

INSTITUTO FEDERAL DO PARANÁ – CAMPUS COLOMBO
ANDERSON DOS ANJOS DE OLIVEIRA

ACI: A CASA INTELIGENTE

COLOMBO
2021

ANDERSON DOS ANJOS DE OLIVEIRA

ACI: A CASA INTELIGENTE

Trabalho de Conclusão de Curso apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal do Paraná, Campus Colombo, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Marcos Dinís Lavarda

COLOMBO

2021

TERMO DE APROVAÇÃO

ANDERSON DOS ANJOS DE OLIVEIRA

ACI: A CASA INTELIGENTE

Trabalho de Conclusão de Curso aprovado como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, pelo Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, do Instituto Federal do Paraná, Campus Colombo, pela seguinte banca examinadora:



Prof. Me. Marcos Dinís Lavarda

Orientador



Prof. Dr. Eduardo Liquio Takao



Prof. Me. Ademir Luiz do Prado

Colombo, 22 de outubro de 2021.

RESUMO

Com o aumento da demanda nas tarefas domésticas, tanto no meio urbano, quanto no rural, os brasileiros despendem de tempo diário para realizar atividades corriqueiras no qual poderiam destinar à atividades de bem-estar. É nesse cenário que foi identificada uma oportunidade para elaboração do presente projeto de domótica, com o objetivo de ajudá-los com as tarefas de vida diária repetitivas. Através de um sistema embarcado ESP32 com Sistema Operacional de Tempo Real FreeRTOS, placas de sensores e atuadores auxiliares e interface com o usuário, é possível acionar e monitorar equipamentos eletroeletrônicos de forma simples e remota. Isso pode proporcionar economia de tempo e/ou recursos aos usuários como, por exemplo, o monitoramento do nível de água em uma cisterna e a automação da rega de plantas de uma horta. Utilizando um aplicativo Android genérico para comunicação *Bluetooth*, foi possível controlar e monitorar o sistema embarcado. O sistema desenvolvido utilizou como componentes o ESP32 e como periféricos os módulos de relés para acionamento de luzes e bomba d'água de irrigação, além de sensor de umidade do solo, sensor de chuva e sensor de nível da cisterna, sendo programados em linguagem C na IDE do Arduino. O projeto tem a finalidade de propiciar estudos de aplicações de baixo custo para esta finalidade de automatização e emprego de tecnologia com o uso de Sistema Operacional de Tempo Real.

Palavras-Chave: Domótica, Sistema Embarcado em Tempo Real, ESP32.

ABSTRACT

With the increase in the demand for domestic tasks, both in urban and rural areas, Brazilians spend daily time to carry out everyday activities which they could allocate to well-being activities. It is in this scenario that an opportunity was identified to make the present home automation project, with the aim of helping them with repetitive daily life tasks. Through an embedded system ESP32 with FreeRTOS Real Time Operating System, sensor boards and auxiliary actuators and user interface, it is possible to activate and monitor electronic equipment in a simple and remote way. This can save users time and/or resources with, for example, monitoring the water level in a cistern and automating the plant region of a vegetable garden. Using a generic Android application for Bluetooth communication, it was possible to control and monitor the embedded system. The system was developed using ESP32 as components and as peripherals the relay modules to activate lights and irrigation water pump, in addition to soil moisture sensor, rain sensor and cistern level sensor, being programmed in C language in the Arduino IDE. The project has a pattern of providing low cost application studies for this high cost of automation and technology use with the use of Real Time Operating System.

Keywords: Home Automation, Real Time Embedded System, ESP32.

SUMÁRIO

1	INTRODUÇÃO	6
1.1	JUSTIFICATIVA	6
2	OBJETIVO	7
2.1	OBJETIVO GERAL	7
2.2	OBJETIVOS ESPECÍFICOS.....	7
3	METODOLOGIA	8
4	MATERIAIS E MÉTODOS	9
4.1	Sistema Embarcado ESP32	9
4.2	Módulo de relés	9
4.3	Fonte de 5 Volts	10
4.4	Sensor de Umidade do solo.....	10
4.5	Sensor Ultrassom	10
4.6	Programação feita em linguagem C.....	11
4.7	Arduino IDE (Ambiente de desenvolvimento)	11
4.8	ASTAH	11
4.9	Sistema FreeRTOS.....	11
5	LEVANTAMENTO DE REQUISITOS	13
5.1	REQUISITOS FUNCIONAIS.....	13
5.2	REQUISITOS NÃO FUNCIONAIS	13
5.3	DIAGRAMA DA UML.....	13
5.3.1	DIAGRAMA DE CASOS DE USO.....	14
5.3.2	DIAGRAMA DE CLASSE	14
5.3.3	DIAGRAMA DE CONEXÕES ELÉTRICAS.....	16
6	PROTÓTIPOS	Erro! Indicador não definido.
6.1	APP SERIAL BLUETOOTH TERMINAL	19
6.2	CÓDIGO DO PROJETO	20
7	CONCLUSÃO	21

1 INTRODUÇÃO

Sabemos que, com o avanço da tecnologia, tarefas simples, tanto residenciais quanto em escritórios, chácaras, fazendas, entre outros, podem ser automatizadas. Porém, os custos de implementação, bem como a dificuldade de encontrar estes dispositivos no mercado brasileiro podem determinar a viabilidade de instalação desses sistemas de automação.

Com isso, a ideia principal do projeto é apresentar um produto que atenda algumas das principais necessidades da automação residencial, utilizando componentes eletrônicos relativamente comuns no mercado nacional e ainda auxiliar na redução de custo nas contas de água e energia elétrica com o melhor gerenciamento dos dispositivos em funcionamento.

O Objetivo deste projeto é desenvolver uma solução visando o emprego de sistema embarcado com Sistema Operacional de Tempo Real FreeRTOS, para possibilitar acender lâmpadas ou ligar equipamentos e monitorar a situação do sistema utilizando smartphone via comunicação Bluetooth.

A metodologia adotada inicialmente foi realizar o levantamento de dados a partir de uma pesquisa com possíveis usuários, através de formulário no google forms.

Na sequência foi elaborado o projeto eletrônico e mecânico, a aquisição dos componentes e a construção do protótipo, em seguida, foi programado o sistema operacional do Sistema Embarcado no ESP32, efetuada sua integração com a eletrônica e a mecânica e, por fim, realizados os ajustes e testes do protótipo funcional em pequena escala.

1.1 JUSTIFICATIVA

A tecnologia avança rapidamente a cada ano e, embora determinada parte da população mundial acompanhe as tendências tecnológicas, a distribuição e alcance dos dispositivos de novas tecnologias não alcançam toda a população.

Com o intuito de facilitar a aquisição de tais produtos, que melhoram a vida das pessoas, a ACI – A Casa Inteligente, terá a finalidade de apresentar o desenvolvimento de um sistema focado inicialmente em alguns problemas mais relevantes para a população de baixa e média renda. Tal sistema será empregado no gerenciamento de consumo de energia elétrica e água, com a automatização e controle remoto de lâmpadas e/ou dispositivos eletrônicos e irrigação por uso de água de cisterna.

2 OBJETIVO

Neste capítulo serão apresentados os objetivos do projeto desenvolvido.

2.1 OBJETIVO GERAL

Desenvolver um sistema embarcado de automação que monitore e controle o uso de água e energia elétrica indiretamente utilizando sistema operacional de tempo real.

2.2 OBJETIVOS ESPECÍFICOS

- Listar e adquirir os componentes e materiais necessários para o projeto;
- Elaborar o projeto eletrônico, mecânico e de programação;
- Desenvolver o protótipo eletrônico, mecânico e código do sistema;
- Integrar a programação e a eletrônica à maquete; e
- Testar as funcionalidades implementadas no sistema.

3 METODOLOGIA

Com o intuito de obter informações sobre a necessidade de possíveis usuários, foi feita uma pesquisa exploratória online através de Google Forms apresentado na Figura 1, para então direcionar o projeto aos interesses dos pesquisados.



ACI - A Casa inteligente

Reserve 3 minutos do seu tempo para responder estas perguntas, desde já agradeço.

Sou aluno do Curso Superior de Tecnologia em ADS ofertado pelo campus Colombo do IFPR. Gostaria de conduzir esta entrevista, enviando este questionário para fins acadêmicos unicamente. Estou desenvolvendo um trabalho sobre Sistema de Casa Inteligente acessível à todos e conto com seu apoio para levantar alguns dados. Me comprometo a manter em sigilo a sua identidade. Agradeço muito seu apoio.

O meu trabalho consiste em tornar uma casa inteligente tecnologicamente acessível a muitos, a fim de facilitar a vida das pessoas.

Utilizando dispositivos para, por exemplo, acender ou regular a intensidade da luz de um cômodo através de aplicativo de smartphone. Assim como monitorar a temperatura interna ou saber se existe um vazamento de gás. Regar suas plantas em horários programados, receber mensagem, e-mail ou avisos sonoros indicando intruso na sua residência, verificar se a porta da casa está trancada. Estas e outras inúmeras funcionalidades podem fazer parte da Casa Inteligente.

Figura 1 – Pesquisa Exploratória Realizada Através do Google Forms.

Fonte: Autor (2021).

Na Figura 2 podemos observar que cerca de 89 % dos entrevistados acreditam que A Casa Inteligente pode trazer alguma melhoria na vida das pessoas.

Na sua opinião, o sistema de casa inteligente pode melhorar a vida das pessoas?

47 respostas

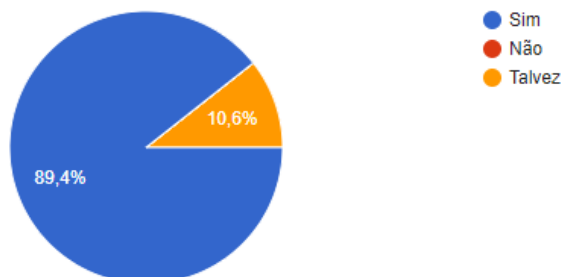


Figura 2 – Gráfico Sobre Pesquisa Exploratória Realizada Pelo Google Forms

Fonte: Autor (2021).

4 MATERIAIS E MÉTODOS

Para o desenvolvimento do projeto foram utilizadas tecnologias, as quais estão descritas neste capítulo.

4.1 Sistema Embarcado ESP32

DOIT ESP32 – ESP-WROOM-32 é uma placa de desenvolvimento, feita pela empresa DOIT (*Doctors of Intelligence & Technology*), muito similar a placa NodeMCU, mas tendo como base o módulo ESP32 ao invés do módulo ESP-12E.

O kit de desenvolvimento de sistema embarcado ESP32 é um módulo baseado no Chip ESP32-D0WDQ6, evolução do ESP8266, feito pela empresa Espressif, que incorpora a utilização de WiFi com Bluetooth BLE (*Bluetooth Low Energy* - Baixíssimo Consumo de Energia) e o Microprocessador.

Este kit atende de forma genérica várias finalidades de projetos simples, como uma rede de sensores de baixa potência, até aplicações complexas, como codificação de voz e transmissão de música (CURTO-CIRCUITO, 2021-c).

4.2 Módulo de relés

O Módulo Relé é um módulo utilizado para facilitar o acionamento de cargas através de um microcontrolador, necessitando apenas conectar Jumpers no Relé e em um sistema de acionamento digital, como um microcontrolador por exemplo, dispensando a necessidade de montar placas ou circuitos de acionamento, tornando o projeto mais prático e organizado.

Os relés podem ser dois tipos: Acionamento Mecânico ou Acionamento Eletrônico (Estado Sólido), sendo utilizado neste projeto o acionamento mecânico.

O Módulo Relé mecânico trabalha com uma chave em três posições: Normal Aberto (NA), Normal Fechado (NF) e Comum (COM). O contato ocorre entre o terminal COM e o NF, com uma tensão acionamento digital nos terminais de comando, o contato passa a ser entre o terminais COM e NA.

Embora a função do Módulo Relé de Estado Sólido seja a mesma, a diferença ocorre na tecnologia empregada no acionamento, não possuindo elementos mecânicos no chaveamento, funcionando a partir de tiristores ao invés de contatos, operando apenas com as posições Normal Aberto e Normal Fechado. Como vantagem, o relé de estado sólido não sofre desgaste mecânico (CURTO-CIRCUITO, 2021-b).

4.3 Fonte de 5 Volts

O Conversor AC/DC (Corrente alternada para corrente contínua) utilizado no projeto opera de forma bivolt (90~240VAC) na entrada, convertendo essa entrada em tensão em corrente contínua (5VDC / 3A). Essa fonte de alimentação pode ser utilizada em diversos aparelhos eletroeletrônicos e projetos eletrônicos que necessitem de fontes com tensão de 5V e corrente de saída nominal de 2,2A (máximo 3A). A fonte utilizada possui ajuste automático à tensão e corrente de entrada, 127V ou 220V, e fornece corrente real de 2,2A com picos de até 3A (CURTO CIRCUITO, 2021-a).

4.4 Sensor de Umidade do Solo

O sensor de umidade do solo é composto por duas estruturas, uma sonda e o módulo de aquisição. Enquanto a sonda fica no solo o módulo de aquisição utiliza um chip comparador LM393 para enviar a condutância do solo, determinada pela presença de umidade.

Como saída, o módulo conta com os pinos DO e o AO. O pino DO é digital, com nível 0 ou 1 dependendo do ajuste de sensibilidade feito no trimpot. O pino AO é analógico, onde sua saída flutua entre 0V e 5V, de acordo com a umidade, o que possibilita monitorar com maior precisão usando uma porta analógica do microcontrolador (FILIPEFLOP, 2021-a).

4.5 Sensor Ultrassom

O Sensor Ultrassônico HC-SR04 empregado neste projeto servirá para verificar a distância do topo da caixa d'água e a superfície de água, sabendo-se se a mesma encontra-se cheia ou vazia.

Para isso, o sensor ultrassônico capaz de realizar leituras de distâncias entre 2 cm e 4 metros, com precisão de 3 mm é acionado pela emissão de um som em alta frequência, calculando a distância do obstáculo por meio do tempo entre a emissão e recebimento do eco. Este dispositivo é principalmente utilizado para medir distâncias entre o sensor e objetos, porém pode servir para outras finalidades, como para um robô desviar de obstáculos, acionar alarmes, etc. (FILIPEFLOP, 2021-b).

4.6 Programação em linguagem C

A linguagem "C" foi desenvolvida por *Brian W. Kernighan* e *Dennis Ritchie* em 1972, tendo sido baseada na linguagem BCPL (*Basic Combined Programming Language*), desenvolvida por Martin Richards. Contudo, a influência do BCPL foi dada por meio da linguagem "B" que foi desenvolvida por Ken Thompson em 1970, a qual foi utilizada no PDP-11, primeiro sistema operacional UNIX (OSÓRIO, F. S., 1992).

Com o passar do tempo a linguagem "C" foi modificada e serviu para reescrever o sistema operacional UNIX (1973), originalmente desenvolvido em linguagem *Assembler* para o PDP-11, passando a ser até hoje associada ao sistema operacional UNIX, porém, a linguagem "C" não limita suas aplicações a um sistema operacional ou dispositivo (OSÓRIO, F. S., 1992).

4.7 Arduino IDE (Ambiente de desenvolvimento)

O software Arduino é um ambiente de desenvolvimento e programação de sistemas embarcados integrado. Em sua maioria, o Arduino apresenta diversos usos com ferramentas de código aberto. Pode-se também utilizar várias bibliotecas e integrar outras linguagens de programação, além do "C" neste ambiente, possibilitando adicionar códigos AVR-C diretamente na IDE do Arduino, que irá compilar e carregar o código corretamente de acordo com o microcontrolador configurado (ARDUINO, 2018).

4.8 ASTAH

Para realizar a modelagem dos diagramas do projeto foi utilizada a ferramenta Astah. Utilizada para desenvolver UML, diagramas ER, diagramas de fluxo de dados, fluxogramas, mapas mentais, entre outros (ASTAH, 2021).

4.9 FRITZING

Para realizar o planejamento prévio e a ilustração das conexões elétricas foi utilizado o *software* Fritzing (FRITZING, 2021).

4.10 Sistema FreeRTOS

O *FreeRTOS* é um sistema operacional de tempo real de código aberto, amplamente utilizado e empregado em microcontroladores. Sua finalidade é permitir a programação de tarefas que são gerenciadas e processadas de forma escalonada por um processador e, até mesmo, de forma paralela em dois processadores, como é o caso do ESP32, que conta com dois núcleos de processamento.

O FreeRTOS é gratuito e de código aberto. Este sistema operacional possui diversas bibliotecas e possibilita várias aplicações de baixo consumo (*Low Energy*) e de internet das coisas (*IoT – Internet of Things*). Seu uso é indicado devido a sua confiabilidade e facilidade de uso, além de oferecer previsibilidade para os novos lançamentos (AMAZON, 2021).

5 LEVANTAMENTO DE REQUISITOS

Neste capítulo serão apresentados os requisitos que o *software* desenvolvido contém em sua estrutura, planejamento e realização de tarefas, sendo apresentadas as condições necessárias para satisfazer os objetivos do projeto.

Como requisitos temos as funções, objetivos, propriedades e restrições que o sistema possui, além dos padrões e especificações relacionadas aos usuários e sua interface. Portanto, “um requisito é um aspecto que o sistema proposto deve fazer ou uma restrição no desenvolvimento do sistema” (DEVIMEDIA, 2013), ocorrendo de forma integrada entre projetista e usuários.

Os requisitos do presente projeto foram divididos em requisitos funcionais e requisitos não funcionais, descritos a seguir:

5.1 REQUISITOS FUNCIONAIS

- RF 01 – O sistema ACI será instalado em ambiente seco, livre de umidade;
- RF 02 – Não requer manutenção constante; e
- RF 03 – Emitir Relatório de Atividades.

5.2 REQUISITOS NÃO FUNCIONAIS

- RNF 01 – ESP32 funciona com conexão WIFI ou Bluetooth para comunicação;
- RNF 02 – Requer o uso de energia elétrica local 127/220 Volts, para ligar a(s) fontes(s); e
- RNF 03 – O cliente não necessitará de acesso à internet.

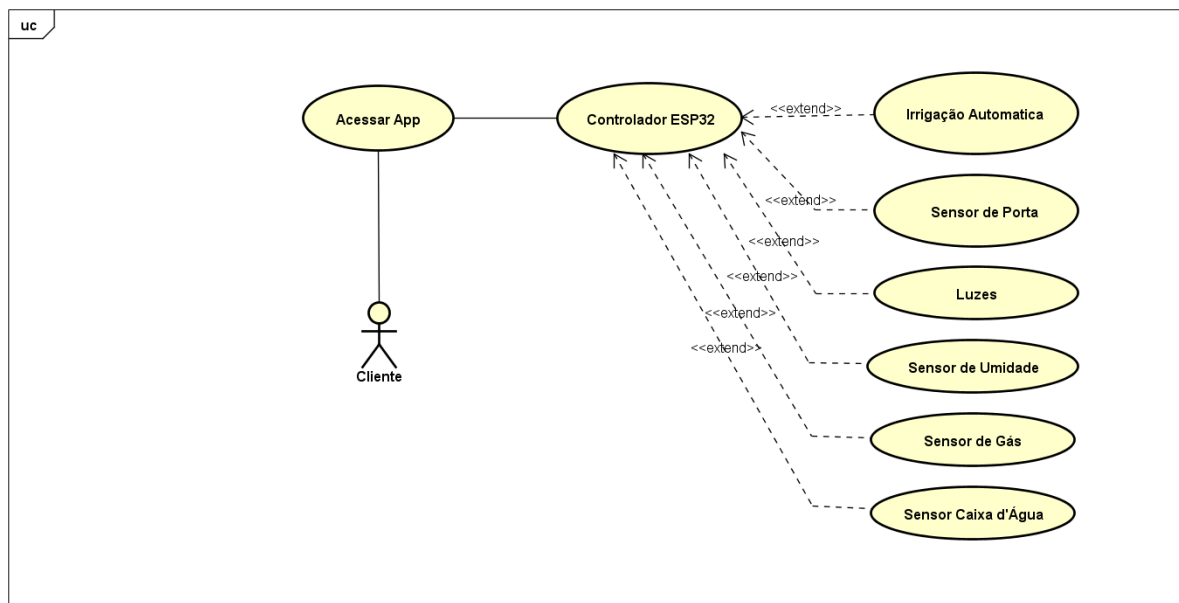
5.3 DIAGRAMAS DE MODELAGEM DE SOFTWARE E HARDWARE

Entre os diagramas existentes para o planejamento da programação está o UML (*Unified Modeling Language*), amplamente empregado na modelagem de projetos de programação, além de auxiliar na documentação dos sistemas orientados a objetos (DEVIMEDIA, 2012). Neste projeto foram utilizados diagramas similares ao UML, como o diagrama de casos de uso e o de classes, apresentados a seguir:

5.3.1 DIAGRAMA DE CASOS DE USO

Partindo do ponto de vista do usuário, o diagrama de casos de uso serve para documentar as atividades que o sistema realiza, suas funcionalidades e a interface do sistema com o usuário, não sendo aprofundados detalhes técnicos do mesmo (DEVMEDIA, 2012).

Figura 3 exibe os Casos de Uso do Projeto. Nela o usuário acessa o aplicativo por meio do seu *Smartphone*, monitorando e interagindo com o sistema que integra o controle de luzes, de irrigação e nível de água da caixa ou cisterna, além de possibilitar outras integrações previstas com sensores de alarme de invasão e vazamento de gás.

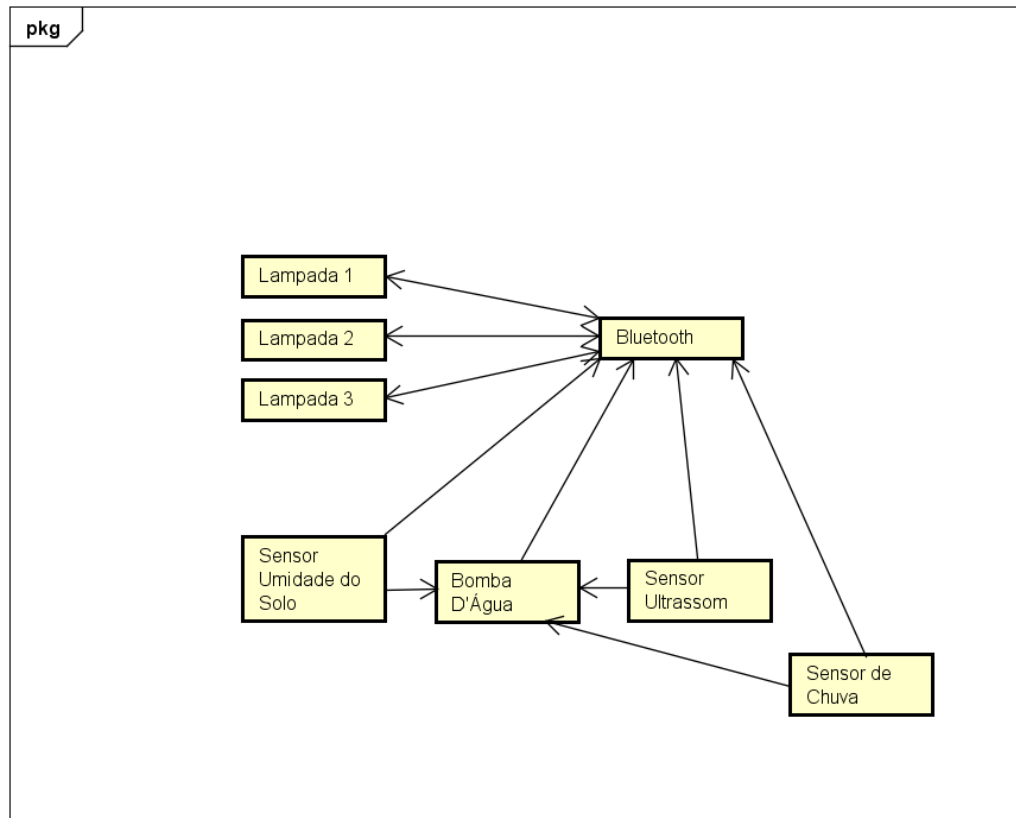


powered by Astah

Figura 3 – Casos de Uso do Protótipo
Fonte: Autor (2021).

5.3.2 DIAGRAMA DE CLASSE

Na Figura 4 é apresentado um Diagrama de Classe onde mostra de uma forma simples a relação entre as tarefas incluídas no projeto através do sistema operacional FreeRTOS.



powered by Astah

Figura 4 – Diagrama de Classe

Fonte: Autor (2021).

Dessa forma, o ESP32 realiza a interface com o usuário por meio da tarefa Bluetooth, a qual troca mensagens entre as demais tarefas. Todas as tarefas informam o usuário a respeito do seu estado atual: se as lâmpadas que estão acesas, se está chovendo, qual o nível da caixa d'água e o estado atual e de operação da horta (umidade do solo e acionamento da bomba d'água).

A tarefa Bomba D'água controla o acionamento da irrigação, esta tarefa depende das condições das tarefas Sensor de Chuva, Sensor Ultrassom e Umidade do Solo. Caso esteja chovendo, o solo esteja úmido ou o nível de água esteja baixo na caixa, a bomba d'água não é acionada.

Todas as tarefas são executadas e gerenciadas pelos processadores do ESP32, o qual realiza escalonamento entre elas, sendo possível incluir novas tarefas para integrar novos sistemas no projeto futuramente.

5.3.3 DIAGRAMA DE CONEXÕES ELÉTRICAS

Na Figura 5 é apresentado um diagrama de conexões elétricas, nele é possível verificar a presença de três sensores, um ultrassônico para identificar o volume restante de água na cisterna, um sensor analógico de precipitação de chuva e um sensor digital de umidade do solo para horta.

Também são ligados pelo módulo de relés de 4 canais, uma bomba d'água para irrigar a horta e três lâmpadas de led para iluminação da residência. Todos controlados pelo ESP32 por meio de conexão *Bluetooth* integrada ao sistema embarcado. A alimentação é realizada por meio de terminal P4 fêmea conectada a uma fonte AC/DC, onde a entrada de corrente alternada (tomada) é entre 90 e 240 VAC, saída de corrente contínua (conector P4 macho) é de 5 VDC, amperagem máxima é de 3A e potência da fonte é de 15 Watts.

Embora não esteja presente no diagrama da Figura 5, recomenda-se o uso de conversores lógicos ou divisores resistivos nas entradas dos sensores, a fim de ajustar a entrada máxima no ESP32 para 3,3V, típico desse modelo de controlador.

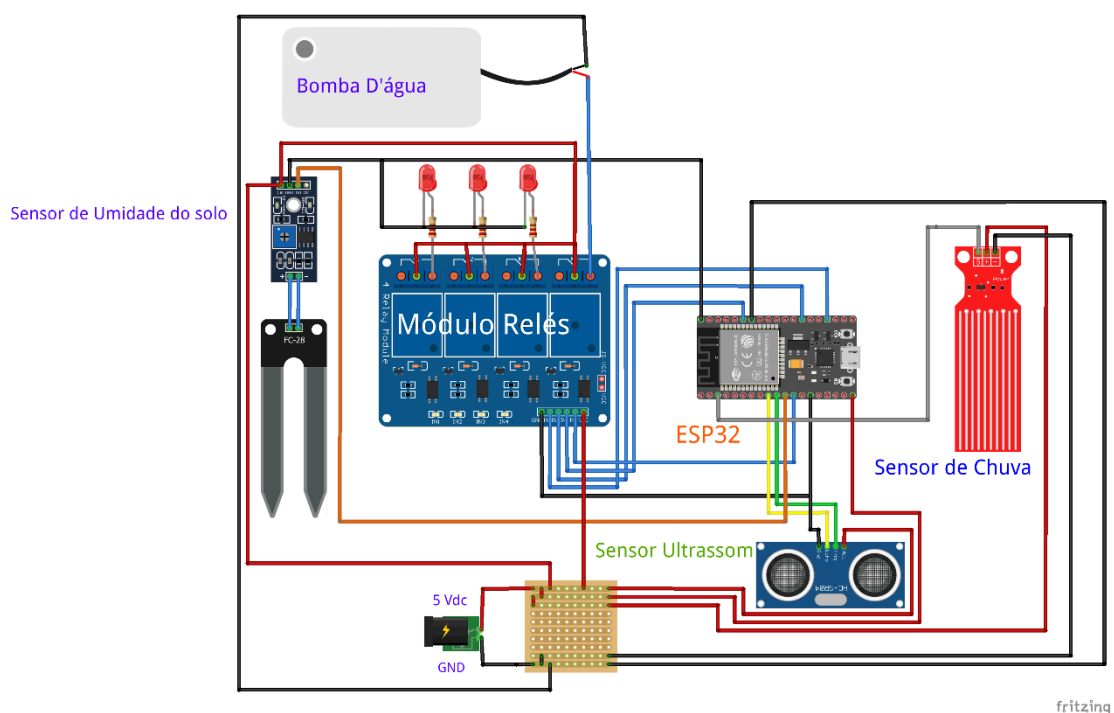


Figura 5 – Diagrama de Ligações Elétricas.

Fonte: Autor (2021).

6 RESULTADOS DO PROJETO

A Figura 6 exibe uma maquete desenvolvida para acondicionar e embarcar o sistema desenvolvido em uma casa em menor escala.



Figura 6 – Protótipo ACI – A Casa Inteligente.

Fonte: Autor (2021).

A Figura 7 exibe um ESP32, dispositivo de sistema embarcado, que foi programado em linguagem C utilizando sistema operacional FreeRTOS que divide tarefas e faz a gestão em tarefas distintas.

O código em FreeRTOS foi desenvolvido de forma concorrente, onde utiliza escalonamento entre as várias tarefas e cada tarefa efetua uma determinada ação específica a respeito do que se espera que ESP32 atue, tais como: Bluetooth, irrigação, acendimento de lâmpadas.



Figura 7 – ESP32

Fonte: Autor (2021).

A Figura 8 exibe módulo de relés usado para acionar a bomba d'água e acender as lâmpadas dos cômodos.



Figura 8 - Módulo de Relés

Fonte: Autor (2021).

A Figura 9 exibe sensor usado para identificar umidade do solo.



Figura 9 – Sensor de Umidade do Solo

Fonte: Autor (2021).

A Figura 10 exibe o sensor ultrassônico usado para identificar o nível da caixa d'água, onde é utilizado a distância entre o sensor que fica no topo da caixa d'água até a superfície de água e com isso consegue identificar o nível por meio de um sensor não invasivo e de fácil instalação.

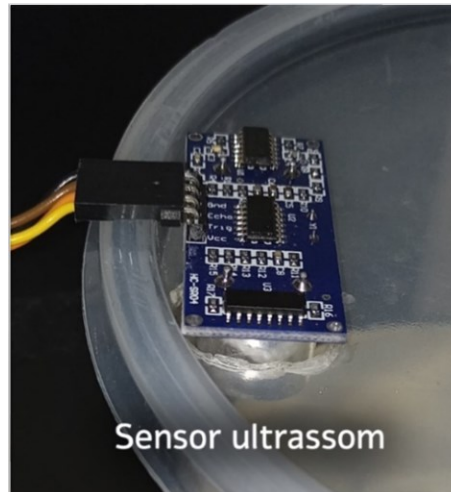


Figura 10 – Sensor Ultrassom

Fonte: Autor (2021).

6.1 APP SERIAL BLUETOOTH TERMINAL

O Terminal Serial Bluetooth é um aplicativo disponível para Android na Google Play. A partir dele é possível enviar e receber comandos por meio de um terminal/console para o ESP32 utilizando o *Bluetooth* do *Smartphone*.

Sistemas microcontrolados que tenham uma interface serial / UART conectada com um conversor Bluetooth podem se comunicar com o Terminal Serial Bluetooth. (GOOGLE, 2021). A Figura 11 apresenta a interface entre o sistema e *Smartphone* do usuário.

```
11:04:49.506 Lâmpada andar 3 acesa
11:04:50.311 Lâmpada andar 3 apagada
11:04:52.845 Lâmpada andar 2 acesa
11:04:53.656 Lâmpada andar 2 apagada
11:04:55.814 Lâmpada andar 1 acesa
11:04:56.648 Lâmpada andar 1 apagada
11:05:04.535 Está chovendo!!
11:05:04.537 Irrigação desligada
11:05:04.537 Solo está Úmido
11:05:04.586 CAIXA PELA METADE !!!
11:05:24.830 Não está chovendo !!!
11:05:24.833 Solo está Úmido
11:05:24.881 CAIXA PELA METADE !!!
11:05:44.631 Não está chovendo !!!
11:05:44.633 Solo está Úmido
11:05:44.681 CAIXA PELA METADE !!!
11:06:04.431 Não está chovendo !!!
11:06:04.433 Solo está Úmido
11:06:04.482 CAIXA PELA METADE !!!
11:06:24.726 Não está chovendo !!!
11:06:24.729 Solo está Úmido
11:06:24.776 CAIXA PELA METADE !!!
11:06:44.526 Não está chovendo !!!
11:06:44.528 Solo está Úmido
11:06:44.577 CAIXA PELA METADE !!!
11:07:04.820 Não está chovendo !!!
11:07:04.821 Solo está Úmido
11:07:04.870 CAIXA PELA METADE !!!
11:07:24.621 Não está chovendo !!!
11:07:24.623 Solo está Úmido
11:07:24.671 CAIXA PELA METADE !!!
11:07:40.902 Disconnected
```

Figura 11 - Serial Bluetooth Terminal

Fonte: Autor (2021).

6.2 CÓDIGO DO PROJETO

As Figuras 12 e 13 representam trechos do código elaborado em linguagem C para atender o projeto. Foram criadas tarefas utilizando FreeRTOS, onde as tarefas são geridas distintamente. Na Figura 12, é possível visualizar a tarefa “Caixa” que identifica o volume de água presente na caixa d’água por meio da leitura do sensor ultrassônico.

```
BluetoothSerial SerialBT;
byte BData;

/* Check if Bluetooth configurations are enabled in the SDK */
#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run 'make menuconfig' to and enable it
#endif

void Caixa (void * Parametro)
{
  for(;;){
    digitalWrite(trigger1, HIGH);
    delayMicroseconds(10); // Added this line
    digitalWrite(trigger1, LOW);
    distancia1 = pulseIn(echo1, HIGH, 50000);
    distancia1 = (distancia1/2) / 29.1;
    Serial.print("Distancia: ");
    Serial.print(distancia1);
    Serial.print("cm - ");

    if(distancia1 > 14){
      Serial.println("CAIXA VAZIA !!!");
      vTaskDelay(2000 / portTICK_PERIOD_MS);
    }

    else if(distancia1 <= 13 && distancia1 > 7){
      Serial.println("CAIXA QUASE VAZIA !!!");
      vTaskDelay(2000 / portTICK_PERIOD_MS);
    }
  }
}
```

Figura 12 – Trecho do Código do Projeto

Fonte: Autor (2021)

Na Figura 13 é possível ver a declaração das tarefas “Umidade” e “Caixa”.

```
xTaskCreate
(
  Umidade,      /*Tarefa ou funcao */
  "Umidade",   /*Nome da tarefa ou funcao */
  1000,        /*Tamanho da Pilha */
  NULL,        /*Parametro da Entrada */
  0,           /*Prioridade da */
  NULL         /*Identificador da */
);

xTaskCreate
(
  Caixa,       /*Tarefa ou funcao */
  "Caixa",    /*Nome da tarefa ou funcao */
  10000,      /*Tamanho da Pilha */
  NULL,       /*Parametro da Entrada */
  0,          /*Prioridade da */
  NULL        /*Identificador da */
);
```

Figura 13 – Trecho do Código do Projeto

Fonte: Autor (2021)

7 CONCLUSÃO

Este projeto possibilita apresentar proposta de aplicação em automação residencial, comercial e até rural para pequenas propriedades, a fim de otimizar o tempo com as tarefas diárias.

Em pequenas plantações ou hortas residenciais ou comunitárias, a automatização do processo de irrigação pode auxiliar no manejo do plantio monitorando e controlando os parâmetros e informações a respeito da umidade e irrigação do solo (seco ou úmido).

Ainda, é possível ampliar as aplicações apresentadas neste projeto, com a inclusão de novas tarefas, visto que o sistema FreeRTOS permite que novas tarefas sejam adicionadas sem interferir no funcionamento das demais, o que é muito importante em dispositivos escaláveis.

Como trabalhos futuros espera-se integrar o sensor de gás e o alarme de porta, além de enviar as informações em um servidor on-line, possibilitando integrar, monitorar e controlar todo o sistema de qualquer lugar, visto sua conectividade com a internet.

REFERÊNCIAS

Amazon. FreeRTOS. Disponível em:< <https://aws.amazon.com/pt/freertos/>>. Acessado em: 28 out. 2021.

Arduino IDE. Arduino. Disponível em:< <https://www.arduino.cc/en/Guide/Introduction>>. Acessado em: 16 out. 2021.

ASTAH UML. ASTAH. Disponível em:< <https://astah.net/products/astah-uml/> >. Acessado em:16 out. 2021.

BARRIQUELLO, C. H.; DENARDIN, G.W. Sistemas Operacionais de Tempo Real e Sua Aplicação em Sistemas Embarcados. São Paulo: Editora Blucher, 2019.

BATRINU, C. Projetos de Automação Residencial com ESP8266. São Paulo: Editora Novatec, 2018.

Curto-Circuito. Conversor AC/DC - 90~240VAC - 5VDC / 3A - Micro USB. Disponível em:< <https://www.curtocircuito.com.br/conversor-ac-dc-90-240vac-5vdc-3a-micro-usb.html>>. Acessado em: 22 set. 2021. 2021-a.

Curto-Circuito. Módulo Rele. Disponível em:< <https://www.curtocircuito.com.br/modulos/modulo-rele>>. Acessado em: 22 set. 2021. 2021-b.

Curto-Circuito. Placa DOIT ESP32 – ESP – WROOM-32-WIFI/Bluetooth. Disponível em:< <https://www.curtocircuito.com.br/placa-doit-esp32-esp-wroom-32-wifi-bluetooth.html>>. Acessado em: 22 set. 2021. 2021-c.

DEVMEDIA. Introdução a Requisitos de Software. Disponível em:< <https://www.devmedia.com.br/introducao-a-requisitos-de-software/29580>>. Acessado em: 16 out. 2021. 2013.

DEVMEDIA. O que é UML e Diagramas de Caso de Uso: Introdução Prática à UML. Disponível em:< <https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>>. Acessado em: 04 out. 2021. 2012.

Filipeflop. Sensor de Umidade do solo. Disponível em:< <https://www.filipeflop.com/blog/monitore-sua-planta-usando-arduino/>>. Acessado em: 16 out. 2021-a.

FilipeFlop. Sensor Ultrassom. Disponível em:< <https://www.filipeflop.com/blog/sensor-ultrassonico-hc-sr04-ao-arduino/>>. Acessado em: 16 out. 2021-b.

FRITZING. Electronics Made Easy. <<https://fritzing.org/>> acessado em 15 ago. 2021. 2021.

Google. Serial Bluetooth Terminal. Disponível em: <https://play.google.com/store/apps/details?id=de.kai_morich.serial_bluetooth_terminal&hl=pt_BR&gl=US>. Acessado em: 10 out. 2021.

JUNIOR, S. L. S. IOT – Internet das Coisas – Fundamentos e Aplicações em Arduino e Nodemcu. São Paulo: Editora Érica, 2018.

OSÓRIO, F. S. Linguagem C. Disponível em:<
<http://osorio.wait4.org/SSC0300/Reference/manualc.pdf>>. Acessado em: 16 out.
2021.

OLIVEIRA, Sergio de. INTERNET DAS COISAS COM ESP8266, ARDUINO E
RASPERRY PI. São Paulo: Editora Novatec 2021.

APÊNDICE A - PLANO DE TRABALHO

ACI: A CASA INTELIGENTE

TURMA 2019

Discente : Anderson dos Anjos de Oliveira – anderson_anjo23@yahoo.com.br

Orientador : Marcos Dinís Lavarda – marcos.lavarda@ifpr.edu.br

Resumo: Com o aumento da demanda nas tarefas domésticas, tanto no meio urbano, quanto no rural, os brasileiros despendem de tempo diário para realizar atividades corriqueiras no qual poderiam destinar à atividades de bem-estar. É nesse cenário que foi identificada uma oportunidade para elaboração do presente projeto de domótica, com o objetivo de ajudá-los com as tarefas de vida diária repetitivas. Através de um sistema embarcado ESP32 com Sistema Operacional de Tempo Real FreeRTOS, placas de sensores e atuadores auxiliares e interface com o usuário, é possível acionar e monitorar equipamentos eletroeletrônicos de forma simples e remota. Isso pode proporcionar economia de tempo e/ou recursos aos usuários com, por exemplo, o monitoramento do nível de água em uma cisterna e a automação da rega de plantas de uma horta. Utilizando um aplicativo Android genérico para comunicação Bluetooth, foi possível controlar e monitorar o sistema embarcado. O sistema foi desenvolvido utilizou como componentes o ESP32 e como periféricos os módulos de relés para acionamento de luzes e bomba d'água de irrigação, além de sensor de umidade do solo, sensor de chuva e sensor de nível da cisterna, sendo programados em linguagem C na IDE do Arduino. O projeto tem a finalidade de propiciar estudos de aplicações de baixo custo para esta finalidade de automatização e emprego de tecnologia com o uso de Sistema Operacional de Tempo Real.

1. Introdução e Justificativa

Sabemos que, com o avanço da tecnologia, tarefas simples, tanto residenciais quanto em empresas, podem ser automatizadas. Porém, ainda existem no mercado companhias que oferecem os serviços com valores fora da realidade de muitos brasileiros. Pensando nisso, a ideia principal, é criar um produto que atenda às necessidades da maioria da população e ainda ter uma redução de custo na energia de cada residência ou empresa. Os produtos aplicados já existem no mercado e são de fácil acesso. O módulo ESP32, que é o módulo principal, custa entre R\$ 30,00 e R\$ 80,00.

Etapas:

1. Definição do sistema e entrevistas com o cliente
2. Escrita do projeto de forma detalhada.
3. Levantamento dos requisitos funcionais e não funcionais
4. Diagramas do sistema (UML e DER)
5. Escolha da plataforma de desenvolvimento (Arduino)
6. Plano de negócios
7. Codificação e testes.

5. Forma de Acompanhamento/Orientação

Acompanhamento será realizado no horário de atendimento ao aluno com a frequência quinzenal entre aluno e orientador, via Google Meet, nas quartas-feiras das 17h00 às 19h00.

6. Referências Bibliográficas

Arduino IDE. Arduino. Disponível em:< <https://www.arduino.cc/en/Guide/Introduction>>.

Acessado em: 16 out. 2021.

ASTAH UML. ASTAH. Disponível em:< <https://astah.net/products/astah-uml/> >.

Acessado em:16 out. 2021.

BATRINU, C. Projetos de Automação Residencial com ESP8266. São Paulo: Editora Novatec, 2018.

OLIVEIRA, Sergio de. INTERNET DAS COISAS COM ESP8266, ARDUINO E RASPBERRY PI. São Paulo: Editora Novatec 2021.

Colombo, 14 de maio de 2021.



Marcos Dinís Lavarda
Orientador



Anderson dos Anjos de Oliveira
Aluno