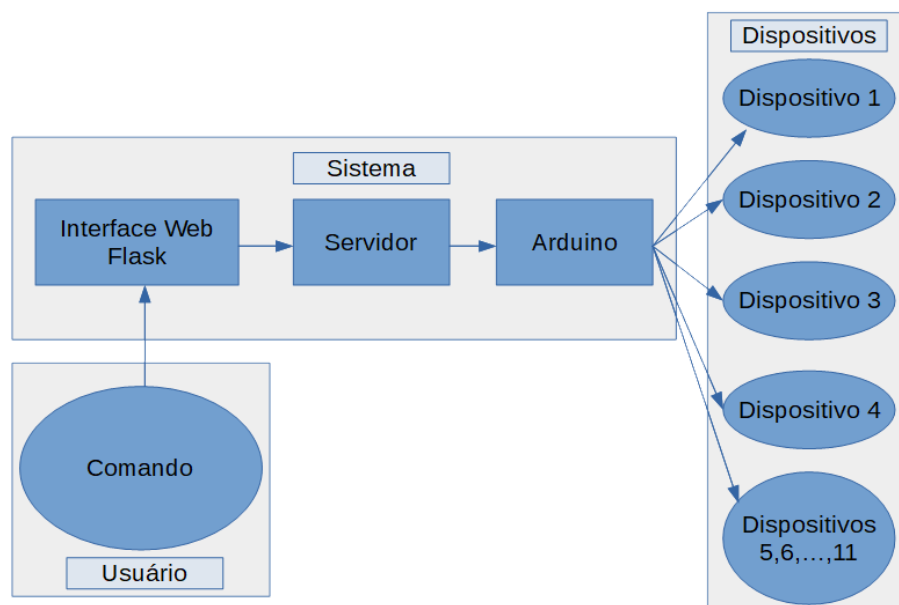


Figura 1 - Diagrama lógico do projeto

Comandos

O usuário terá liberdade de adicionar, excluir e renomear os dispositivos que serão controlados por meio do sistema. Este também poderá ligar ou desligar os dispositivos conectados à energia elétrica. Nosso projeto permite que o usuário possa controlar até n

tomadas, onde n é o número de gpios, por meio de uma interface Web que se faz possível graças ao Flask.

Interface Web

A interface web é criada com o auxílio do Flask, que permite ao usuário dar os comandos para o sistema de maneira gráfica. A interface em questão é um elo entre o usuário e o servidor do sistema, assim tornando mais agradável o uso do sistema para o usuário, eliminando o uso de linhas de comando ou mesmo uso de simuladores de terminal para a execução de um comando.

Servidor

O servidor é a parte do sistema que manipula toda a informação que é recebida da interface web e promove toda a computação necessária do processo e então entrega ao microcontrolador do sistema qual ação ele deve tomar.

Arduino

O Arduino é a fração do sistema que recebe dados do servidor e com isso alimenta relês ou transístores que permitem ligar ou desligar dispositivos.

As portas digitais deste microcontrolador em questão permitem interagir com cargas pequenas, genericamente falando, de apenas duas maneiras: Alimentando-as com 5 Volts (CC), considerando o Arduino Uno como microcontrolador, ou não alimentando-as. Desta forma este sistema poderia ligar estas saídas, dimensionando resistores para tal, em relês, que são elementos eletromecânicos permitindo que dispositivos eletricamente alimentados (geralmente de grande potência) possam perder esta alimentação assim que a alimentação destes relês mudam de estado, o que também é verdade para elementos elétricos semicondutores chamados transistores que podem trabalhar da mesma maneira dita anteriormente, porém com cargas de menor potência, como alto-falantes de mesa, por exemplo.

Dispositivos

Os dispositivos são os elementos que vão receber uma ação, esta ação pode ser, ligar ou desligar. Estes dispositivos são tomadas de uso geral ou específico.

Desenvolvimento

O projeto[1] foi dividido em duas partes, servidor e microcontrolador. O primeiro, responsável por todo processo desde a interface usuário-máquina passando pelo gerenciador e enviando um sinal para o microcontrolador. O segundo, responsável por receber um sinal e a partir deste ativar ou desativar suas portas lógicas de saída.

Servidor

Interface web

A interface é a parte responsável pela comunicação entre o usuário e a classe do gerenciador. Ela foi pensada de forma a tornar o uso o mais amigável possível, evitando possíveis problemas que possam surgir do usuário solicitando ações que não existem, ou inserindo dados de forma incorreta.

Para fazer isso, além do Flask[2], foi utilizado o Bootstrap para otimizar os acréscimos de estilização à interface.

Gerenciador

O gerenciador é uma classe responsável por controlar as tomadas, com seu status, nome e seu identificador. A partir do gerenciador, a interface web consegue se conectar com o dispositivo a partir da comunicação, também, o gerenciador permite ter controle de quais portas ainda estão disponíveis para registrar um novo dispositivo e controlar seu nome e status.

Comunicação

A comunicação é uma classe responsável por estabelecer conexão serial[4] com o arduino, trocando dados com o mesmo. Os dados foram convertidos para valores em binário, com o protocolo descrito abaixo.

Para o envio, o primeiro byte corresponde ao número subsequente de bytes a serem enviados. O segundo byte corresponde ao comando que o microcontrolador deverá executar, os próximos bytes são valores que correspondem aos parâmetros de cada comando. Este protocolo visa garantir que os dados sejam um pacote fechado de n bytes.

Para receber é adicionado um ouvinte (Tarefa em segundo plano) que verifica se chegou alguma informação do serial. O primeiro byte recebido corresponde ao número subsequente de bytes a serem recebidos, o segundo byte corresponde ao identificador do comando, o terceiro byte corresponde ao número de agrupamento, e os seguintes bytes são parâmetros de cada comando. Após receber os dados e convertê-los devidamente, é chamada a função callback, contendo o comando e o objeto de retorno.

Microcontrolador

Foi implementado a conexão serial e validado as gpios(input/output de uso geral) disponíveis . A cada iteração é verificado se existe uma mensagem na conexão serial. É seguido o protocolo apresentado na seção Servidor/Comunicação.

Referências

[1]O repositório do projeto se encontra em:

<https://github.com/GUI-FERREIRA/Projeto_IOT>

[2]Documentação Flask. Disponível em: <<https://flask.palletsprojects.com/en/1.1.x/>>.

Acesso em: 28 de abril, 2021.

[3]Digital GPIO of Arduino. ElectronicWings. Disponível em:

<<https://www.electronicwings.com/arduino/digital-gpio-of-arduino>>. Acesso em: 28 de abril, 2021

[4]Pyserial. Disponível em: <<https://pythonhosted.org/pyserial/>>. Acesso em: 28 de abril, 2021.

Arduino. Disponível em: <<https://www.arduino.cc/>>. Acesso em: 30 de abril, 2021.

Bootstrap. Disponível em: <<https://getbootstrap.com/docs/5.0/>>. Acesso em: 30 de abril, 2021