



UNIVERSIDADE FEDERAL DO CEARÁ  
CENTRO DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA  
DISCIPLINA DE TÓPICOS ESPECIAIS EM ENGENHARIA DE  
TELEINFORMÁTICA



FRANCISCO CARLOS FREIRE NUNES JUNIOR - 371802

LAILSON AZEVEDO DO REGO - 371828

SAYONARA SANTOS ARAÚJO - 371869

## **FORTALEZA**

**2016**

Controle de versão		v2.0
Data	Modificação	Autor
25/10	Criação do documento	Carlos, Lailson e Sayonara
17/11	Adição do tópico de Comunicação em Resultados	Carlos
08/12	Adição do tópico do sensor de temperatura	Sayonara

## **SUMÁRIO**

<b>1. INTRODUÇÃO</b>	<b>3</b>
<b>2. MATERIAL</b>	<b>4</b>
<b>3. OBJETIVO</b>	<b>5</b>
<b>4. FUNDAMENTAÇÃO TEÓRICA</b>	<b>6</b>
4.1. Arduino	6
4.2. Sensor de Chuva	7
4.2.1. Especificações	7
4.2.2. Funcionamento	7
4.3. Módulo sensor de Umidade	8
4.3.1. Especificações	8
4.3.2. Funcionamento	8
4.4. Módulo relé	9
4.4.1. Especificações	9
4.4.2. Funcionamento	10
4.5. Sensor de temperatura	11
4.5.1. Especificações	11
4.5.2. Funcionamento	11
4.6. ESP8266	12
4.6.1. Introdução	12
4.6.2. Especificações	12
4.6.3. Tecnologia de baixo consumo	13
4.6.4. MCU	13
4.6.5. Programação	13
4.6.6. Funcionamento	13
<b>5. RESULTADOS</b>	<b>14</b>
5.1. Sensoriamento	14
5.2. Comunicação	15
5.2.1. Servidor	15
5.2.2. Módulo cliente	15
5.3. Aplicação	17
<b>6. BIBLIOGRAFIA</b>	<b>20</b>

## **1. INTRODUÇÃO**

O manejo de água é um tema preocupante. Em grandes plantações, sistemas de pivôs abastecem áreas de cerca de 250 mil hectares. Logo, grandes investimentos estariam correndo um risco de desabastecimento e desperdício caso não exista um bom controle de fornecimento de água nessas regiões.

O problema de desperdício de água persiste em ambientes caseiros como em jardins e hortas. Dessa forma, o uso de água de irrigação em ambientes caseiros foi escolhido como ponto de partida desta pesquisa devido a facilidade de abstração e por ser um dos segmentos que mais utiliza recursos hídricos.

Diante disso, For Garden's Sake tem como objetivo montar um sistema de monitoramento de jardim ou plantação, focando no controle do fornecimento da água de irrigação, no momento e em quantidade adequados. Portanto, o sistema fornecerá mais água em períodos de seca, e evitará desperdício em períodos chuvosos. Além disso, a gestão será feita através de uma plataforma IoT que possibilitará a consulta de informações coletadas em tempo real.

## **2. MATERIAL**

- 2 x Arduino Uno;
- Módulo relé;
- Módulo sensor de umidade do solo;
- Módulo sensor de chuva;
- 2 x Módulo Wifi esp8266 esp-01;
- 2 x Protoboard;
- Resistores diversos;
- Válvula Solenóide 220V.
- Computador.

### **3. OBJETIVO**

O sistema funcionará em quatro etapas:

1. A abertura da torneira de irrigação se dará por meio de uma válvula solenóide controlada por um Arduino, que será responsável por obter dados do consumo a partir de um sensor de vazão acoplado à torneira. Os comandos de ligar ou desligar serão recebidos do servidor, e as informações de consumo serão enviadas ao servidor.
2. O monitoramento da plantação será realizado por um conjunto de sensores formado por sensor de chuva, sensor de umidade e temperatura (ambiente), além de sensor de umidade do solo. Estes serão controlados por um Arduino responsável por comutar as informações e enviá-las ao servidor.
3. O servidor será composto por um computador que receberá as informações coletadas pelos sensores, e as armazenará em um banco de dados. Através dessas informações, o servidor decidirá quando irrigar a plantação enviando uma tomada de decisão ao Arduino responsável por controlar a torneira.
4. O cliente é composto por uma aplicação móvel que mostrará ao usuário as informações trocadas com o servidor sobre consumo e meio ambiente (temperatura e umidade da plantação). Além disso, a aplicação possibilitará o desligamento do sistema de irrigação.

## 4. FUNDAMENTAÇÃO TEÓRICA

### 4.1. Arduino

O Arduino é uma placa fabricada na Itália com um microcontrolador Atmel AVR, suporte de entrada/saída embutido e uma linguagem de programação essencialmente C/C++, bastante utilizado como plataforma de prototipagem eletrônica de Hardware livre cujo objetivo é tornar a robótica mais acessível.

A fonte de alimentação recebe energia externa por uma tensão de, no mínimo, 7 volts e máximo de 35 volts com corrente mínima de 300mA. A placa e demais circuitos funcionam com tensões entre 5 e 3,3 volts.

As unidades são constituídas por um microcontrolador Atmel AVR de 8 bits, pinos digitais e analógicos de entrada e saída, entrada USB que permite conexão com computadores, entrada serial e código aberto. Por possuir código aberto, há várias modificações que deram origem a derivados “ino”, tais como Marminino, Produino, Garagino, etc.

O Arduino pode ser combinado com outras extensões chamadas shields que são responsáveis por gerar recursos de rede à placa. Embutido na placa há ainda um firmware carregado na memória da placa controladora que aceita Windows, Linux e Mac OS X.

As aplicações com Arduino abrangem diversas áreas tais como designer, arquitetura, engenharia, agronomia, etc; com a placa é possível enviar ou receber informações de basicamente qualquer sistema eletrônico, como desligar uma válvula solenóide ao detectar a presença de chuva, e abrir as janelas de um escritório de acordo com a intensidade da luz do sol e temperatura ambiente.



Figura 1: Arduino Uno.

## **4.2. Sensor de Chuva**

O módulo sensor de chuva é composto por um sensor de chuva e uma placa de controle. Este módulo pode ajudar no monitoramento condições climáticas, detectando gotas de chuva ou neve.

### **4.2.1. Especificações**

- Tensão de Operação: 3,3-5v
- Corrente de Saída: 100mA
- Potenciômetro (trimpot)
- Saída Digital e Analógica
- Led indicador para tensão
- Led indicador para saída digital
- Comparador LM393
- Dimensões Sensor de Chuva: 5x4 cm
- Dimensões Placa de Controle: 2,1x1,4 cm

### **4.2.2. Funcionamento**

O sensor de chuva é revestido de uma malha condutora, desenhada no formato de linhas paralelas e de modo que linhas adjacente estão isoladas umas das outras. Então, quando o sensor está energizado, as linhas apresentam sinal positivo e negativo de forma alternada. Assim, quando cai um líquido condutor no sensor, fechando um circuito entre uma linha positiva e uma negativa, a corrente passa pela malha.

A placa de controle é constituída por um comparador de tensão, um potenciômetro para ajuste de sensibilidade, pinos de alimentação, uma saída digital e uma saída analógica, com valores de 0 (zero) a 1024. Basicamente, a placa amplifica o sinal lido pelo sensor e converte o sinal analógico em digital.

Com isso, quando está chovendo, o módulo apresenta o sinal digital em estado baixo e o sinal analógico mais próximo de zero; no clima seco, o módulo apresenta o sinal digital em estado alto e o sinal analógico mais próximo de 1024.



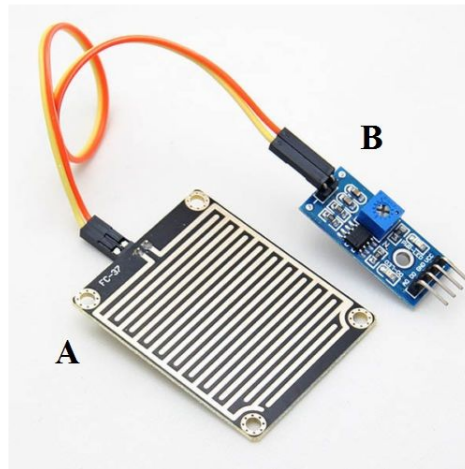


Figura 2: Módulo sensor de chuva. A - Sensor de chuva. B - Placa de controle.

### 4.3. Módulo sensor de Umidade

O sensor de umidade do solo higrômetro é usado para detectar variações de umidade no solo, de modo que sua saída fica em estado alto e baixo quando o solo está seco e úmido, respectivamente. Há dois tipos de saída no sensor de umidade, a analógica A0 que permite melhor resolução possível e a saída digital D0 que é regulada por um potenciômetro interno ao sensor; este potenciômetro permite que o limite entre seco e úmido seja ajustado.

#### 4.3.1. Especificações

- Tensão de Operação: 3,3-5v
- Potenciômetro
- Saída Digital e Analógica
- Led indicador para tensão (vermelho)
- Led indicador para saída digital (verde)
- Comparador LM393
- Dimensões PCB: 3x1,5 cm
- Dimensões Sonda: 6x2 cm

#### 4.3.2. Funcionamento

O sensor de umidade do solo consiste em duas partes: uma sonda que entra em contato com o solo e um pequeno módulo contendo um chip comparador LM393.

Os dois terminais da sonda que entram em contato com o solo, quando energizados, conduzem uma tensão conforme a umidade da terra.

A placa de controle, semelhante a do sensor de chuva, é constituída por um comparador de tensão, um potenciômetro para ajuste de sensibilidade, pinos de alimentação, uma saída digital e uma saída analógica, com valores de 0 (zero) a 1024. Basicamente, a placa amplifica o sinal lido pelo sensor e converte o sinal analógico em digital.

Dessa forma, quando o solo está úmido, o módulo apresenta o sinal digital em estado baixo e o sinal analógico mais próximo de zero; já quando o solo apresenta umidade baixa, o módulo apresenta o sinal digital em estado alto e o sinal analógico mais próximo de 1024.

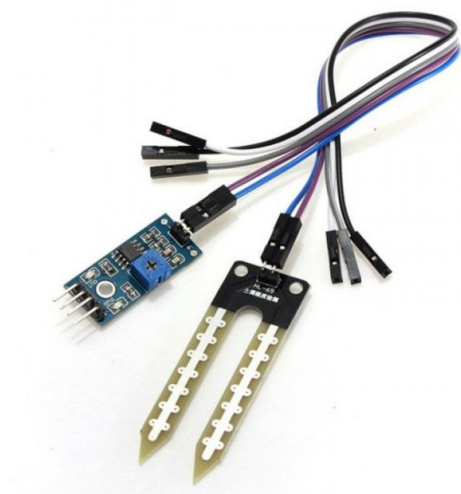


Figura 3: Módulo sensor de umidade do solo higrômetro

#### **4.4. Módulo relé**

O módulo relé, utilizado no projeto, permite uma integração com uma variedade de microcontroladores como Arduino, AVR, PIC, ARM. A partir das suas saídas digitais pode-se controlar cargas altas e dispositivos como motores AC ou DC, eletroímãs, solenóides e lâmpadas incandescentes.

##### **4.4.1. Especificações**

- Tensão de operação: 5 DC (VCC e GND)

- Tensão de sinal: TTL - 5 DC (IN)
- Corrente típica de operação: 15~20mA
- Terminais: NA, NF e o Comum
- Capacidade: 30V DC a 10A ou 250V AC a 10A
- Indicador LED de funcionamento
- Dimensões: 43mm (L) x 17mm (C) x 19mm (H)

#### 4.4.2. Funcionamento

Os relés são chaves eletromecânicas usadas para o acionamento de cargas de alta tensão ou corrente a partir de um circuito de baixa tensão. Um relé é, basicamente, constituído por um eletroímã (bobina), armadura de ferro móvel, conjunto de contatos, mola de rearme e terminais. Quando a bobina é percorrida por uma corrente, é criado um campo magnético, que irá atuar sobre a armadura, atraindo-a e fechando os contatos. Ao cessar a corrente da bobina, o campo magnético deixa de existir, e os contatos voltam à posição original.

O módulo utilizado apresenta pinos de alimentação (VCC e GND); o pino IN que recebe o sinal para comutar a chave; o contato NC (NF) que é normalmente fechado, ou seja, abre quando a bobina recebe corrente, e o contato NO (NA) que é normalmente aberto, fechando quando a bobina recebe corrente.



Figura 4: Módulo relé.

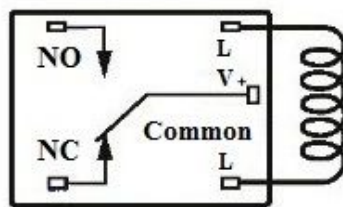


Figura 5: Esquemático do relé.

#### 4.5. Sensor de temperatura e umidade

O DHT11 é um sensor composto de temperatura e umidade relativa do ar, capaz de medir temperaturas de 0 a 50 Celsius e umidade na faixa de 20 a 90%. O sensor possui 4 pino, um apenas para a leitura tanto da temperatura e quanto da umidade. O componente também apresenta baixo consumo de potência e capacidade de transmissão de sinal de até 20 metros. Com isso, o DHT11 é comumente empregado junto à placa de desenvolvimento Arduino para aplicações como desumidificador, estações meteorológicas, regulador de umidade, etc.

##### 4.5.1. Especificações

- Modelo: DHT11
- Faixa de medição de umidade: 20 a 90% UR
- Faixa de medição de temperatura: 0° a 50°C
- Precisão de umidade:  $\pm 5\%RH$
- Precisão de temperatura:  $\pm 2^{\circ}C$
- Tensão de operação: 3-5VDC
- Corrente: 200uA a 500mA em medição e 100uA a 150 uA em espera.
- Tempo de espera: 1s
- Dimensões: 23 x 12 x 5mm

##### 4.5.2. Funcionamento

O sensor é constituído por um termistor do tipo NTC e o sensor de umidade HR202 conectados a um microcontrolador de 8 bits. O circuito interno faz a leitura dos sensores e transmite essa leitura ao microcontrolador. Os valores da temperatura e da umidade podem ser obtidos por uma única saída digital, que garante uma alta confiabilidade e estabilidade a longo prazo.

## 4.6. ESP8266

### 4.6.1. Introdução

O módulo ESP8266-01 é um conjunto SOC (System On Chip) de alta performance e alta integração sem fio, de baixo custo, projetado para plataformas móveis, ocupando pouco espaço e capaz de operar com restrições de energia.

O módulo possui memória ROM e SRAM on-chip, nos quais é armazenado e executado o firmware, que implementa TCP/IP e todo protocolo 802.11 b/g/n/e/i WLAN MAC e WiFi Direct. Podendo funcionar como provedor de conexão sem fio para outro controlador ou microprocessador, ou operando como aplicação através do seu MCU.

### 4.6.2. Especificações

- MCU integrado 32-bit de baixo consumo
- Conversor integrado 10-bit ADC
- Integrado com pilha de protocolo TCP/IP
- Tensão de operação: 3,3V
- Suporte à redes: 802.11 b/g/n
- Alcance: 90m aprox.
- Comunicação: Serial (TX/RX)
- Suporta comunicação TCP e UDP
- Conectores: GPIO, I2C, SPI, UART, Entrada ADC, Saída PWM e Sensor de Temperatura interno.
- Modo de segurança: OPEN/WEP/WPA\_PSK/WPA2\_PSK/WPA\_WPA2\_PSK
- Dimensões: 25 x 14 x 1mm
- Peso: 7g

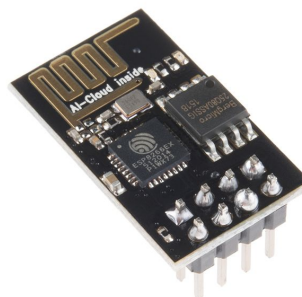


Figura 6: Módulo ESP8266-01.

#### 4.6.3. Tecnologia de baixo consumo

Projetado para aplicações móveis, vestíveis e internet das coisas, com o objetivo de alcançar o menor consumo, combina várias técnicas e propriedades. A arquitetura de economia de energia opera principalmente em 3 modos: active mode, sleep mode e deep sleep mode.

Paramêtros	Valor	Unidade
Tx802.11b, CCK 11Mbps, P OUT=+17dBm	170	mA
Tx 802.11g, OFDM 54Mbps, P OUT =+15dBm	140	mA
Tx 802.11n, MCS7, P OUT =+13dBm	120	mA
Rx 802.11b, 1024 bytes packet length , -80dBm	50	mA
Rx 802.11g, 1024 bytes packet length, -70dBm	56	mA
Rx 802.11n, 1024 bytes packet length, -65dBm	56	mA
Modem-Sleep	15	mA
Light-Sleep	0.9	mA
Deep-Sleep	10	uA
Power Off	0.5	uA

Figura 7: Consumo de acordo com datasheet.

#### 4.6.4. MCU

Embarcado com o microcontrolador Tensilica L106 32-bit e 16-bit RSIC, com clock speed de 80MHz, podendo alcançar o máximo de 160MHz. Com RTOS (Real Time Operation System) habilitado, e com apenas 20% do MIPS ocupados pela pilha WiFi, com possibilidade de utilizar o restante para programação e desenvolvimento.

#### 4.6.5. Programação

É possível embarcar o firmware no ESP8266 através da IDE do Arduino ou com programa de gravação de memória flash por comunicação serial. Para isto, inicializa-se o módulo com o GPIO0 em GND e faz upload do firmware. Com o firmware gravado o módulo passará executar e manterá o programa em memória não volátil, sendo necessário após desligar e ligar modulo manter o pino GPIO0 em alto, ou desconectado para manter o firmware anteriormente gravado.

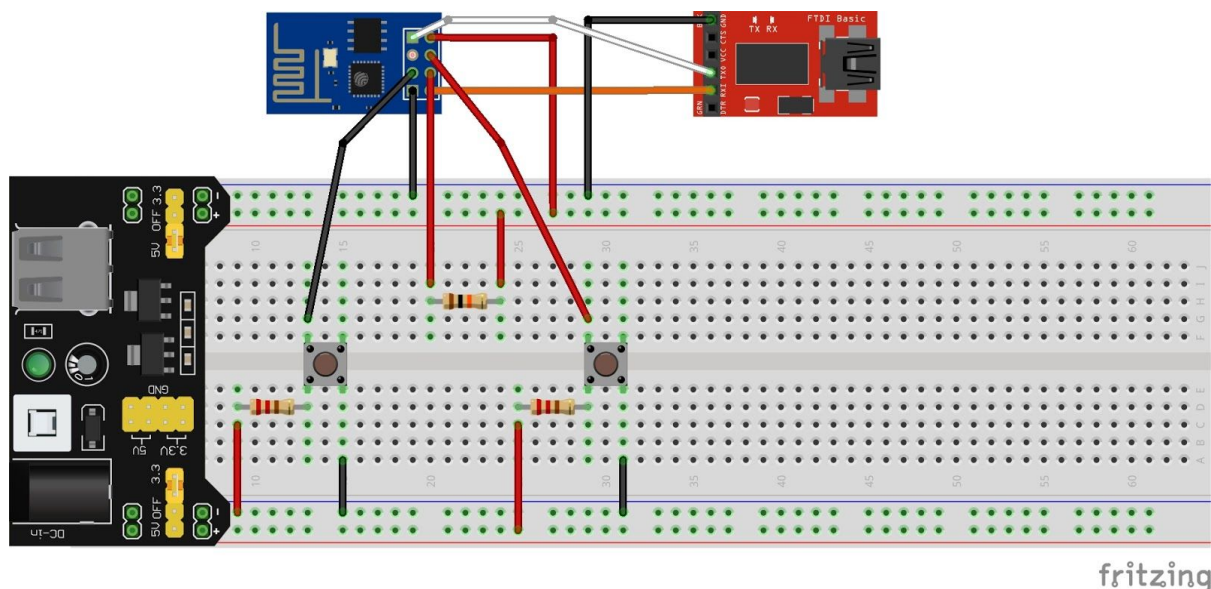
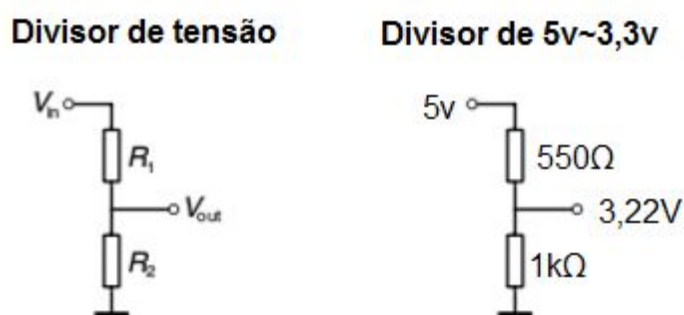


Figura 8: Ligação para programação com conversor serial.

#### 4.6.6. Funcionamento

Após gravação do firmware para utilizar o módulo em conjunto com arduino, é necessário utilizar uma fonte de alimentação externa, pois as portas do ATmega328 suportam no máximo 50mA e o ESP8266 pode consumir 200mA. Para comunicar com o módulo utiliza-se as portas RX (Read) e TX (Transmitte) através de comunicação serial, como o módulo funciona em 3.3v e os pinos digitais da plataforma Arduino trabalham a 5v, é possível fazer uma conversão através de um divisor de tensão.



A tensão de saída,  $V_{out}$ , é dada pela equação

$$V_{out} = \frac{R_2}{R_1 + R_2} \cdot V_{in}$$

Figura 9: Divisor de tensão de 5v para ~3,3v.

## 5. RESULTADOS

### 5.1. Sensoriamento

A seguir, é apresentado o script onde é possível verificar o funcionamento dos sensores que, de acordo com os valores coletados, é possível acionar o relé e consequentemente a válvula solenóide.

```
void loop()
{
  //Recebe o valor do pinoChuva
  valorChuva = analogRead(pinoChuva);

  //Recebe o valor do pinoUmidade
  valorUmidade = analogRead(pinoUmidade);

  //Imprime as informacoes no monitor serial
  mostraChuva(valorChuva);
  mostraUmidadeSolo(valorUmidade);

  if (valorChuva<600 || valorUmidade<800)
  {
    // Esta chovendo com intensidade moderada
    //OU o solo esta umido -> desliga o rele
    digitalWrite(pinoRele,HIGH);
  }
  else{
    //Caso contrario -> liga o rele
    digitalWrite(pinoRele,LOW);
  }

  // Repete a cada 10s
  delay(10000);
}
```

Figura 10: Código do sensores com o relé.

Como foi mencionado em fundamentos, os valores de saída analógica do sensor de chuva e umidade variam entre 0 e 1024, sendo que um valor de chuva coletado em até 600 é considerado como chuva moderada, e um valor de umidade coletado em até 800 é considerado como umidade moderada.

Dessa forma, esses valores são verificados e o relé é acionado, ou seja, ficará em nível alto caso os valores coletados estejam entre mencionados acima. Um nível alto para o relé significa que ele não acionará a válvula solenóide e ocasionalmente o jardim não será irrigado.



## **5.2. Comunicação**

### **5.2.1. Servidor**

O servidor foi implementado em java a partir do código disponibilizado por Andsel no GitHub, intitulado de Moquette.

Construindo para responder ao protocolo MQTT, o servidor funciona como Broker recebendo mensagens de Publish e Subscribe, e respondendo aos tópicos em que haja inscritos.

O servidor também é responsável pelo tratamento dos dados recebidos via mensagem, contendo os valores de leitura dos sensores. Com os dados recebidos, é feita uma análise e uma tomada de decisão sobre o ligamento ou desligamento da válvula. A decisão será publicado para o módulo de comunicação conectado ao controlador responsável pela manutenção da válvula.

### **5.2.2. Módulo cliente**

Para comunicação com o servidor foi utilizado o módulo ESP8266-01 conectados às placas de arduino, ficando a cargo do módulo a transmissão e recebimento de mensagens com o servidor. Após obter uma conexão com a rede local, a partir das bibliotecas padrão do ESP8266, é possível trocar mensagens com servidor com a utilização do protocolo de comunicação MQTT, implementado pela biblioteca PubSubClient de Knolleary, disponibilizado pelo GitHub.

Para transmitir mensagens da placa arduino para o ESP8266 é utilizado uma comunicação serial implementada a partir dos pinos digitais do arduino com a utilização da biblioteca SoftwareSerial, já disponibilizado por padrão pela IDE, que simula uma comunicação serial a partir dos pinos digitais, sendo possível criar uma comunicação serial diretamente com o ESP8266 que estará em laço esperando ler dados. Ao enviar os dados, o ESP8266 a partir da programação do firmware, fará uma leitura dos dados identificando o conteúdo e transmitirá por wifi para o servidor, o recebimento de dados pelo arduino se dará de forma inversa, ao receber dados o ESP8266 escreverá na serial de comunicação e arduino após identificar dados a serem lidos, receberá a mensagem.

Para transmitir dados ao servidor com protocolo MQTT é necessário indicar o tópico que identifica o conjunto do valor lido, no caso o sensor, e a mensagem a ser enviada que será o valor lido. Após a leitura via serial da mensagem com o ESP8266 e identificação da mensagem indicando o tópico e valor lido, é feito um Publish ao servidor.

Para receber mensagens do servidor, realiza-se um Subscribe indicando o tópico que se deseja receber mensagens. O módulo responsável por receber mensagens do servidor será o que estará ligado ao arduino da válvula, que a partir das mensagens publicadas no tópico da válvula de ‘Ligado’ ou ‘Desligado’, como o módulo estará inscrito, se dará um SelfPublish no servidor no tópico indicado e o módulo receberá a mensagem e enviará ao arduino para realizar a determinada ação.

### 5.3 Aplicação

O sistema usa o aplicativo IoT MQTT Dashboard desenvolvido em Android pela Nghia TH. O aplicativo tem como objetivo controlar projetos de Internet das coisas usando o protocolo MQTT.

Características do aplicativo:

- Permite múltiplas ligações entre dispositivos;
- Conexão feita por SSL;
- Permite que o administrador/usuário especifique a unidade dos dados;
- Apresenta valores numéricos em um gráfico em tempo real;
- Em sua interface, há disponível vários componentes customizáveis como texto, botão, interruptor, barra de procura, caixa de combinação e seletor de cores.

A seguir é mostrado a visão inicial e final do aplicativo após toda configuração, bem como a descrição de cada imagem:

Na Figura 11 é mostrada a visão inicial do aplicativo para configuração com o servidor, onde é solicitado o IP do servidor, a porta, usuário e senha. A seguir, a Figura 12 representa a visão do usuário inscrito em quatro tópicos: temperatura, umidade, chuva e vazão. E finalmente, na Figura 13 é mostrado três botões, inscritos no tópico ‘solenóide’. Os botões ‘Ligar’, ‘Desligar’ são responsáveis por ligar e desligar a válvula solenóide, ou seja, ao clicar nestes botões a torneira é acionada. Quanto ao botão ‘Desativar’, ao ser clicado o usuário desativa o sistema, desligando-o.

Connection CREATE

Client ID

Server

Port

Username

Password

SSL ☐

Key store file

Select .BKS file... CLEAR

Key store password

Figura 11: Visão inicial de conexão do aplicativo.

MQTT Dashboard

Connected to 192.168.25.8

SUBSCRIBE PUBLISH

Temperatura 31.0 now

Umidade Umidade baixa now

Chuva Sem chuva now

Vazao 0.0 now

Figura 12: Visão Subscriber. Os dados do sistema são visualizados.

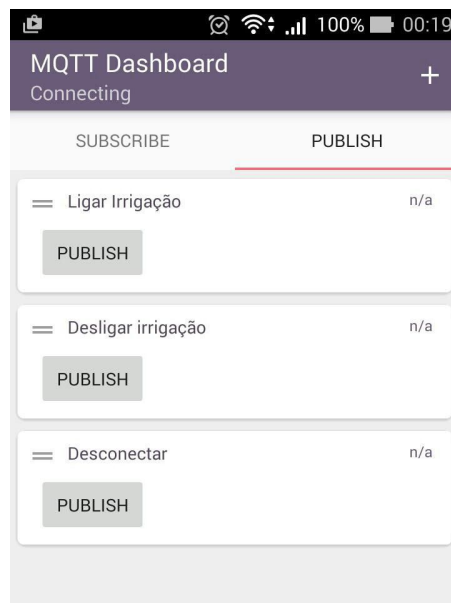


Figura 13: Visão Publisher. O usuário como atuador.

## 6. BIBLIOGRAFIA

GITHUB. **Compartilhamento e repositório de códigos**. Conteúdo disponível em:

<<https://github.com/knolleary/pubsubclient>>

<<https://github.com/andsel/moquette>>

Acessado em: 17 de novembro, 2016.

BLOG FILIPEFLOP. **Tutorial módulo wireless ESP8266 com arduino**. Conteúdo disponível em:

<<http://blog.filipeflop.com/wireless/esp8266-arduino-tutorial.html>>

<<http://blog.filipeflop.com/wireless/upgrade-de-firmware-do-modulo-esp8266.html>>

Acesso em: 12 de outubro, 2016.

EMBARCADOS. **ESP8266 com Arduino**. Conteúdo disponível em:

<<http://www.embarcados.com.br/esp8266-com-arduino/>>. Acesso em: 19 de outubro, 2016.

ESPRESSIF SYSTEMS. **Datasheet ESP8266EX v4.3**. Conteúdo disponível em:

<<http://download.arduino.org/products/UNOWIFI/0A-ESP8266-Datasheet-EN-v4.3.pdf>>.

Acesso em: 25 de outubro, 2016.

BLOG FILIPEFLOP. **Sensor de chuva YL-83**. Conteúdo disponível em:

<<http://blog.filipeflop.com/sensores/sensor-de-chuva-yl-83.html>>. Acesso em: 16 de outubro, 2016.

NEWTON C. BRAGA. **Tudo Sobre Relés**. Conteúdo disponível em:

<<http://www.newtoncbraga.com.br/index.php/como-funciona/597-como-funcionam-os-reles?showall=1&limitstart=>>>. Acesso em: 24 de outubro, 2016.

ARDUINOBR. **Usando um relé para controlar dispositivos elétricos.** Conteúdo disponível em:

<<http://www.arduino.br/arduino/acionadores/usando-um-rele-para-controlar-dispositivos-eletricos/>>. Acesso em: 24 de outubro, 2016.

BLOG FILIPEFLOP. **Sensor de umidade de solo Higrômetro.** Conteúdo disponível em:

<<http://www.filipeflop.com/pd-aa99a-sensor-de-umidade-do-solo-higrometro.html>>. Acesso em: 16 de outubro, 2016.

BLOG FILIPEFLOP. **Monitoramento de planta usando Arduino.** Conteúdo disponível em:

<<http://blog.filipeflop.com/sensores/monitore-sua-planta-usando-arduino.html>>. Acesso em: 24 de setembro, 2016.