

Planta Care: Sistema de irrigação remoto de plantas domésticas

Bruno Antonelli¹, Matheus Amboni², Raphael A. F. Jesus³, Leomar M. Marschalk⁴, Yan W. L. Bentes⁵

¹Engenharia de Computação – Universidade Federal de Santa Catarina(UFSC)

***Resumo.** Esse relatório descreve a implementação de um sistema de irrigação de plantas que pode ser controlado remotamente por um aplicativo mobile, através do envio e leitura de dados para a plataforma móvel Firebase, utilizando um ESP32 conectado à uma rede Wi-Fi.*

1. Introdução

Segundo a postagem no site da Exame(“10 bons motivos para você ter mais plantas em casa”), plantas ajudam no alívio do estresse do dia a dia e possuem diversos outros benefícios em um ambiente doméstico. Levando em consideração os diversos benefícios e o aumento da procura por plantas domésticas(“Pandemia e isolamento aumentam procura por cultivo de plantas em casa”), nós pensamos em um sistema de irrigação que pode ser controlado remotamente através de um smartphone com acesso à internet.

O projeto é um circuito de irrigação automatizado e um aplicativo mobile capaz de controlar a irrigação, Para realizar este projeto utilizamos um ESP32, microcontrolador de baixo custo e baixo consumo de energia que possui Wi-Fi e Bluetooth integrados, sensores de umidade e temperatura para coletar dados sobre a planta, uma instância do banco de dados Firebase Realtime Database para armazenar os dados coletados pelo ESP32 em nuvem, e por último, um aplicativo mobile capaz de se comunicar com o banco de dados. Neste relatório iremos descrever a construção deste projeto em mais detalhes com o foco em entender melhor como é realizada a comunicação do ESP32 com o banco de dados em nuvem.

2. Leitura de dados

O procedimento para recolher os dados do ambiente foi estruturado na Plataforma ESP32, uma placa de desenvolvimento criada pela Espressif Systems, podendo ser programada pelo Software Arduino IDE, tendo suporte a todas as bibliotecas do mesmo. O ESP32 CAM é uma versão especial da placa de desenvolvimento baseada no módulo ESP32S WiFi, mostrando-se um módulo eletrônico altamente tecnológico e que foi utilizado para o desenvolvimento da aplicação.

A mesma placa não tem suporte direto ao USB, tendo que necessitar de um conversor serial, ou melhor, programador FTDI para dar upload no código.

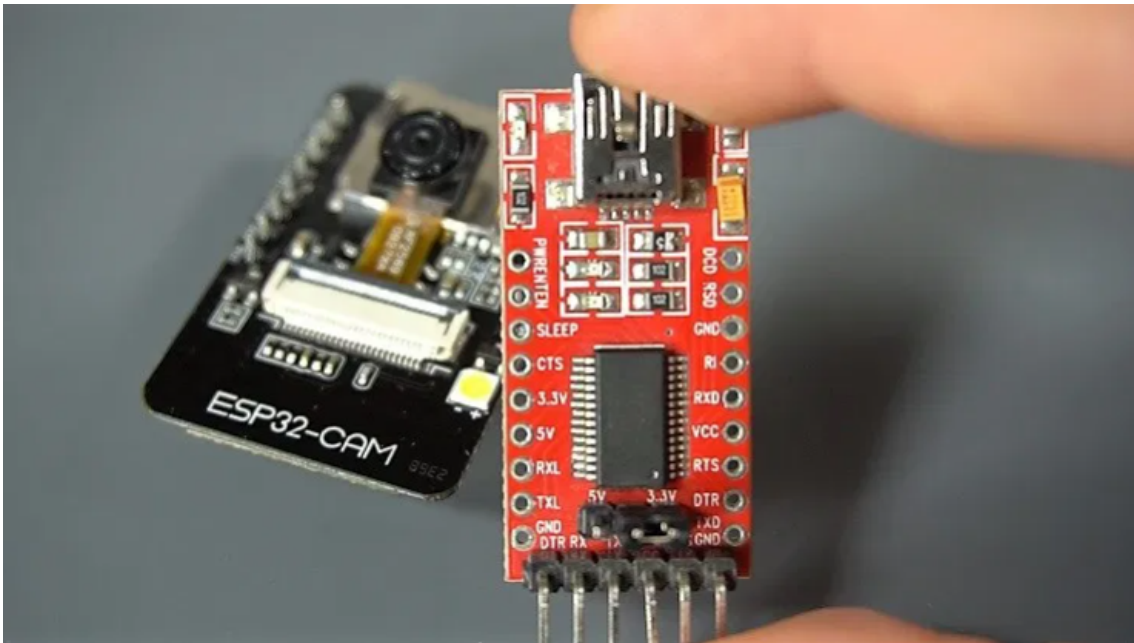


Figura 1 - Placa ESP32Cam e conversor FTDI serial

Na placa foi desenvolvido um código para acionamento de um servo motor que será responsável de abrir a tampa da garrafa para regar a planta e simultaneamente controlar um sensor de temperatura e umidade.

O motor servo utilizado tem o seu Serial Number como: SM-S4306R, podendo girar 360 Graus, tem a particularidade de ser um servo de rotação contínua, abstendo de um encoder para controle da posição em graus, mas já sendo o suficiente para a implementação da aplicação.



Figura 2 - Servo Motor SM-S4306R

Também foi utilizado um sensor de temperatura e umidade DHT11 e um sensor de umidade de solo do tipo resistivo, sendo este implementado em uma outra placa modelo esp32.

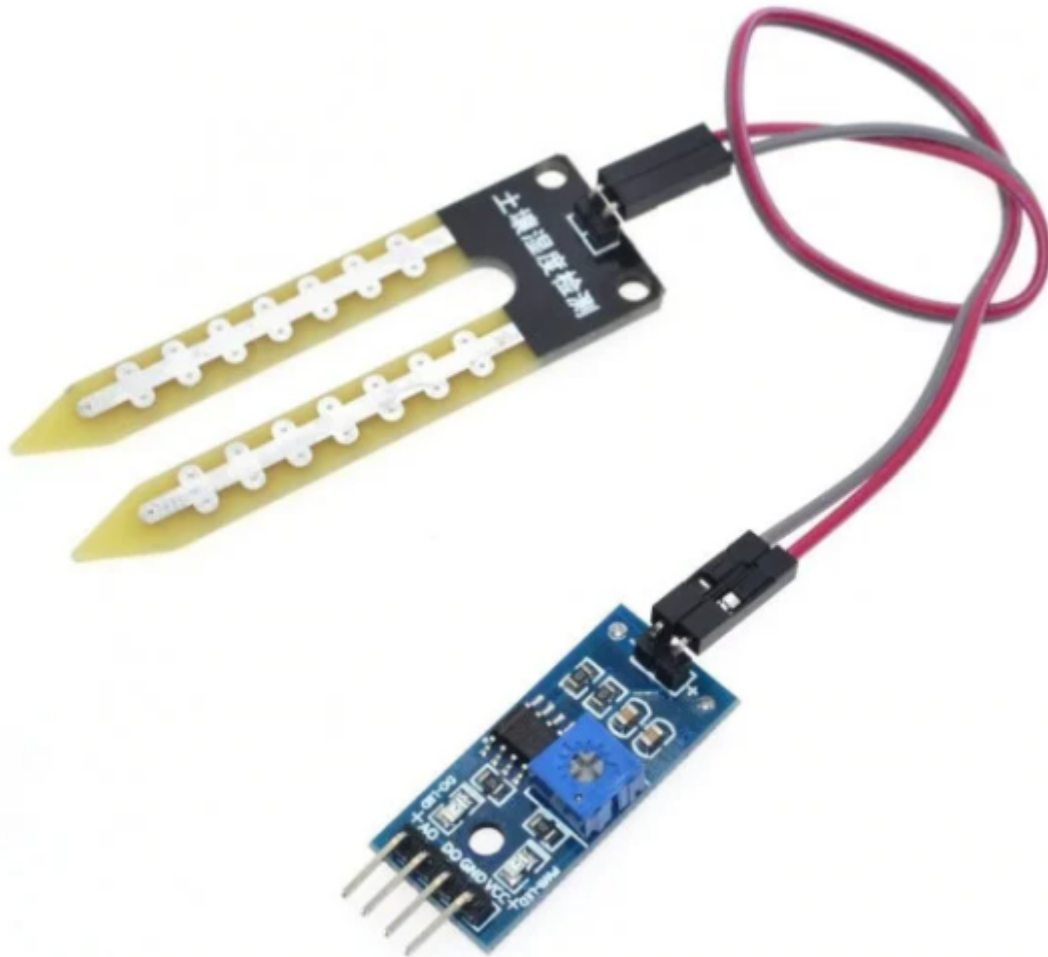


Figura 3 - Sensor resistivo de umidade de solo

3. Protocolo de comunicação

Para que o aplicativo pudesse acessar os dados coletados pelos sensores, criamos um servidor noSQL utilizando a plataforma do Google chamada Firebase para armazenar os dados coletados, dentro do Firebase foi usada uma instância chamada Realtime Database(RTDB). O Firebase Realtime Database é um banco de dados hospedado na nuvem. Os dados são armazenados como JSON e sincronizados em tempo real para cada cliente conectado. Utilizamos a biblioteca de código aberto “Firebase Arduino Client Library for ESP8266 and ESP32”, disponível em github.com/mobizt/Firebase-ESP-Client, para realizar a comunicação com o Firebase RTDB. Esta biblioteca se comunica com o RTDB através do protocolo HTTP(Hypertext

Transfer Protocol), que é um protocolo de comunicação cliente-servidor, onde o cliente manda requisições para o servidor por meio de uma URL(Uniform Resource Locator).

4. Envio de dados

A biblioteca Firebase Arduino Client Library for ESP8266 and ESP3 possui uma lista de funções que podem ser utilizadas tanto para enviar quanto para ler dados armazenados no RTDB. Ao enviar os dados dos ESPs para o Firebase, eles ficaram armazenados no banco de dados do próprio Firebase, o Realtime Database (RTDB), onde é lido de forma síncrona pela aplicação e exibe em tempo real os dados obtidos, como a temperatura e umidades do ar e solo.

Para o caso de recepção de dados através dos ESPs, foi definido um valor booleano no banco de dados, chamado de “regarPlanta”, onde ao ser definido como true através da aplicação, o servo motor era acionado fazendo assim a rega da planta e, após executar sua função, definia o valor de volta para false.

5. Aplicação

A aplicação, inicialmente feita apenas para dispositivos móveis, foi dividida em 2 para melhor organização na hora do desenvolvimento, primeiro foi desenvolvido o design da aplicação e após foi aplicada a parte do desenvolvimento, onde se deu por transformar as imagens das telas feitas no design em uma aplicação real com funcionalidades reais.

5.1 Design

O design do aplicativo foi feito todo utilizando o software Figma, que é um editor gráfico de vetor e prototipagem de projetos de design baseado principalmente no navegador web, com ferramentas offline adicionais para aplicações desktop para GNU/Linux, macOS e Windows

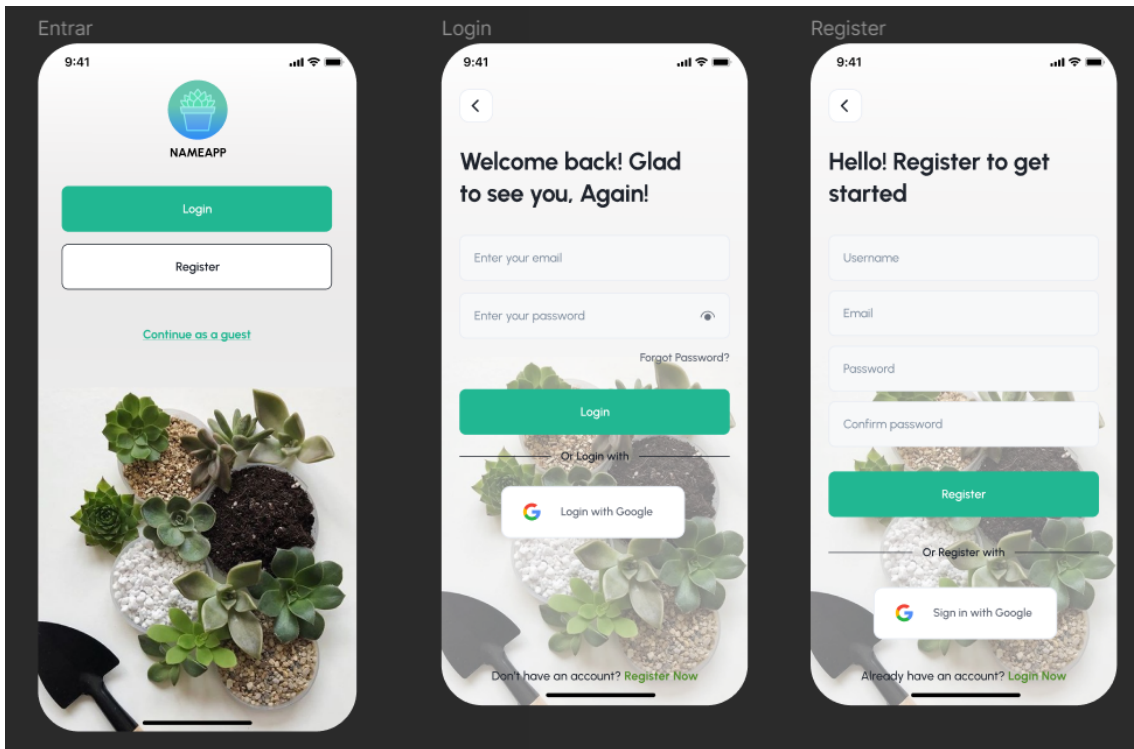


Figura 4 - Entrar, Login e Register

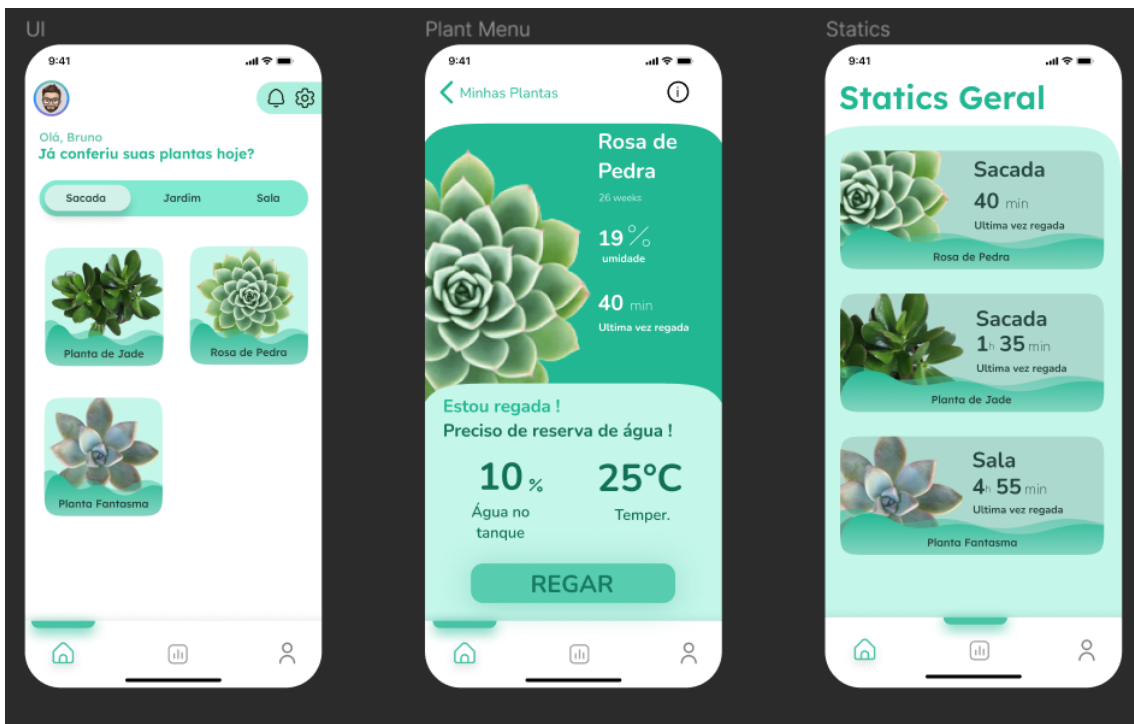


Figura 5 - Homepage, Plant Menu, Statics Pages.

5.2 Desenvolvimento

Todo o aplicativo foi desenvolvido em flutter, framework desenvolvido pela Google que compila sobre a linguagem Dart, visto que assim é possível ser instalado em IOS (Apple) e Android (Google).

5.3 Funções

Após efetuar o login o usuário será redirecionado para a Homepage. Lá é possível visualizar sua foto de perfil, os ambientes criados pelo próprio usuário e suas plantas. Ao clicar sobre a imagem de alguma planta abrirá uma nova página contendo as suas informações. Dentre elas temos: Seu nome, sua idade, umidade do ar, quando foi regada pela última vez, água no reservatório e sua temperatura. No inferior da página temos o botão para regar a planta, ao clicar o ESP 32 acionará o sistema de irrigação e atualizará as informações no banco de dados.

Referências

Mobizt, K. S. (2021, Janeiro 13). *mobizt/Firebase-ESP-Client*:  *Firebase Arduino*

Client Library for ESP8266 and ESP32. The complete, fast, secured and reliable Firebase Arduino client library that supports RTDB, Cloud Firestore, Firebase and Google Cloud Storage, Cloud Messaging and ... GitHub.

<https://github.com/mobizt/Firebase-ESP-Client>

Pandemia e isolamento aumentam procura por cultivo de plantas em casa. (2021, April 17). Agência Brasil.

<https://agenciabrasil.ebc.com.br/geral/noticia/2021-04/pandemia-e-isolamento-aumentam-procura-por-cultivo-de-plantas-em-casa>

10 bons motivos para você ter mais plantas em casa. (2017, February 8). Exame.

<https://exame.com/casual/10-bons-motivos-para-voce-ter-mais-plantas-em-casa/>