



IoT: Desenvolvimento de sistema de controle remoto via internet

IoT: Development of a remote control system via internet

Giovani Pereira Sella*, Lyssa Setsuko Sakanaka[†],
João Gabriel Franchini Câmara[‡], Eduardo Watanabe de Oliveira[§],
Romualdo Rubens de Freitas[¶], Claudio Takeo Ueno^{||}, Janksyn Bertozzi[§]

RESUMO

A conexão de objetos à internet das coisas (IoT) traz comodidade e facilita as atividades do dia a dia. O procedimento da conexão desses objetos a IoT não é predeterminado e apresenta diversas possibilidades. O presente trabalho relata o desenvolvimento da conexão de um controlador de temperatura de um fermentador de dupla camisa à internet. A conexão do protótipo do controlador à internet utiliza um microcontrolador ESP32, que tem um módulo de Wifi embutido. A escolha da interface do sistema foi um aplicativo mobile, desenvolvido em Flutter. O sistema utiliza o banco de dados em tempo real *Realtime Database* e o *Firestore database*, ambos do *Google*. Testes preliminares confirmaram a viabilidade do uso desse procedimento para automatizar sistemas. O controle em tempo real do controlador de temperatura foi bem sucedido ao utilizar o aplicativo, o banco de dados e o microcontrolador. Funcionalidades complementares para possibilitar o uso do sistema ainda devem ser desenvolvidas.

Palavras-chave: Internet das coisas, Automação, *Flutter*, Controle Remoto, *Realtime Database*.

ABSTRACT

The connection of objects to the Internet of things brings comfort and makes everyday activities easier. Connecting these objects to the IoT is not straightforward and presents many possibilities. The present work shows the development of the connection of a controller of a double jacket fermenter to the internet. The connection of the fermenter controller utilizes the microcontroller ESP32, which has an onboard wifi module. The choice of interface of the whole system was a mobile App, builded with Flutter. The system utilizes Google's Realtime database and the Firestore database. Tests confirm the viability of the use of this procedure of automatization of systems. The real time control was successfully applied connecting the App, the database and microcontroller. Some additional features still need to be developed to make the whole system usable.

Keywords: IoT, Automation, Flutter, Remote Control, Realtime Database.

1. INTRODUÇÃO

IoT (em português: internet das coisas) é o conceito de conexão dos mais diversos objetos à rede mundial de computadores. Essa conexão traz diversas vantagens, tais como a comodidade do controle desses objetos remotamente. Para essa conexão à internet, segundo Santos (2021), os objetos devem possuir “capacidade computacional e de comunicação”. A conexão entre os mais diversos objetos via internet necessita do desenvolvimento de softwares embarcados para que a comunicação seja possível.

* TINFEM, Instituto Federal do Paraná, Londrina, Paraná, Brasil; gipereirasella@gmail.com

[†] Universidade Tecnológica Federal do Paraná, Campus Londrina, PR, Brasil; lyssa@utfpr.edu.br

[‡] Arbo Imóveis, Londrina, Paraná, Brasil; franchini211@gmail.com

[§] TINFEM, Instituto Federal do Paraná, Londrina, Paraná, Brasil; eduardowatanabe1227@gmail.com

[¶] Instituto Federal do Paraná, Londrina, Paraná, Brasil; romualdo.freitas@ifpr.edu.br

^{||} Universidade Tecnológica Federal do Paraná, Campus Londrina, PR, Brasil; takeo@utfpr.edu.br

[§] Universidade Tecnológica Federal do Paraná, Campus Londrina, PR, Brasil; janksynbertozzi@utfpr.edu.br



SEI-SICITE 2021

Pesquisa e Extensão para um mundo em transformação

A possibilidade de monitorar, e controlar equipamentos remotamente, é uma solução muito prática pois permite essa ação sem a necessidade de deslocamento do usuário ao local do equipamento. Segundo Câmara (2020), “No laboratório de pesquisa A303, na UTFPR campus Londrina, há um fermentador de dupla camisa que não possui um dispositivo, como um banho termostático, para o controle de temperatura do processo de fermentação”. Em face a este problema, deu-se o desenvolvimento e construção de um dispositivo de controle de temperatura para o fermentador com a capacidade computacional e de comunicação para a conexão via internet das coisas. A interface de comunicação do dispositivo em questão é um aplicativo mobile.

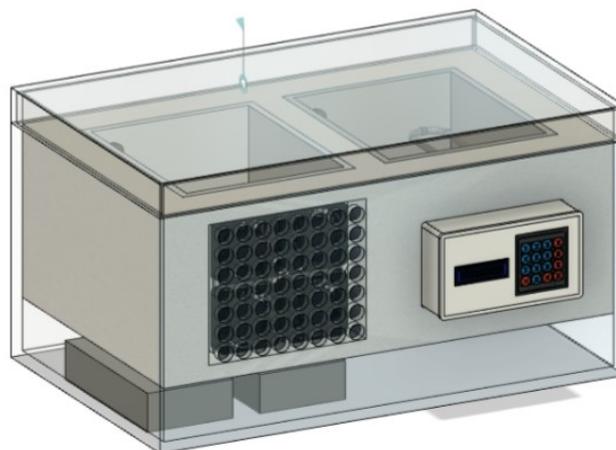
Assim sendo, o objetivo do presente trabalho foi o desenvolvimento de um sistema de conexão e comunicação entre o celular e o dispositivo de controle do fermentador, por meio de uma aplicativo mobile via internet.

2. METODOLOGIA

2.1 MATERIAL

A base deste projeto foi um protótipo de controle de temperatura para um fermentador de dupla camisa, desenvolvido por Câmara (2020). Este protótipo possui atuadores e sensores, além de um microcontrolador com conexão à internet, que formam um instrumento de implementação do sistema de controle remoto. O projeto do protótipo seguiu-se da seguinte forma: Há dois reservatórios, o primeiro contém quatro sensores de temperatura e duas resistências de 500w, denominado de reservatório de água quente; o segundo, contém quatro sensores de temperatura e seis células de peltier, denominado de reservatório de água fria. Para promover uma boa circulação de água dentro dos reservatórios, foram adicionadas duas bombas de água em cada um. O microcontrolador que irá viabilizar o controle desses componentes é o ESP32, que atende a necessidade do projeto por ter um custo baixo e conexão à internet via WIFI.

Figura 1 – Design do protótipo de controlador do fermentador.



Fonte: Câmara (2020).



A aplicação mobile, que possibilita a comunicação do pesquisador com o equipamento, utiliza o *Flutter*, *framework* do *Google*, com a linguagem de programação *Dart*. Essa foi a escolha da tecnologia do App, pois essa ferramenta apresenta fácil compreensão e conta com diversos tutoriais. O *Flutter* facilita o uso do estilo *Material Design*, criado pelo *Google*, que apresenta alta capacidade de visualização de dados e navegabilidade pelo App.

Para que seja possível a comunicação remota via internet, foi necessário o uso de um banco de dados (BD) em tempo real. Dentre os disponíveis, optou-se pelo *Realtime Database* do *Google*. Além do *Flutter* ter uma interação fácil com esse BD por ser do ecossistema *Google*, é gratuito e relativamente simples de usar, com muitos materiais de referência. Seu papel é o de armazenar os dados do protótipo em tempo real. Juntamente com o *Realtime Database*, o uso do *Firestore Database*, também do ecossistema *Google*, se mostrou importante para o armazenamento de dados que não precisam ser utilizados em tempo real, como o histórico das leituras de temperatura do fermentador.

2.2 COMUNICAÇÃO

O papel do BD no projeto é receber as informações vindas do microcontrolador, como a medição de temperatura, atividade dos atuadores e dados do produto fermentado. Uma vez com esses dados, o BD permite que a aplicação mobile obtenha essas informações e as disponibilize para o usuário. De maneira inversa, a aplicação mobile envia as instruções para o BD e permite que o microcontrolador leia e execute sua função de acordo com os parâmetros do usuário. Para isso é necessário a criação de variáveis no *Realtime database* que possuem um endereço para que outros dispositivos conectados à internet possam interagir sistematicamente com os dados.

A programação do microcontrolador foi construída na linguagem de programação C. O papel desse código, além de controlar os atuadores e sensores, é o de, com a ajuda de bibliotecas, enviar as medições para o BD. A leitura de dados do BD também é feita pelo código do ESP32, as informações coletadas são armazenadas e utilizadas no funcionamento do protótipo de controle de temperatura. Para a programação, foi utilizado o *Integrated Development Environment (IDE) Arduino*, disponível gratuitamente em endereço eletrônico (<https://www.arduino.cc/en/software>).

A aplicação mobile, que poderá ser executada pelo smartphone pessoal de cada pesquisador, é a interface de comunicação do usuário com o aparelho de pesquisa. O App possibilita a visualização e modificação das variáveis do BD em tempo real. Para a programação, foi utilizado um simulador de dispositivos Android integrado ao IDE *Android Studio* (<https://developer.android.com/studio>).

2.3 MANEJAMENTO DE DADOS

A maneira em que as informações devem ser direcionadas, entre a aplicação mobile, os BD e o microcontrolador é muito importante, pois é provável que aconteçam falhas no sinal de internet. É necessário que o protótipo, caso não consiga se conectar ao BD, continue sua tarefa para não comprometer a pesquisa em andamento. Para minimizar esses problemas, para cada início de um processo fermentativo utilizando o fermentador do laboratório, o usuário deve fornecer todos os parâmetros do processo a ser seguido. O papel da aplicação mobile, além de fornecer essas instruções e dar início ao processo, é de visualizar e alterar quaisquer instruções durante o processo. Caso haja perda na conexão, o microcontrolador seguirá as últimas instruções recebidas.



A coleta de dados é realizada por meio do *Firestore Database*. O usuário informa no início do processo quantas medições serão realizadas e o intervalo entre elas. Esses dados são enviados pelo microcontrolador ao BD. Porém, caso aconteça alguma falha na conexão, o microcontrolador irá armazenar temporariamente esses dados em sua memória interna, até que a conexão se estabeleça novamente.

2.4 DESIGN DO APLICATIVO

A construção do design foi feita utilizando o programa *Adobe XD*, e foi estruturado pensando em uma futura integração de outros dispositivos disponíveis em um laboratório de pesquisa, como uma câmara climatizada, possibilitando uma conexão de vários equipamentos de um laboratório de pesquisa ao mesmo aplicativo. Para tanto, três telas foram criadas com propósitos diferentes, conforme indicado no Quadro 1.

Quadro 1 – Telas do aplicativo e seus itens.

Tela 1 - Home	Tela 2 - Processo	Tela 3 - Novo processo
Visualização de todos os dispositivos conectados;	visualização de todas as informações do processo	Inserção de dados para novos processos
Botão de criação de novo processo;	Alteração de instruções e parâmetros	
Pré visualização das informações do processo	Armazenamento de dados coletados no dispositivo	

Fonte: Autoria própria (2021).

3. RESULTADOS E DISCUSSÃO

Inicialmente foi necessário a determinação, para cada reservatório e para o fermentador, de algumas variáveis no *Realtime database* do Google (Quadro 2). Neste Quadro estão definidas as variáveis e a funcionalidade da escolha de cada uma no campo “Função”.

Quadro 2 – Variáveis do Banco de Dados.

Variável	Tipo	Função
turn	Boolean (verdadeiro ou falso)	Representa a disponibilidade dos reservatórios e do fermentador, caso haja algum problema nos reservatórios, aqui será indicado seu mau funcionamento.
temperatura atual	Float (números racionais)	indica a leitura sendo feita pelo microcontrolador.
temperatura máxima e temperatura mínima	Float (números racionais)	indicam o intervalo de temperatura que os reservatórios e fermentador devem manter, determinada pelo usuário.

Fonte: Autoria própria (2021).

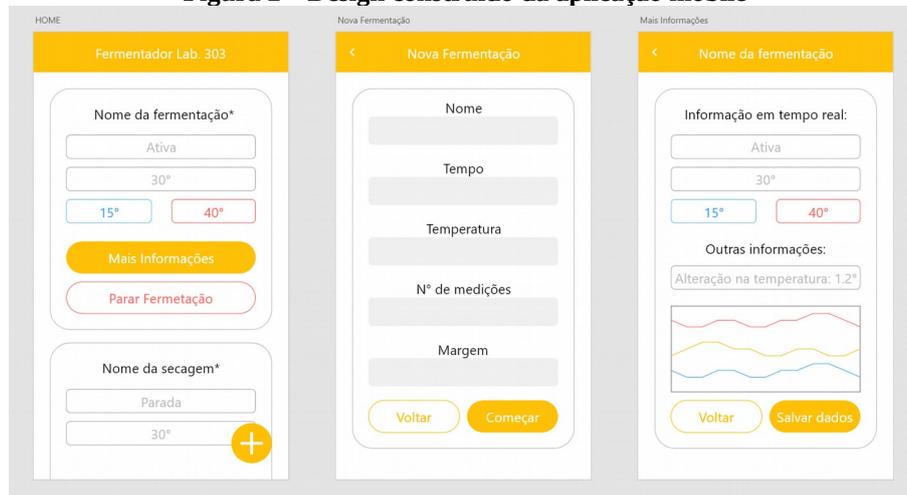
Uma vez definidas as variáveis, devem ser definidos o tipo desta variável, que pode ser do tipo “Boolean” ou “Float”. A escolha de cada uma delas foi feita com base nas informações que iriam armazenar; por exemplo, as variáveis “turn” só poderão armazenar dois valores, disponível (verdadeiro) ou não (falso), e



por isso foram definidas como “Boolean”. As variáveis de temperatura são do tipo “Float”, pois representam valores em graus celsius, e estes podem ser diversos e constituírem números não inteiros.

O Design do App (Figura 2) foi realizado no programa *Adobe XD* e conta com recursos de navegação entre telas. Sua elaboração foi utilizada para guiar o desenvolvimento do App em *Flutter*, assim como averiguar com os pesquisadores se o App contém todas as ferramentas necessárias.

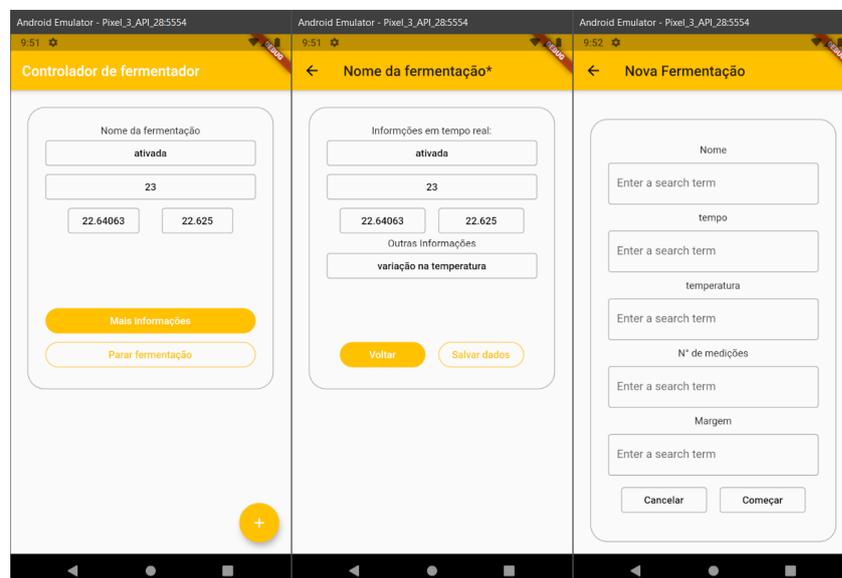
Figura 2 – Design construído da aplicação mobile



Fonte: Autoria própria (2021).

A partir das telas planejadas, iniciou-se o desenvolvimento do App (Figura 3). Alguns recursos como o uso do *Firestore Database*, conseqüentemente o armazenamento dos dados no dispositivo e visualização do gráfico, *login* com conta do *Google* e envio dos dados iniciais para início da fermentação não puderam ser concluídos, pois dependem da finalização do protótipo. Com o App desenvolvido até agora foi possível enviar e buscar as variáveis do BD sistematicamente.

Figura 3 – Simulador executando a aplicação mobile.



Fonte: Autoria própria (2021).



Após a programação do microcontrolador, os testes com o controle de todos os atuadores e sensores, assim como a conexão WIFI, foi bem sucedido. Com o auxílio de bibliotecas para o uso do *Google Firebase*, foi possível enviar e buscar as variáveis do BD sistematicamente. O código não está completo, pois ainda há pendências nos componentes do protótipo (reservatórios de água quente e fria).

4. CONCLUSÃO

Apesar de incompleto, o presente trabalho apresenta resultados promissores. Os principais elementos do projeto, que são o microcontrolador do protótipo, o BD e o App, se comunicam e o sistema se mostrou uma solução viável e de baixo custo para a implementação do controle via internet de equipamentos do laboratório. O desenvolvimento das programações do App, do microcontrolador e do BD dependem do protótipo físico que ainda não foi concluído. Quando o protótipo estiver em pleno funcionamento, será possível melhorar o desenvolvimento da parte do Software.

AGRADECIMENTOS

Ao CNPq pela bolsa concedida (PIBIC-EM para GPS e EWO), à UTFPR - campus Londrina, em especial ao técnico Márcio José Alves e ao Instituto Federal do Paraná (IFPR) - campus Londrina pela parceria e financiamento. À empresa CINFEL-Comércio e Indústria de Ferragens pela doação da peça de alumínio.

REFERÊNCIAS

CÂMARA, J.G.F. et al. **Desenvolvimento de unidade de controle de temperatura de baixo custo para fermentador de bancada**. In: XXV Seminário de Iniciação Científica e Tecnológica da UTFPR 2020, Toledo, PR. Disponível em : <http://seisicite2020.td.utfpr.edu.br/trabalhos/> Acesso em: 14 mar. 2021.

SANTOS, B.P. et al. **Internet das Coisas: da Teoria à Prática**. 2021. Disponível em : <http://35.238.111.86:8080/jspui/handle/123456789/329> Acesso em: 14 set. 2021.