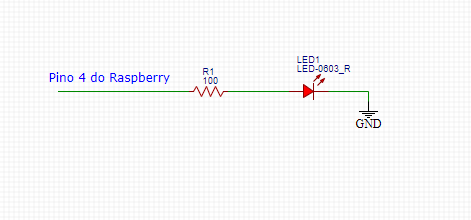
Desafio Estagio ICC HW

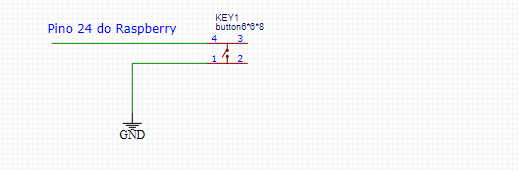
# desenvolvimento

Para o desafio proposto primeiramente será necessario configurar o acesso a Internet via Wifi, depois de configurado, fiz o esquema elétrico para o led e o botão respecitvamente, utilizando o software EasyEDA para elaboração do mesmo.



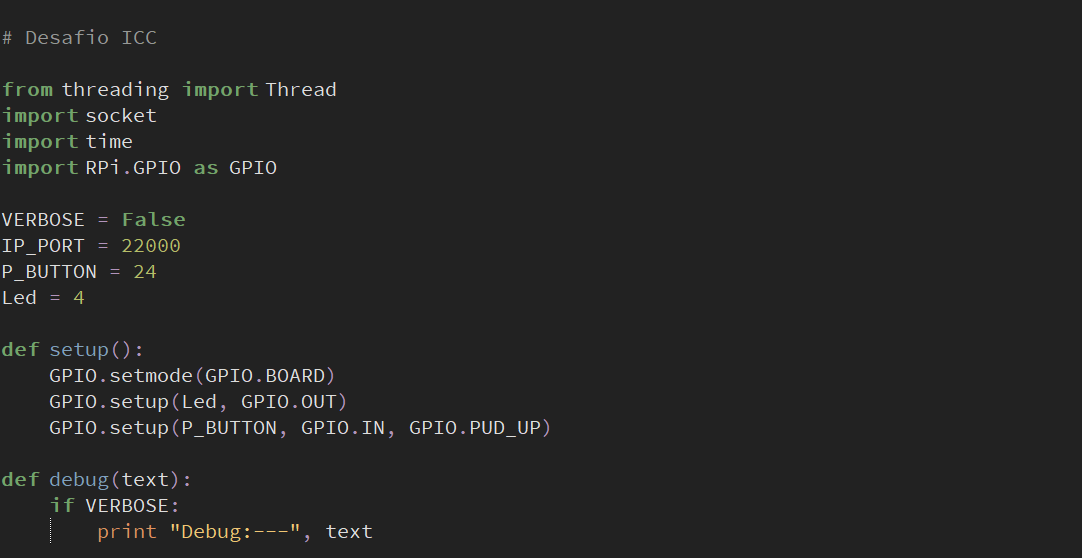
Decidi escolher o pino 4 para o led, para o calculo do resistor utilizei as informações do datasheet do modelo zero w e do led para encontrar o valores de tensão e corrente. Na saida do pino escolhido o valor é de 3,3V, e o valor da queda de tensão do led que é de 2V, já para a corrente do led o valor é de 20mA.

Utilizando a formula R1 = VR1 / I em que R1 é o valor do resistor desejado, VR1 é o valor da tensão encontrada no resistor que é dado pela formula (V – Vled) e I é o valor da corrente do Led, obtem-se o valor do resistor é de 65 ohms mas como não há esse valor no mercado utilizei o resistor de 100 ohms.

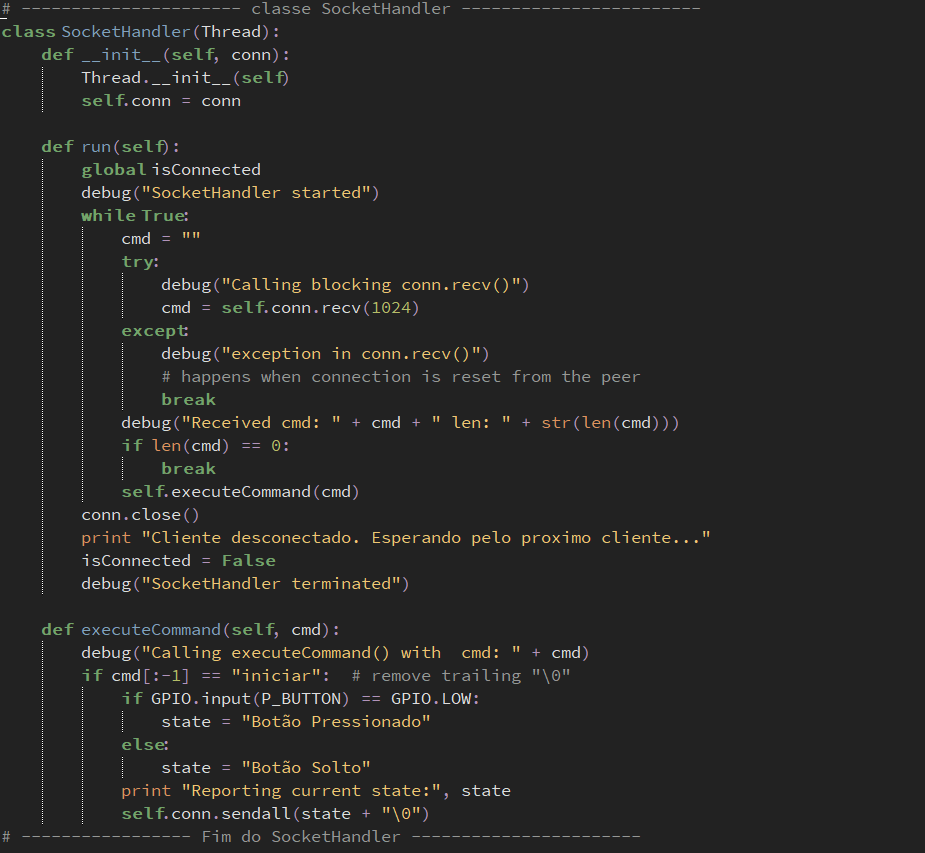


Para o botão decidi escolher o pino 24, não foi necessario um resistor então somente o liguei diretamente ao pino escolhido do microcontrolador.

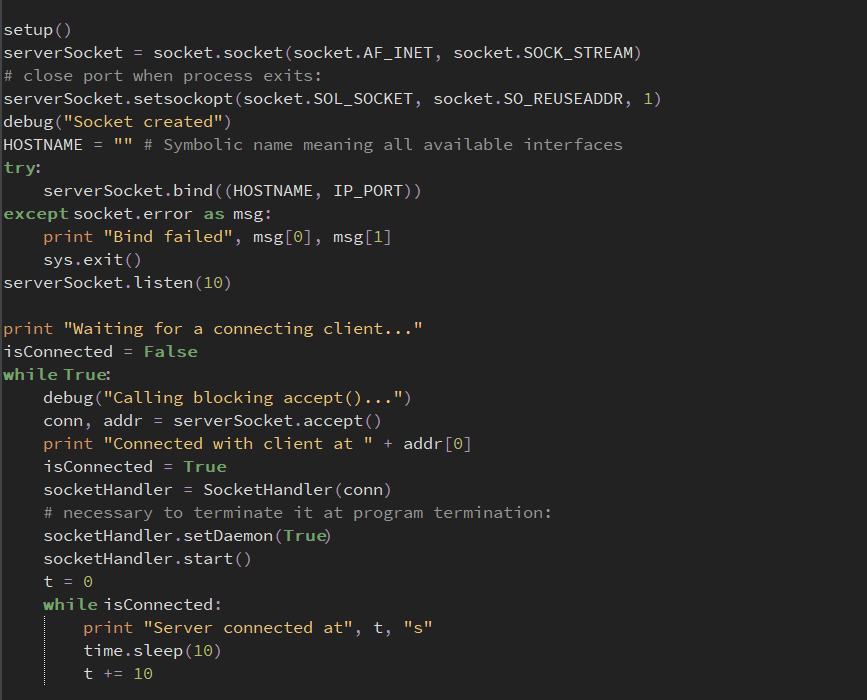
Para a parte de codificação decidi usar a IDE Mu por acha-la mais intuitiva e simples para se utilizar



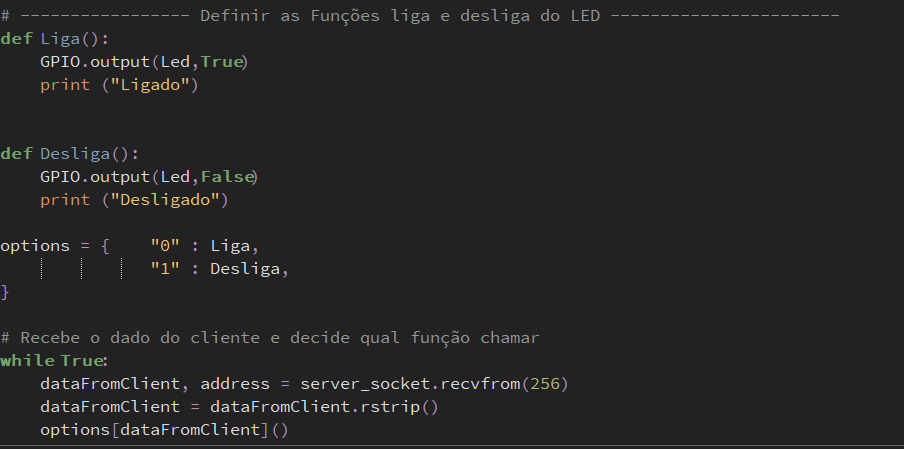
Aqui é onde importei as bibliotecas necessarias para implementação do codigo, decidi por usar a porta 22000 para não haver possiveis conflitos com outras aplicações tcp/ip que poderão ser usadas, defini as respectivas entradas GPIO para maior organização do codigo.



A classe SocketHandler é responsavel pela verificação constante se há algum servidor conectado e se o mesmo foi desconectado. A função executeCommand é responsavel por verificar se o usuario digitou o comando iniciar para começar o envio do estado do botão.



Essa parte do codigo é responsavel pela conexão com o servidor ele tenta fazer a conexão se não consegue exibe o erro no console e se conseguir com exito mostra qual o endereço IP do servidor e a quanto tempo ele está conectado.



E por fim a parte responsavel pela indicação no led se o botão está ligado ou não, criei duas funções cujo nome é Liga e Desliga e configurei para caso o servidor envie 0 será entendido como o botão pressionado assim ligando o led e caso envie o comando 1 será entendido que o botão está solto e desligará o mesmo.

# Referências

Raspberry. **Raspberry:** Pi Documentation. Página inicial. Disponível em: < https://www.raspberrypi.org/documentation/usage/gpio/>. Acesso em: 19 de jun. de 2020.

ARCADE, BurgZerg. Raspberry Pi Button Input. 2017. (13m07s). Disponível em: <https://www.youtube.com/watch?v=LEi\_dT9KDJI>. Acesso em: 19 jun. 2020.

Raspberry Foruns. **Raspberry**: Python TCP/UDP Server-Client for controlling RPi. Página inicial. Disponível em: <https://www.raspberrypi.org/forums/viewtopic.php?t=205197>. Acesso em: 19 de jun. de 2020.

Python exemplarisch. **Python exemplarisch:**TCP/IP COMMUNICATION. Página inicial. Disponível em: <http://www.pythonexemplarisch.ch/index\_de.php?inhalt\_links=navigation\_de.inc.php&inhalt\_mitte=raspi/de/raspi.inc.php/index\_en.php?inhalt\_links=navigation\_en.inc.php&inhalt\_mitte=raspi/en/communication.inc.php >. Acesso em: 19 de jun. de 2020.