

HIDRÔMETRO INTELIGENTE:

auxiliar na visualização do consumo e desperdício de água e seu impacto financeiro

Pedro Henrique Silva Marçal
Graduando em Sistemas de Informação – Uni-FACEF
pedro.h.silva.marcal@gmail.com

Vinícius Gabriel Pereira Brito
Graduando em Ciência da Computação – Uni-FACEF
viniciustabriel.pb@gmail.com

Débora Pelicano Diniz
Mestra em Ciência da Computação – UFSCar
deboradiniz@facef.br

Resumo

A água é um recurso vital para a vida no planeta Terra e a escassez dela é um desafio global, sendo fundamental a adoção de medidas para gerenciar e conservar sua disponibilidade. Nesse contexto, este artigo descreve o desenvolvimento de um projeto que tem como objetivo auxiliar as pessoas no gerenciamento do consumo de água, promovendo a economia financeira e conscientizando a população sobre a importância da preservação deste recurso. Foi levantada uma base teórica relacionada à importância da água e seu consumo consciente, e também os conceitos de sistemas embarcados, Internet das Coisas e o protocolo MQTT. A partir desse embasamento teórico, foi realizado o projeto do sistema, utilizando princípios da Engenharia de Software e os métodos de documentação. Em seguida, foi realizada a implementação do aplicativo que utiliza o Sistema Gerenciador de Bancos de Dados PostgreSQL no *backend* criado em NodeJS. Os dados são coletados por Hidrômetros Inteligentes construídos com o microcontrolador ESP8266 NodeMCU conectado a um sensor de fluxo, e são instalados em residências ou estabelecimentos da população que adquire esse sistema embarcado. Este hidrômetro foi programado utilizando o Arduino IDE, que persiste dados no *backend* que são acessados por um aplicativo *mobile* desenvolvido em React Native. O hidrômetro coleta dados de consumo por meio do sensor de fluxo, os quais são transmitidas para o *backend* utilizando o protocolo MQTT, e esses dados são apresentados em um gráfico no aplicativo e convertidos para valores monetários pagos a cada mês. O aplicativo também apresenta o consumo regional e fornece notícias e dicas com o intuito de incentivar o uso consciente da água.

Palavras-chave: Consumo de Água. ESP8266. Hidrômetro. Internet das Coisas. NodeMCU.

Abstract

Water is a vital resource for life on planet Earth and its scarcity is a global challenge, making it essential to adopt measures to manage and conserve its availability. In this context, this article describes the development of a project that aims to help people manage their water consumption, promoting financial savings and making the population aware of the importance of preserving this resource. A theoretical basis was raised related to the importance of water and its conscious consumption, as well as the concepts of embedded systems, Internet of Things and the MQTT protocol. Based on this theoretical basis, the system design was carried out using Software

Engineering principles and documentation methods. Then, the implementation of the application that uses the PostgreSQL Database Management System was carried out in the backend created in NodeJS. Data are collected by Smart Hydrometers built with the ESP8266 NodeMCU microcontroller connected to a flow sensor, and are installed in homes or establishments of the population that purchases this embedded system. This hydrometer was programmed using the Arduino IDE, which persists data in the backend that are accessed by a mobile application developed in React Native. The water meter collects consumption data through the flow sensor, which are transmitted to the backend using the MQTT protocol, and this data is presented in a graph in the application and converted into monetary amounts paid each month. The application also presents regional consumption and provides news and tips in order to encourage the conscious use of water.

Keywords: ESP8266. Internet of Things. NodeMCU. Water Consumption. Water Meter.

1. Introdução

A água é um recurso vital para a manutenção da vida no planeta Terra, sendo utilizada em diversas finalidades. De acordo com Batista (2022), a presença da água é fundamental para questões relacionadas à natureza e aos seres vivos, incluindo a sobrevivência e a regulação do clima, bem como em atividades cotidianas como as indústrias, o transporte e a limpeza.

Considerando que a água é um recurso natural de suma importância, o desperdício dela se torna uma prática totalmente inviável, pois como foi abordado por Hexag Educação (2022), o desperdício de água tem acarretado diversos problemas para bilhões de pessoas, e a tendência é que esses problemas se intensifiquem se não houver mudança de comportamento da população. Dentre os problemas destacados, incluem-se a falta de acesso à água limpa, a ausência de saneamento básico, o surgimento de doenças decorrentes do consumo de água contaminada, a poluição (resultado do uso inadequado do recurso), dentre outros.

Além do consumo de água, outro aspecto relevante é o custo mensal para a sua utilização. De acordo com informações divulgadas pelo Correio do Povo (2022), em 2022 o Brasil apresentou o maior índice de contas atrasadas de água e luz, totalizando mais de 37 milhões de faturas em atraso, o que representa 23,7% do total. Desse modo, é possível perceber que a conta de água representa um fator de grande impacto financeiro para a população brasileira, de tal forma que um controle do consumo pode evitar altas cobranças.

Tendo em vista a relevância da água para a sustentabilidade do planeta e a urgência de mudança de hábitos de consumo por parte da população, visando a preservação ambiental e a economia financeira, o objetivo do trabalho apresentado neste artigo foi desenvolver uma solução que auxilie os usuários a controlar a utilização de água, para isso o cliente adquire o Hidrômetro Inteligente e esse aparelho coleta as informações de consumo no ambiente que é instalado (por exemplo: em casa, na empresa, etc.), e realiza o monitoramento do uso e o valor financeiro aproximado do custo por meio de um aplicativo. Como objetivos específicos, o trabalho propôs uma conscientização dos usuários da solução por meio de notícias e dicas que incentivam a adoção de boas práticas para controlar a utilização, além de um monitoramento de consumo por região, para conhecer melhor quais pontos da cidade sofrem de desperdício em volumes maiores, e para que programas, políticas públicas e boas práticas possam ser adotadas nessas regiões.

A solução apresentada é um projeto em fase de desenvolvimento, que requer aprimoramento antes de se tornar um produto comercial. Nesse sentido, é fundamental dedicar mais tempo e envolver mais pessoas na revisão e correção de questões importantes, como segurança, conformidade com as leis vigentes, disponibilidade de uso e outros fatores que possam impactar a experiência do usuário e o êxito do projeto.

Nas próximas seções são abordados os temas relacionados à importância da água para a população, visando compreender melhor a sua necessidade e do meio ambiente para adotar melhores hábitos de consumo. São tratados também os Sistemas Embarcados com tecnologias de Internet das Coisas (do inglês, *Internet of Things* - IoT) por meio de coleta e transmissão de dados com base no protocolo MQTT, para que fosse possível compreender como IoT pode solucionar o problema de projeto identificado.

Após isso, é apresentada a etapa do desenvolvimento, iniciando pelos princípios de Engenharia de Software e as documentações criadas para o desenvolvimento do sistema proposto, e em seguida a modelagem do banco de dados e o desenvolvimento da solução com base nos documentos.

2. Importância do Consumo Consciente da Água

Em 22 de março de 1992, a Organização das Nações Unidas (ONU) redigiu a Declaração Universal dos Direitos da Água, afirmando que:

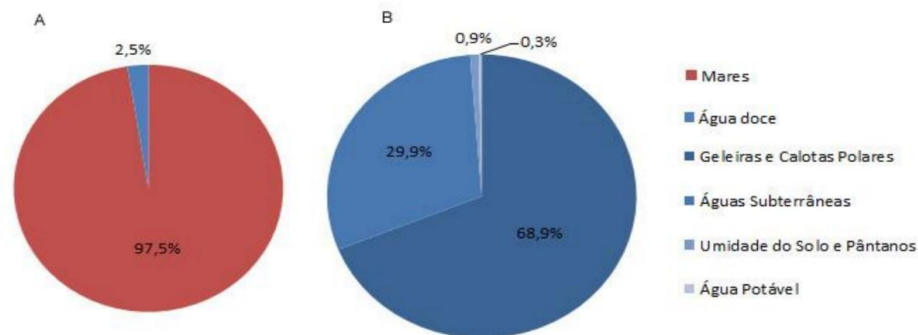
A água é a seiva de nosso planeta. Ela é condição essencial de vida de todo vegetal, animal ou ser humano. Sem ela não poderíamos conceber como são a atmosfera, o clima, a vegetação, a cultura ou a agricultura. (DECLARAÇÃO UNIVERSAL DOS DIREITOS DA ÁGUA, Art. 2º, São Paulo 2017 p. 1)

Percebe-se, então, que a água é muito importante para todo o planeta. Contudo, a demanda pela água tem aumentado consideravelmente nas últimas décadas, a uma taxa média de 1% ao ano desde os anos 80 (KONCAGÜL, 2019). Este cenário deve se manter até o ano de 2050, segundo o autor, conforme previsões de especialistas, em função do crescimento populacional e do aumento das demandas dos países emergentes. Kongagül (2019) afirma, ainda, que o consumo de água em 2050 será entre 20% e 30% maior do que os níveis atuais, o que gera preocupação quanto à disponibilidade desse recurso natural tão importante.

Essa crescente demanda por água e os problemas ambientais decorrentes têm gerado preocupações quanto à sua disponibilidade para as próximas gerações. Embora a água cubra cerca de 71% da superfície da Terra, apenas 2,5% está disponível para uso humano e a quantidade de água potável é ainda menor, chegando somente a 0,3%, conforme pode ser observado nos gráficos apresentados na Figura 1 (VERIATO et al, 2015). Esses dados alarmantes indicam a importância de conscientizar a população sobre o consumo consciente de água, a fim de preservar este recurso natural vital para a sobrevivência da humanidade.

Segundo dados do Instituto Trata Brasil (2022), em 2020 cerca de 40% da água potável produzida no país foi desperdiçada, o que representa uma perda de 7 bilhões de metros cúbicos de água. O desperdício pode ser causado por diversos fatores, como vazamentos em tubulações, torneiras mal reguladas, descargas de banheiro defeituosas, entre outros. Essa perda de água pode ser evitada com medidas simples, como a instalação de sistemas de medição de consumo e a realização de manutenções regulares em encanamentos e sistemas hidráulicos.

Figura 1: Quantidade total de água no planeta (A) e de água doce (B)



Fonte: VERIATO et al, 2015, p. 2

Conforme noticiado pelo G1 (2022), em relação ao desperdício de água, o Brasil apresentou índices preocupantes no ano de 2020, registrando o quinto ano consecutivo de piora nesse aspecto. Nesse período, foi constatado um desperdício de 40% da água, o que equivale a uma quantidade capaz de preencher cerca de 8 mil piscinas olímpicas, enquanto 35 milhões de brasileiros não possuem acesso sequer à água limpa.

Em setembro de 2021, a cidade de Franca/SP passou por um período sem chuva, o que resultou na falta de água nos reservatórios da provedora da cidade. Como abordado pelo G1 (2021), “os moradores vinham sendo submetidos a um sistema de rodízio no abastecimento, que chegou a deixar as casas por até 36 horas sem o recurso”, e mesmo após este ocorrido, a provedora ainda pediu que se faça o bom uso do recurso, para evitar problemas no futuro.

Além disso, como pontuado por Mierzwa e Hespanhol (2014), as indústrias são grandes consumidoras de água, em especial aquelas que utilizam da transformação e processamento de recursos naturais. A indústria utiliza a água para diversas finalidades, incluindo agricultura, geração de energia e transporte. No entanto, é comum o uso inadequado desse recurso natural, resultando em desperdício e poluição. Para solucionar essa questão, é importante que as empresas adotem técnicas de economia e cuidado com a água, a fim de reduzir os custos e contribuir para a preservação do meio ambiente, beneficiando toda a população pela conservação desse bem vital.

Os autores Mierzwa e Hespanhol (2014) ainda pontuam que “um dos primeiros passos para fazer um uso consciente da água é a quebra do paradigma de abundância que se estabeleceu no Brasil”, e que para isso, é necessário conscientizar os usuários da limitação deste recurso, pois a sua depuração natural é mais lenta que a frequência em que a população a desperdiça e a polui.

Diante da importância do consumo consciente da água, é fundamental que as políticas públicas e as ações individuais sejam voltadas para a preservação desse recurso vital para a vida no planeta. A conscientização e a adoção de medidas para o uso racional da água são essenciais para garantir um futuro sustentável para as próximas gerações.

3. Sistemas Embarcados

De acordo com Chase (2007), “um sistema é classificado como embarcado quando este é dedicado a uma única tarefa e interage continuamente com o ambiente a sua volta por meio de sensores e atuadores”, esses sistemas

Todo sistema embarcado é composto por uma unidade de processamento, que é um circuito integrado, fixado a uma placa de circuito impresso. Possuem uma capacidade de processamento de informações vinda de um software que está sendo processado internamente nessa unidade, logo o software está embarcado na unidade de processamento (CHASE, 2007, p. 3).

Na Figura 2 está apresentada a lógica de um sistema embarcado.

Figura 2: Lógica de um sistema embarcado usando um microprocessador como unidade de processamento



Fonte: CHASE, 2007, p. 3

Segundo Barros e Cavalcante (2010), normalmente os sistemas embarcados têm algumas características em comum, como:

- São projetados para executar uma única tarefa de forma contínua, diferente de um computador, por exemplo, que tem inúmeras funcionalidades;
- Possuem algumas limitações mais rigorosas, tais como: um baixo custo, ocupar pouco espaço, ter um melhor desempenho, não necessitar de uma alta capacidade de processamento, entre outros. Essas restrições são aplicáveis em outros projetos, mas em sistemas embarcados elas tendem a ser mais rígidas;
- Têm uma execução em tempo real, reagindo de acordo com o ambiente em que se encontram para que tenham uma resposta também em tempo real. Isso possibilita que o sistema, por meio de sensores, possa captar alterações e tomar uma decisão, como enviar dados para um outro aparelho ou iniciar uma ação em um atuador.

3.1. Internet das Coisas

Internet das Coisas é uma tecnologia emergente que possibilita a conexão de dispositivos físicos à internet, permitindo a troca de informações entre eles e com outros sistemas. Segundo Gubbi *et al* (2013), a Internet das Coisas é uma rede formada por objetos físicos, dispositivos, veículos, edifícios e outras coisas que estão conectadas e equipadas com sensores, atuadores e sistemas de comunicação, capazes de coletar, processar e analisar dados para automatizar tarefas e tomar decisões.

Com a IoT, dispositivos como eletrodomésticos, automóveis, equipamentos de saúde, entre outros, podem se comunicar e interagir entre si, tornando a vida cotidiana mais eficiente e conveniente. Segundo Atzori *et al* (2010), a IoT tem o potencial de transformar os serviços oferecidos pelas cidades.

indústrias, saúde, transporte, entre outros setores.

Conforme destacado por Gubbi *et al* (2013), para tornar possível a aplicação da IoT, é imprescindível que os dispositivos possuam sensores, atuadores e sistemas de comunicação. Enquanto os sensores são responsáveis por coletar os dados dos dispositivos, os atuadores são responsáveis por executar ações em resposta às informações recebidas. Além disso, os sistemas de comunicação são necessários para transmitir os dados coletados e ações realizadas pelos dispositivos.

A comunicação entre os dispositivos pode ser realizada de diferentes formas, como por meio de Wi-Fi, Bluetooth, RFID, entre outras tecnologias. Além disso, a IoT também envolve o uso de tecnologias como a nuvem e a análise de dados para processar as informações coletadas pelos dispositivos (GUBBI *et al*, 2013).

3.2. Protocolo MQTT

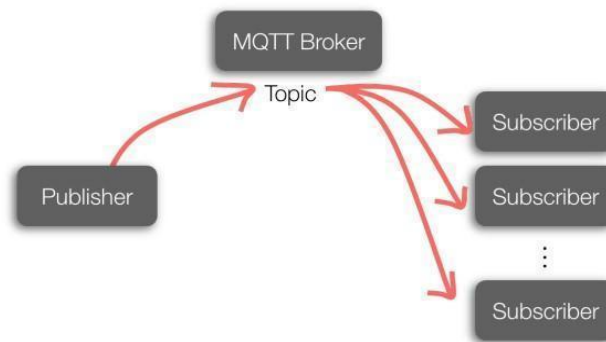
O protocolo MQTT (do inglês, *Message Queuing Telemetry Transport*) é um protocolo de comunicação. De acordo com Martins e Zem (2015), é um protocolo de mensagens de arquitetura publicação/assinatura, projetado para trabalhar com redes limitadas, com baixa largura de banda e alta latência, visando a confiabilidade e garantia da entrega.

Pelo fato de o protocolo MQTT trabalhar com o modelo de publicação/assinatura, em que os dispositivos publicam informações em um tópico e outros dispositivos assinam esse tópico para receber as informações, esse modelo permite que os dispositivos IoT transmitam informações de maneira eficiente, sem sobrecarregar a rede ou os dispositivos envolvidos. O protocolo usa um intermediário chamado de MQTT *Broker*, que é responsável por rotear as mensagens entre os publicadores e os assinantes, como pode ser visto na Figura 3 (JUCÁ e PEREIRA, 2018).

A Qualidade de Serviço (QoS) é um recurso fundamental do protocolo MQTT que permite diferentes níveis de confiabilidade e segurança para as operações realizadas em ambientes IoT. Segundo Al-Fuqaha *et al* (2015), a QoS é definida como "a capacidade de um sistema de garantir a entrega de mensagens com um determinado nível de confiabilidade, que pode ser medido por critérios como a entrega no máximo uma vez, a entrega pelo menos uma vez e a entrega exatamente uma vez".

O MQTT oferece três níveis de QoS, cada um adequado para diferentes necessidades de aplicações IoT. No nível de QoS 0, ou Entrega no máximo uma vez, a mensagem é enviada apenas uma vez, sem garantia de que será entregue ao destinatário. Este nível é adequado para aplicações que não requerem confiabilidade na entrega da mensagem. No nível de QoS 1, ou Entrega pelo menos uma vez, a mensagem é enviada e o receptor confirma o recebimento. Caso a confirmação não seja recebida, a mensagem é enviada novamente até que a confirmação seja recebida. Esse nível de QoS é adequado para aplicações que requerem uma entrega confiável, mas em que a possibilidade de receber mensagens duplicadas não é crítica. Finalmente, no nível de QoS 2, ou Entrega exatamente uma vez, a mensagem é enviada e o receptor confirma o recebimento, garantindo que a mensagem seja entregue apenas uma vez, sem duplicação ou perda (MARTINS; ZEM, 2015).

Figura 3: Recepção e transmissão de mensagens pelo Broker



Fonte: JUCÁ e PEREIRA, 2018, p. 200

A escolha do nível de QoS adequado deve ser baseada nas necessidades específicas da aplicação e pode afetar significativamente o desempenho e a eficiência da rede. Aplicações que envolvem o controle de dispositivos críticos, como sistemas de segurança ou equipamentos médicos, podem exigir um nível de QoS mais alto para garantir que as mensagens sejam entregues corretamente e em tempo hábil.

4. Engenharia de Software e Suas Documentações

A Engenharia de Software é uma disciplina que busca a aplicação sistemática de princípios, métodos e ferramentas para o desenvolvimento de software de qualidade. Uma das atividades essenciais nesse processo é a documentação, que pode ser definida como a descrição clara e completa do software e de todo o processo de desenvolvimento (SOMMERVILLE, 2011).

A documentação é importante para a comunicação entre membros da equipe de desenvolvimento, bem como para a compreensão do software por parte dos usuários. De acordo com Pressman e Maxim (2016), a documentação é uma forte conexão entre o processo de desenvolvimento e a solução final, que tem toda a base para a comunicação entre os desenvolvedores do projeto e as partes interessadas.

Além disso, a documentação é importante para a manutenção do software. Com o passar do tempo, o software pode precisar de atualizações ou correções de *bugs*. A documentação pode ajudar a equipe de manutenção a entender o software e identificar áreas que precisam ser modificadas. Como destacado por Sommerville (2011), no ciclo de vida do software, a manutenção é a etapa mais cara, e é fundamental que aqueles desenvolvedores que forem prestar o suporte compreendam a solução, sua estrutura, funcionalidade e comportamento.

Por fim, a documentação pode ajudar a garantir a qualidade do software. Ao registrar todas as etapas do desenvolvimento, a equipe de desenvolvimento pode garantir que todos os requisitos foram atendidos e que o software foi testado adequadamente. Segundo Pfleeger e Atlee (2009), é fundamental fazer uso das documentações para a garantia da qualidade de software, uma vez que assegura que os requisitos sejam atendidos corretamente.

O desenvolvimento do sistema apresentado neste artigo seguiu as boas práticas da Engenharia de Software e várias documentações foram geradas e estão apresentadas a seguir.

4.1. Engenharia de Requisitos

Definir os requisitos do sistema é uma tarefa muito importante para o projeto e deve ser iniciada logo no planejamento pois, segundo Sommerville (2011), esses requisitos definem o que a solução deve oferecer, bem como também as suas limitações, uma vez que esse sistema precisa agregar valor ao usuário, atendendo às suas necessidades.

Assim, no início do projeto a engenharia de requisitos foi realizada, envolvendo procedimentos da elicitação de requisitos, análises e a sua devida documentação.

Inicialmente, por meio de reuniões entre os autores e uso da ferramenta 5W1H, foram estabelecidos os objetivos do sistema.

O 5W1H é uma ferramenta usada para auxiliar o planejamento do desdobramento de ações. Por meio desta ferramenta, respostas às seguintes questões são definidas:

What: o que fazer? Who: quem é o responsável e quem é o beneficiado com a ação? Where: onde a ação deve ser feita? Why: por que a ação deve ser feita? When: quando a ação deve ser feita? How: como a ação deve ser realizada, quais as etapas e sua sequência? (FAESARELLA; SACOMANO; CARPINETTI, 2006, p. 88)

Cada questão problema pode ter diferentes possíveis soluções, de tal forma que se torna possível analisar e escolher um caminho a seguir.

A partir do referencial teórico estudado, foram elaboradas três questões problema e, para cada uma, três possíveis soluções, para assim escolher um dos caminhos a seguir. Nas Figuras 4, 5 e 6 estão apresentadas as três questões problema e a solução escolhida.

Sendo assim, os objetivos definidos com o auxílio do 5W1H foram:

- Construir um hidrômetro inteligente com um aplicativo de celular que consulta o consumo de água e o seu custo financeiro;
- Construir um aplicativo com uma seção para notícias e dicas sobre o consumo consciente de água e afins;
- Divulgar em uma seção do aplicativo os dados de consumo da cidade e de certas regiões, de forma anônima e com permissão do usuário.

O documento com todas as possíveis soluções está disponível no GITHUB, com acesso via link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

Figura 4: 5W1H - Proporcionar uma análise de consumo de água da residência e seu impacto financeiro

1º	Proporcionar uma análise de consumo de água da residência e seu impacto financeiro					
	What	Why	Where	When	Who	How
	Construir um hidrômetro inteligente com um aplicativo de celular que consulta o consumo de água e o seu custo financeiro	A marcação de consumo é feita automática e pode ser transformado em valor monetário (R\$)	Hidrômetro disponível inicialmente para a população do município de Franca SP; Aplicativo disponível nas plataformas Android e IOS	Início no primeiro semestre do ano de 2022 até o final do último semestre de 2023	Pedro H. S. Marçal e Vinicius G. P. Brito	O hidrômetro será construído com o ESP8266 e um sensor do fluxo de vazão; O consumo é marcado de hora em hora e enviado ao back-end; A conversão do consumo para R\$ é feito com a tabela de preços disponibilizada pela provedora de água; A consulta das informações é feita por um Aplicativo que proporciona gráficos do consumo e do impacto financeiro.

Fonte: OS AUTORES

Figura 5: 5W1H - Conscientizar a população dos perigos do gasto desnecessário da água

2º	Problema:	Conscientizar a população dos perigos do gasto desnecessário da água				
	What	Why	Where	When	Who	How
1	Construir um aplicativo com uma seção para notícias e dicas sobre o consumo consciente de água e afins	A qualquer momento o usuário pode acessar a partir de seu celular as mais novas atualizações sobre os perigos relacionados à água em sua cidade, incentivando a economia e controle do gasto de água	Aplicativo disponível nas plataformas Android e IOs	Início no primeiro semestre do ano de 2022 até o final do último semestre de 2023	Pedro H. S. Marçal e Vinicius G. P. Brito	Reunir diariamente as notícias e dicas referentes ao consumo de água; O cadastro das notícias será feito por um usuário administrador; Planejar e desenvolver um Aplicativo com uma seção de notícias que estão cadastradas no banco de dados.

Fonte: OS AUTORES

Figura 6: 5W1H - Divulgar para a população os dados de consumo de água da cidade e regiões, a fim de alertar sobre o excesso de gastos

3º	Problema:	Divulgar para a população os dados de consumo de água da cidade e regiões, a fim de alertar sobre o excesso de gastos				
	What	Why	Where	When	Who	How
1	Divulgar através de uma seção do aplicativo os dados de consumo da cidade e de certas regiões, de forma anônima e com permissão do usuário	Com esses dados será possível elaborar diversos tipos de relatórios diferentes para a população que poderão ser acessados a qualquer momento	O relatório é emitido a partir do banco de dados e divulgado no app que estará disponível nas plataformas Android e IOs	Início no primeiro semestre do ano de 2022 até o final do último semestre de 2023	Pedro H. S. Marçal e Vinicius G. P. Brito	A qualquer momento o usuário poderá acessar uma sessão do aplicativo que vai apresentar gráficos sobre consumo de água na cidade (apenas dos usuários que permitirem a divulgação dos dados de forma anônima)

Fonte: OS AUTORES

O documento com todas as possíveis soluções está disponível no GITHUB, com acesso via link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

No decorrer do projeto, outras técnicas e documentos foram utilizados para definir e detalhar melhor os requisitos de acordo com esses objetivos, e serão melhor apresentados nas próximas seções.

4.2. Documento de Requisitos

Com os objetivos definidos, foi possível iniciar a definição e detalhamento dos requisitos do projeto, por isso a próxima etapa foi desenvolver o Documento de Requisitos.

Os requisitos são separados em duas categorias diferentes, os Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF). De acordo com Sommerville (2011), os requisitos funcionais descrevem os serviços que o sistema deve oferecer e como ele deve se comportar em determinadas situações, já os requisitos não funcionais impõe limitações nas funções e serviços que a solução deve fornecer, tais como: restrições de tempo, de desenvolvimento e no uso de padrões, esses requisitos estão relacionados com toda a solução e acaba por desencadear na sua usabilidade, desempenho, segurança e demais aspectos importantes.

Como exemplo, na Tabela 1 é apresentado o Requisito Funcional Cadastrar Usuário Cliente.

Tabela 1: RF001 - Cadastrar Usuário Cliente

ID: RF001	Nome do Requisito: Cadastrar Usuário Cliente
Descrição ->	O sistema deverá ser capaz de fornecer o cadastro de usuários do tipo cliente.

Categoria: **Prioridades:** Essencial

Informações -> As informações do cliente são:

- Nome;
- Email;
- Senha;
- Foto;
- Tipo da conta como cliente.

Regras de Negócio -> Não deve existir um usuário com o mesmo email cadastrado;
A senha deve ter ao menos 8 caracteres, e um pelo menos 1 número e 1 letra.

Fonte: OS AUTORES

O documento completo está disponível no GITHUB, podendo ser acessado por meio do link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

Após finalizar o Documento de Requisitos, foram criados alguns diagramas para melhor entendimento do sistema e estão apresentados a seguir.

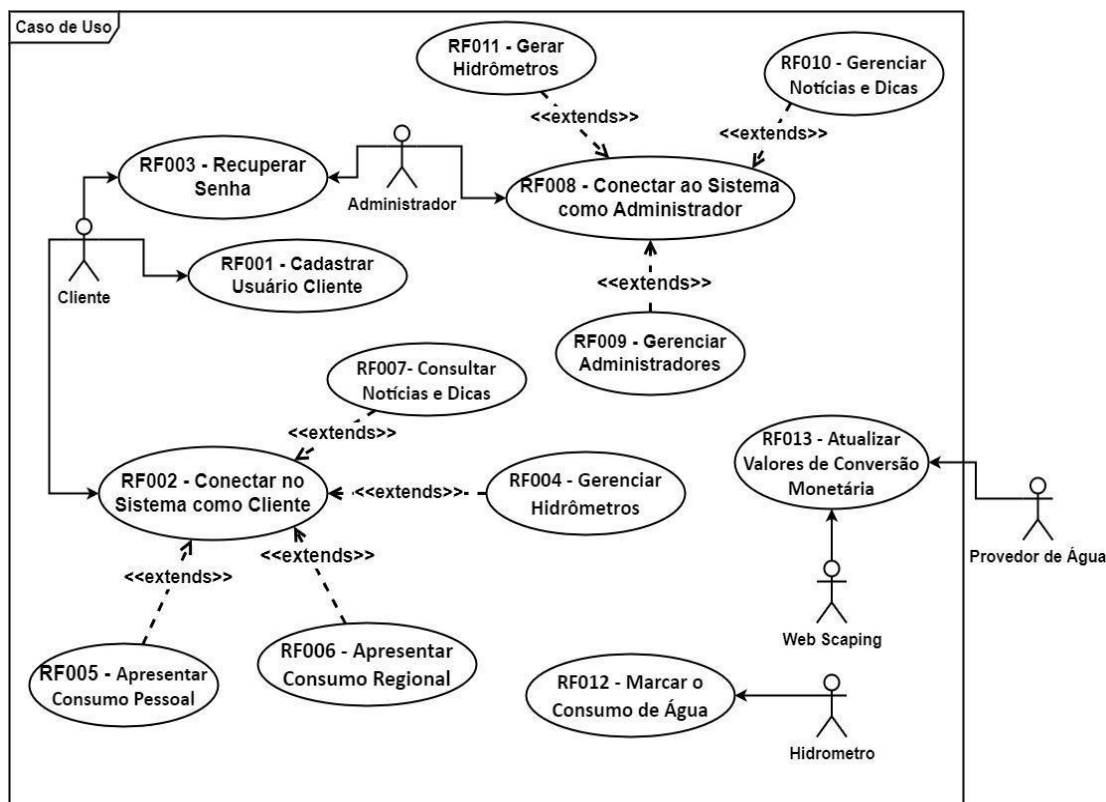
4.3. Caso de Uso

O Diagrama do Caso de Uso, de acordo com a UML (2005), “descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário”, isso utilizando de **Atores, Casos de Uso e Relacionamentos** entre eles.

Na Figura 7 é apresentado o Diagrama de Caso de Uso do projeto. Cada caso de uso apresentado na figura foi melhor detalhado, com os dados apresentados em forma de tabelas. Como exemplo, na Tabela 2 é apresentado o Caso de Uso Cadastrar Usuário Cliente.

O documento completo está disponível no GITHUB, com acesso via link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

Figura 7: Diagrama de Caso de Uso



Fonte: OS AUTORES

Tabela 2: UC001 - Cadastrar Usuário Cliente

Caso de Uso – Cadastrar Usuário Cliente	
ID	UC001
Descrição	Este caso de uso tem como objetivo permitir o cadastro do usuário cliente, de acordo com o RF001
Ator Primário	Cliente
Ator Secundário	Não possui
Pré Condição	Não possui
Cenário Principal	<ol style="list-style-type: none"> 1. Tudo se inicia quando o usuário clica em “Cadastrar-se” ao acessar o aplicativo 2. O sistema apresenta um formulário contendo Nome, Email, Senha e a opção de enviar uma Foto 3. O usuário preenche o formulário e envia 4. O sistema verifica se não existe um outro usuário cadastrado com o Email informado 5. O novo usuário é cadastrado 6. O usuário é redirecionado para a tela inicial
Cenário Alternativo	<p>*a. O usuário pode se desconectar do sistema a qualquer momento</p> <p>4a. O sistema alerta o usuário que o email informado já está</p>

cadastrado

Pós Condições O sistema permanece conectado na conta

Fonte: OS AUTORES

4.4. Matrizes de Rastreabilidade

Com o Documento de Requisitos pronto, junto das suas Regras de Negócio e do Caso de Uso, foi possível iniciar o desenvolvimento das Matrizes de Rastreabilidade.

Matriz de Rastreabilidade é uma representação tabular formada de artefatos nas linhas e colunas em que os cruzamentos indicam algum tipo de relacionamento muitos para muitos, estes artefatos podem ser requisitos, casos de uso, casos de teste e demais componentes de software (PINHEIRO, 2011, p. 27).

Foram desenvolvidas duas matrizes: a primeira relaciona os Requisitos (Funcionais e Não Funcionais) com as Regras de Negócio (Figura 8), enquanto a segunda relaciona os Requisitos (Funcionais e Não Funcionais) com os Casos de Uso (Figura 9).

Figura 8: Matriz de Rastreabilidade (Requisitos e Regras de Negócio)

Matriz de Rastreabilidade (Requisitos X Regras de Negócio)									
	RN001	RN002	RN003	RN004	RN005	RN006	RN007	RN008	RN009
RF001			X	X					
RF002		X							
RF003				X	X	X			
RF004		X					X		
RF005		X					X		
RF006		X						X	
RF007		X							
RF008	X								
RF009	X		X	X					
RF010	X								
RF011	X								
RF012									X

Fonte: OS AUTORES

Figura 9: Matriz de Rastreabilidade (Requisitos e Casos de Uso)

Matriz de Rastreabilidade (Requisitos X Caso de Uso)													
	UC001	UC002	UC003	UC004	UC005	UC006	UC007	UC008	UC009	UC010	UC011	UC012	UC013
RF001	X												
RF002		X											
RF003			X										
RF004				X									
RF005					X								
RF006						X							
RF007							X						
RF008								X					
RF009									X				
RF010										X			
RF011											X		
RF012												X	
RF013													X
RNF01	X	X	X	X	X	X	X	X	X	X	X	X	X
RNF02	X	X	X	X	X	X	X	X	X	X	X	X	X
RNF03	X	X	X	X	X	X	X	X	X	X	X	X	X
RNF04												X	
RNF05													X

Fonte: OS AUTORES

4.5. Modelagem do Banco de Dados

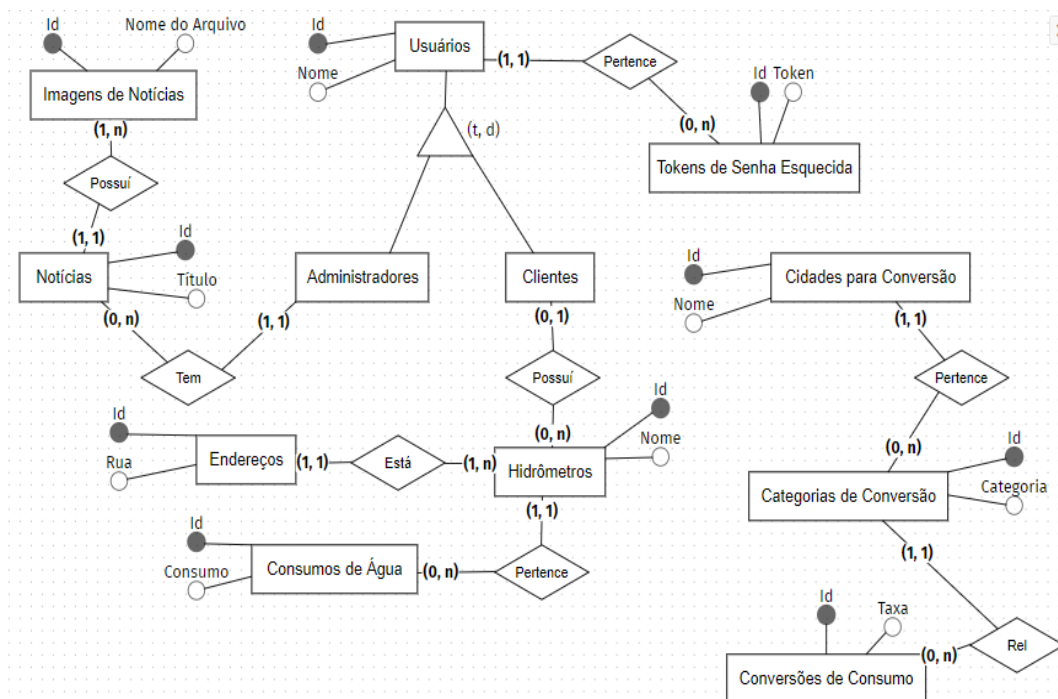
Os bancos de dados são ferramentas essenciais para o armazenamento e gerenciamento de informações em diversas áreas, desde sistemas empresariais até aplicações web. De acordo com Connolly e Begg (2014), um banco de dados é uma coleção de dados inter-relacionados, representando informações sobre um domínio específico.

De acordo com Elmasri e Navathe (2003), a eficiência no armazenamento e gerenciamento dos dados é fundamental para o desempenho de uma aplicação que utiliza um banco de dados. Portanto, é importante escolher o tipo de banco de dados mais adequado para a aplicação em questão, considerando aspectos como escalabilidade, segurança, desempenho e facilidade de manutenção.

Após a definição dos requisitos e as suas devidas documentações, o próximo passo foi modelar as tabelas do banco de dados. O banco de dados que foi utilizado no projeto é relacional, sendo assim, a modelagem conceitual foi construída no formato de Modelo Entidade Relacionamento (MER) que, segundo Puga (2014), “tem como base a perspectiva do mundo real como construído por um conjunto de objetos, chamados de entidades e relacionamentos”, trazendo consigo uma semântica maior.

Com a modelagem conceitual (Figura 10), é possível perceber que no sistema serão necessárias tabelas para armazenar: Os Usuários da solução (sendo eles Clientes ou Administradores); as Notícias cadastradas pelos Administradores e as suas respectivas Imagens; os Hidrômetros dos Clientes e o Endereço em que está instalado; os Consumos de Água marcados pelo Hidrômetro; as Cidades com tabelas de conversão do consumo em valor monetário, juntamente com as Categorias de Conversão e as suas Conversões (uma vez que cada cidade tem diferentes categorias e cada categoria tem diferentes regras de conversão); e, por fim, um local para salvar os Tokens de solicitações para recuperação de senha do usuário.

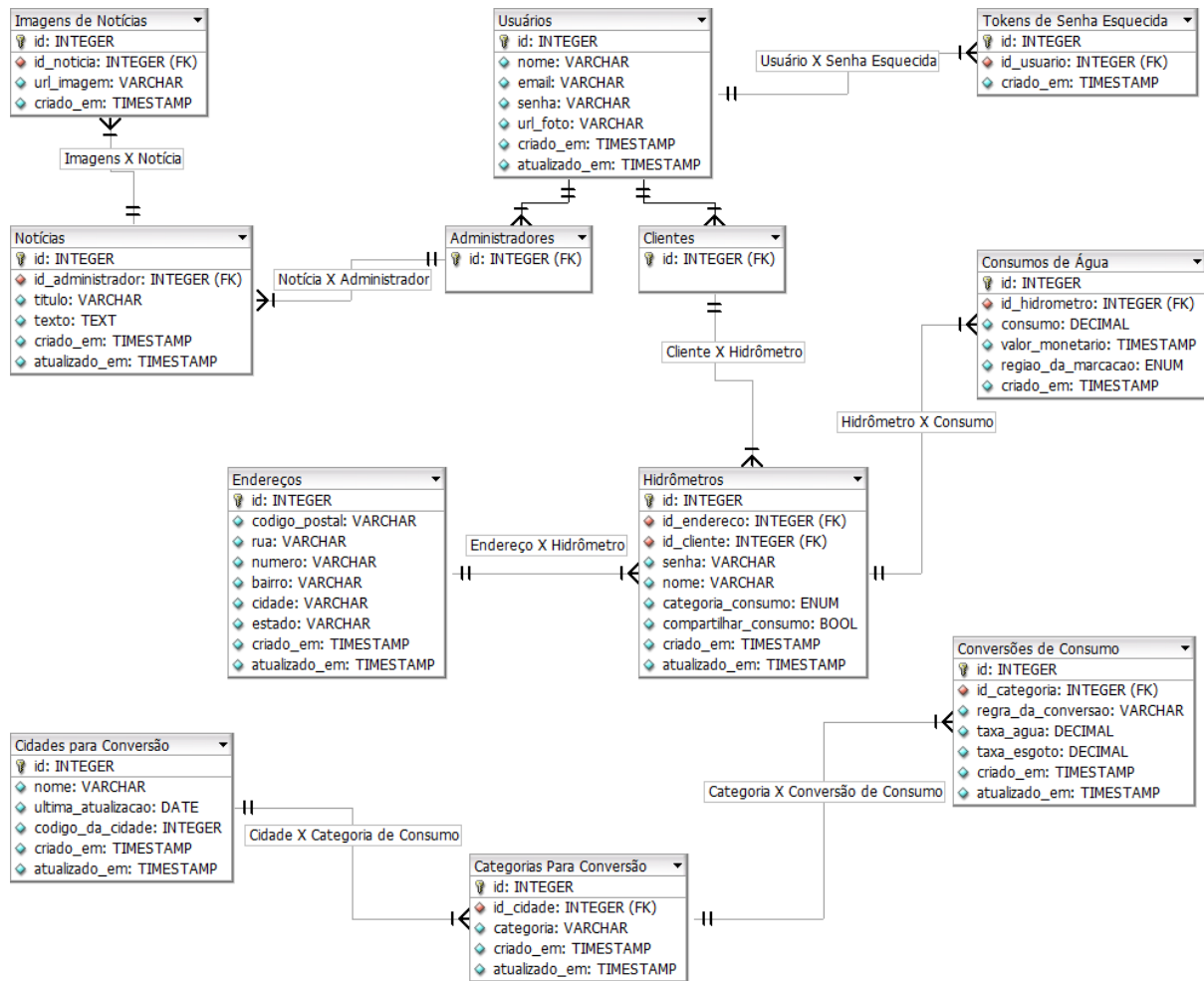
Figura 10: Modelo Entidade Relacionamento do Banco de Dados



Fonte: OS AUTORES

Em seguida, foi desenvolvida a versão lógica da modelagem (Figura 11) que, segundo Puga (2014) “reflete as propriedades necessárias para a tradução do modelo conceitual, de maneira que seja possível a descrição dos elementos capazes de serem interpretados por SGBD (Sistema de Gerenciamento de Banco de Dados)”.

Figura 11: Modelo Lógico do Banco de Dados



Fonte: OS AUTORES

Depois de definir a modelagem conceitual e a modelagem lógica, foi construída a modelagem física do banco (Figura 12), na qual “são descritas as estruturas físicas do banco de dados e, por isso, deve-se considerar o SGBD que será utilizado para a sua implementação” (Puga, França e Goya, 2014). Para o projeto, foi utilizado o banco de dados PostgreSQL, por ser uma solução livre, de código aberto e eficiente, em conjunto com o Prisma ORM, pois é uma ferramenta que facilita a conexão com o banco de dados e sua construção através do NodeJS.

Na Figura 12 pode-se observar que, através do Prisma ORM, a conexão com o PostgreSQL é realizada e a tabela intitulada **Users** (os usuários do sistema) é criada com as suas respectivas colunas, além de fazer uma associação com a tabela **ForgotPasswordTokens** (os tokens de senha esquecida para a recuperação).

Figura 12: Início do Modelo Físico do Banco de Dados

```

1  generator client {
2    provider = "prisma-client-js"
3  }
4
5  datasource db {
6    provider = "postgresql"
7    url      = env("DATABASE_URL")
8  }
9
10 enum AccountType {
11   CUSTOMER
12   ADMIN
13 }
14
15 model Users {
16   id          String      @id @default(uuid())
17   name        String
18   email       String
19   password    String
20   avatar_file String?
21   account_type AccountType @default(CUSTOMER)
22   created_at  DateTime    @default(now())
23   updated_at  DateTime    @default(now())
24
25   forgotPasswordToken ForgotPasswordTokens?
26
27   @@map("users")
28 }

```

Fonte: OS AUTORES

5. Desenvolvimento Backend

Para o desenvolvimento do *backend* foi utilizado o NodeJS.

Node.js é um interpretador de código JavaScript que funciona do lado do servidor. Seu objetivo é ajudar programadores na criação de aplicações de alta escalabilidade (como um servidor web), com códigos capazes de manipular dezenas de milhares de conexões simultâneas, numa única máquina física (PAULINO, 2018, p. 1).

Para a programação, além de trabalhar com o NodeJS, foi utilizada a linguagem Typescript, que segundo Typescript Lang (2012), “é uma linguagem de programação fortemente tipada que se baseia em JavaScript, oferecendo melhores ferramentas em qualquer escala.”

O objetivo principal do *backend* foi fornecer uma API RESTful para armazenar os dados da aplicação, bem como o controle de acesso e manipulação das informações no banco de dados.

Foi escolhido o NodeJS com TypeScript devido à facilidade de desenvolvimento e a vasta quantidade de bibliotecas disponíveis que auxiliam a programação.

No código apresentado na Figura 13, a classe **SeeNewsUseCase** é responsável por consultar no banco de dados as informações de uma determinada notícia, utilizado quando o usuário do tipo **cliente** deseja visualizar a notícia.

Figura 13: Backend - Código para Buscar Uma Notícia

```

import { injectable, inject } from 'tsyringe'
import { INewsRepository } from '@modules/news/repos'
import { AppError } from '@shared/errors/AppError'
import { HTTP_STATUS_CODE } from '@utils/constants'

interface IUseCaseProps {
  id: string
}

const NEWS_NOT_FOUND = 'News not found!'

@injectable()
export class SeeNewsUseCase {
  constructor (
    @inject('NewsRepository')
    private newsRepository: INewsRepository
  ) {}

  public async execute({ id }: IUseCaseProps) {
    const newsData = await this.newsRepository.findById(id)
    if (!newsData) {
      throw new AppError(NEWS_NOT_FOUND, HTTP_STATUS_CODE.NOT_FOUND)
    }
    return newsData
  }
}

```

Fonte: OS AUTORES

O código fonte completo da implementação *backend* está disponível no

GITHUB, com acesso via link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

6. Desenvolvimento Mobile

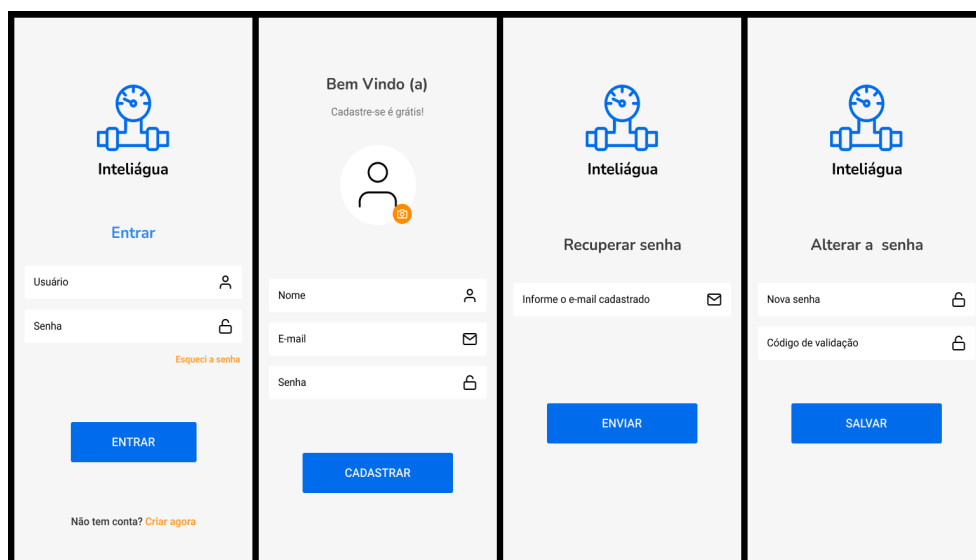
O desenvolvimento *mobile*, ou seja, do aplicativo, foi dividido em duas fases. A primeira fase foi criar o protótipo das telas e em seguida fazer a implementação por meio da programação. Ambas as etapas são descritas a seguir.

6.1. Protótipo das Telas do Aplicativo

Para iniciar a prototipação, as telas foram separadas em três conjuntos diferentes, sendo eles: Antes da Autenticação; Autenticado como Administrador; Autenticado como Cliente.

As telas do conjunto **Antes da Autenticação** contemplam a navegação que qualquer usuário pode fazer sem estar autenticado no sistema. Na Figura 14 são apresentadas as telas desse conjunto, que são: a de Autenticação, a de Cadastro de Cliente, a de Recuperação de Senha e a de Alteração de Senha, respectivamente.

Figura 14: Telas Antes da Autenticação

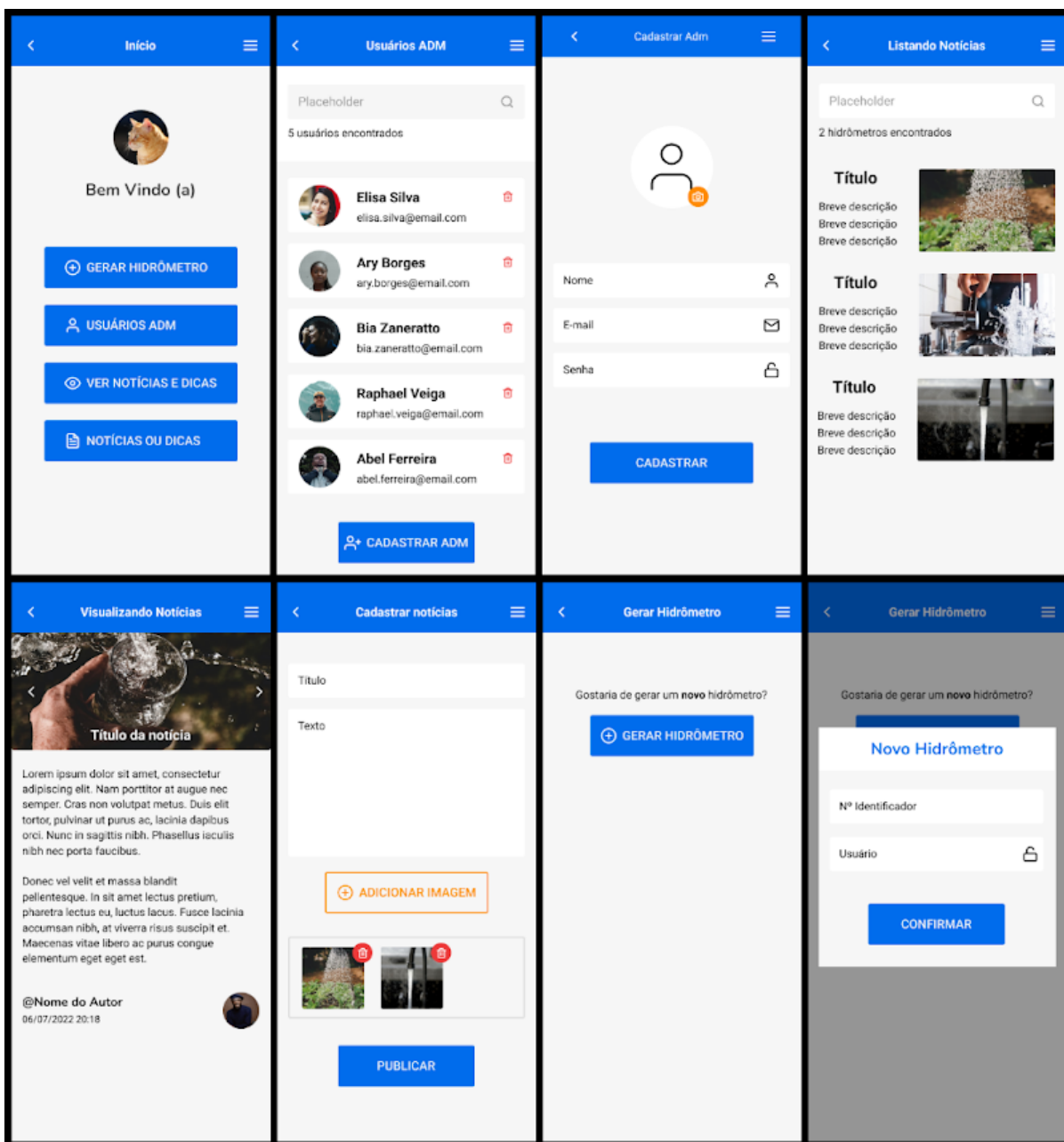


Fonte: OS AUTORES

Já o conjunto **Autenticado como Administrador** abrange as telas que os usuários administradores têm acesso após autenticados. Como apresentado na Figura 15, as telas deste conjunto são compostas pela Tela Inicial, a Listagem dos Administradores, o Cadastro de Administrador, a Listagem de Notícias, a Visualização de Notícia, o Cadastro de Notícia, Gerar Hidrômetro, o Popup para o Hidrômetro Gerado, respectivamente.

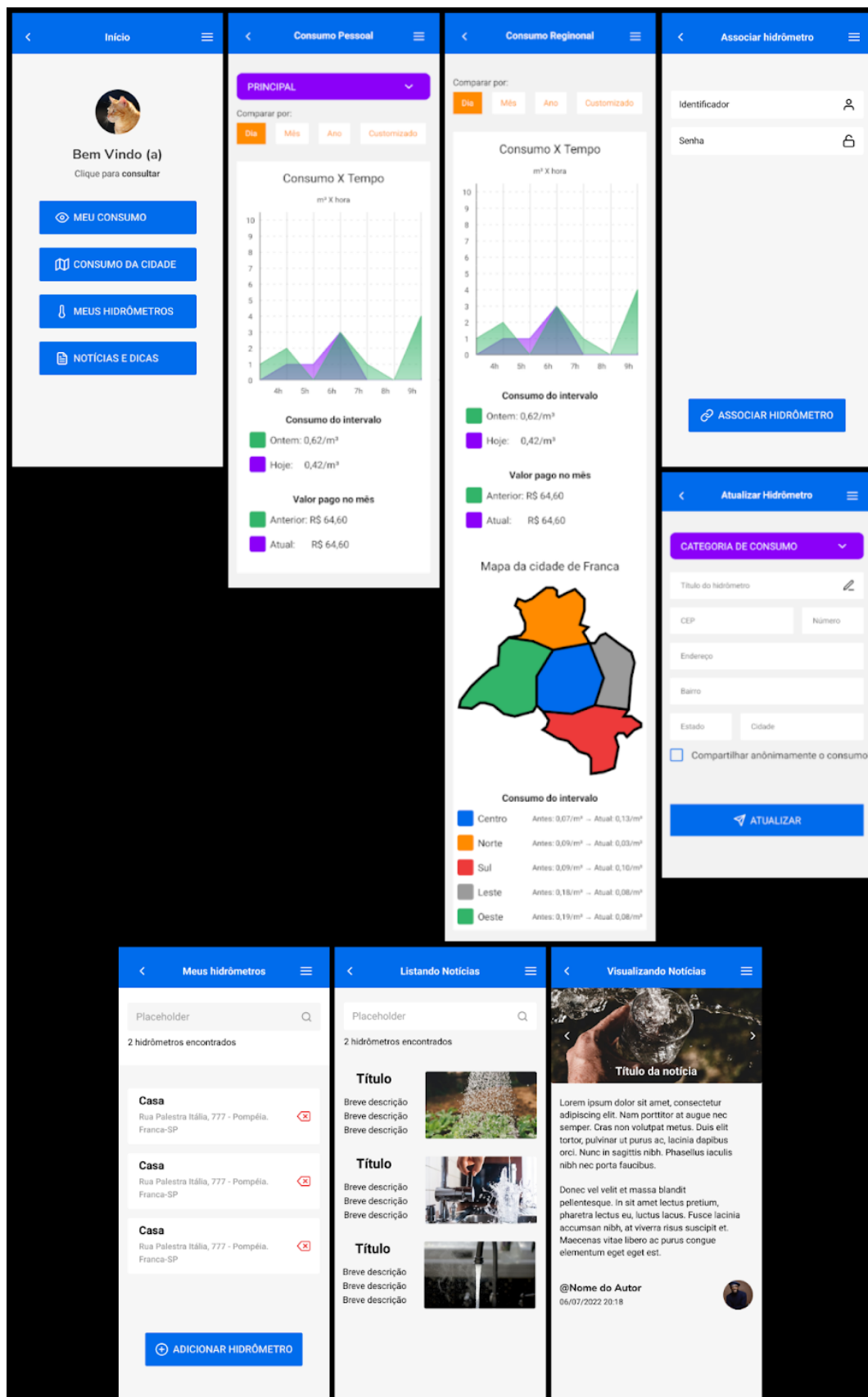
Por fim, o conjunto de telas **Autenticado como Cliente** aglomera as telas que os usuários clientes têm acesso depois de se autenticar. A Figura 16 apresenta esse conjunto de telas, com a Tela Inicial, o Consumo Pessoal, o Consumo Regional, Associar um Novo Hidrômetro e Configurar o Hidrômetro Associado, a Listagem de Hidrômetros do Cliente, a Listagem de Notícias e Visualização de Notícia, respectivamente.

Figura 15: Telas Autenticado como Administrador



Fonte: OS AUTORES

Figura 16: Telas Autenticado como Cliente



Fonte: OS AUTORES

6.2. Implementação das Telas

Com as telas prototipadas, já foi possível iniciar a implementação delas. Para isso, foi utilizado o React Native, que é uma poderosa biblioteca Javascript voltada à criação de aplicativos para Android e iOS, e com base em conceitos e tecnologias do desenvolvimento *web*, essa biblioteca oferece uma curva de aprendizado acessível para programadores familiarizados com o ambiente *web*, além de ser amplamente reconhecida como uma das ferramentas mais populares e amplamente adotadas no mercado atualmente (STATISTA, 2022).

O React Native foi escolhido por conta da facilidade de implementação, uma vez que se assemelha com o desenvolvimento *web*, da possibilidade de gerar a distribuição do aplicativo para dispositivos Android e iOS a partir de um mesmo código fonte, e da grande quantidade de bibliotecas que agiliza o desenvolvimento. O Typescript também foi utilizado junto do React Native para a programação.

Na Figura 17 é apresentado o trecho de código responsável por exibir um componente de botão na tela do aplicativo. Esse componente é utilizado para executar uma função quando o usuário clica sobre ele. Além disso, por ser um componente, ele pode ser reutilizado em diferentes telas.

Figura 17: *Mobile* - Código do Componente de Botão

```
import React from 'react';
import { RectButtonProperties } from 'react-native-gesture-handler';

import { Container, ButtonText } from './styles';

interface ButtonProps extends RectButtonProperties {
  children: string;
}

const Button: React.FC<ButtonProps> = ({ children, ...rest }) => {
  return (
    <Container {...rest}>
      <ButtonText>{children}</ButtonText>
    </Container>
  );
}

export default Button;
```

Fonte: OS AUTORES

O código fonte completo da implementação *mobile* está disponível no GITHUB, com acesso via link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

7. Hidrômetro Inteligente

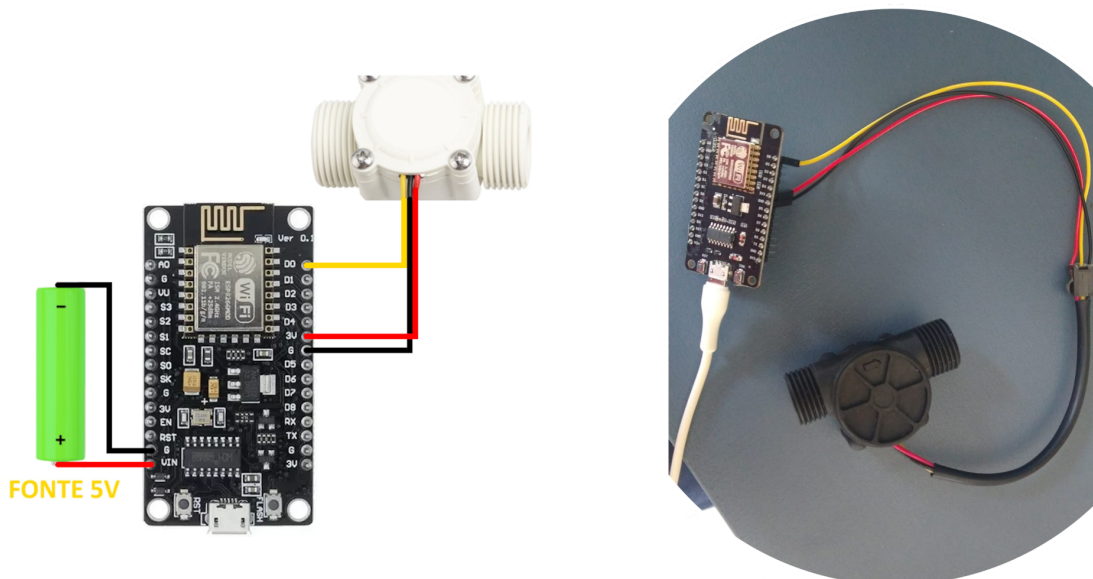
Após a implementação do aplicativo, o próximo passo foi projetar o esquema elétrico do hidrômetro inteligente, e para isso foram utilizados três componentes principais: o ESP8266 NodeMCU, um sensor de fluxo e uma fonte de energia de 5 volts. Além desses componentes, também seriam necessários alguns fios para realizar as conexões entre eles.

De acordo com Sena (2018), o NodeMCU é um microcontrolador muito utilizado para a implementação de soluções IoT, uma vez que já tem o módulo de conexão Wi-Fi nativo, tendo sido esta a característica da escolha para o projeto pois

é ele quem vai calcular o consumo captado pelo sensor de fluxo e enviar para o sistema a cada hora via conexão Wi-Fi. O NodeMCU é alimentado pela fonte de 5 volts.

A Figura 18 apresenta o esquema elétrico do hidrômetro inteligente e o hidrômetro já construído com base no esquema elétrico.

Figura 18: Esquema Elétrico e o Hidrômetro Inteligente



Fonte: OS AUTORES

Com o hidrômetro pronto, o passo seguinte foi programar as funcionalidades de coletar o consumo, realizar as conversões e enviar as informações para o *backend* com base no protocolo MQTT.

Para a programação foi utilizado o *Arduino IDE*, pois, de acordo com Graça (2017), ele já oferece um ambiente e uma linguagem de programação com vários recursos e bibliotecas úteis para o desenvolvimento no NodeMCU.

Na Figura 19 é apresentada uma parte do código do hidrômetro, em que existem duas funções: a ***calculateFlow***, que é responsável por calcular o consumo com base na quantidade de pulsos emitidos pelo sensor de fluxo e pelo intervalo de uma hora, convertendo o valor para metros cúbicos, e a ***sendDataToMQTT***, que é responsável por enviar o valor do consumo para o *backend* por meio do MQTT.

O código fonte completo da implementação do hidrômetro está disponível no GITHUB, com acesso via link: <https://github.com/Vini7Dev/tcc-smart-hydrometer>.

Figura 19: Parte do Código do Hidrômetro Inteligente

```
void calculateFlow() {  
    double totalFlow = (count * FLOW_PER_PULSE);  
    totalFlow = totalFlow * SECONDS_PER_MINUTE;  
    totalFlow = totalFlow * MINUTES_PER_HOUR;  
    totalFlow = totalFlow / MILLILITER_PER_LITER;  
    totalFlow = totalFlow * METERS_PER_LITER;  
  
    flowRate = totalFlow / (elapsedTime / 3600000.0);  
  
    Serial.println(flowRate);  
}  
  
void sendDataToMQTT() {  
    if (!client.connected()) {  
        reconnect();  
    }  
    client.loop();  
  
    char msg[10];  
    sprintf(msg, "%.2f", flowRate);  
    client.publish("createConsumptionMarking", msg);  
    delay(5000);  
}
```

Fonte: OS AUTORES

8. Conclusão

A água é um recurso fundamental para a vida no planeta, e o seu desperdício é um mal frequente na nossa sociedade. Ao perceber esse problema, os autores propuseram uma solução com o objetivo geral de ajudar o usuário a controlar o seu consumo de água, e com os objetivos específicos de incentivar a economia financeira para quando for pagar a conta da concessionária, e também conscientizar as pessoas a adotar boas práticas para a economia da água.

Assim, foi desenvolvido um protótipo físico, semelhante a um hidrômetro, que é responsável por coletar o consumo de água em um ambiente (por exemplo, uma casa) e enviar para o sistema por meio de uma conexão Wi-Fi, além de um aplicativo para consultar o consumo. O protótipo do aplicativo também inclui outras funcionalidades, como a conversão do consumo em valores monetários, a visualização do consumo na região e a conscientização por meio de notícias e dicas publicadas no aplicativo.

A partir de técnicas e documentações da Engenharia de Software, de conhecimentos adquiridos em programação, hardware e IoT, os objetivos propostos foram atingidos com êxito, oferecendo uma proposta exequível a nível de mercado para a população geral.

Um grande desafio do projeto foi conciliar diferentes áreas do desenvolvimento, para implementar o *backend*, o *mobile*, a programação do *hardware* e a sua prototipação e construção. Isso exigiu alguns conhecimentos que os autores não eram tão familiarizados, porém, tiveram a oportunidade de adquirir novas habilidades.

Pensando no futuro, os autores propõem algumas ideias para evoluir a solução, sendo elas: o atendimento a outras cidades além de Franca-SP; adicionar uma comparação de consumo entre dois ou mais hidrômetros do usuário; além do consumo de água, também oferecer um coletor para o consumo de energia elétrica.

Referências

- AL-FUQAHA, A.; GUIXANI, M.; MOHAMMADI, M.; ALEDHARI, M.; AYYASH, M. *Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications*. IEEE Communications Surveys & Tutorials, v. 17, n. 4, p. 2347-2376, Fourthquarter 2015, doi: 10.1109/COMST.2015.2444095. Disponível em: <https://ieeexplore.ieee.org/document/7123563>. Acesso em: 05 mai. 2023
- ATZORI, Luigi; LERA, A.; MORABITO, G. *The Internet of Things: A survey*. *Computer Networks*, v. 54, n. 15, p. 2787-2805, 2010. Disponível em:

<https://www.sciencedirect.com/science/article/abs/pii/S1389128610001568>.
Acesso em: 05 mai. 2023.

BARROS, E; CAVALCANTE, S. *Introdução aos Sistemas Embarcados*. Universidade Federal de Pernambuco, 2010. Disponível em:
<https://www.cin.ufpe.br/~vba/periodos/8th/s.e/aulas/STP%20-%20Intro%20Sist%20Embarcados.pdf>. Acesso em: 05 mai. 2023

BATISTA, C. *A Importância da Água*. Toda Matéria, 2022. Disponível em:
<https://www.todamateria.com.br/a-importancia-da-agua/>. Acesso em: 05 mai. 2023.

BEGG, C.; CANOLLY, T. *Database Systems: A Practical Approach to Design, Implementation, and Management*. 6. ed. Londres: Pearson Education Limited, 2014.

BRASIL chega aos 40% de perdas de água potável. *Trata Brasil*, 2022. Disponível em: <https://tratabrasil.org.br/brasil-chega-aos-40-de-perdas-de-agua-potavel/>. Acesso em: 05 mai. 2023.

CAUSAS e consequências do desperdício de água. Hexag Educação, 2022. Disponível em:
<https://cursinhoparamedicina.com.br/blog/atualidades/causas-e-consequencia-do-desperdicio-de-agua/>. Acesso em: 05 mai. 2023.

CHASE, O. *Sistemas Embarcados*. Max Pezzin, 2007. Disponível em:
http://www.maxpezzin.com.br/aulas/6_EAC_Sistemas_Embarcados/1_SE_Introducao.pdf. Acesso em: 05 mai. 2023.

CIDADES brasileiras desperdiçam 40% da água disponível por falta de cuidado ou mau uso. G1, 2022. Disponível em:
<https://g1.globo.com/jornal-nacional/noticia/2022/06/01/cidades-brasileiras-de-sperdicam-40percent-da-agua-disponivel-por-falta-de-cuidado-ou-mau-uso.gh.html>. Acesso em: 05 mai. 2023.

DECLARAÇÃO Universal dos Direitos da Água. Faculdade de Saúde Pública da Universidade de São Paulo, São Paulo, 2017. Disponível em:
http://www.cecol.fsp.usp.br/dcms/uploads/arquivos/1483371864_UNU-Declara%C3%A7%C3%A3o%20Universal%20dos%20Direitos%20da%20%C3%81gua.pdf. Acesso em: 05 mai. 2023.

ELMASRI, R.; NAVATHE, S. B. *Fundamentals of Database Systems*. 4. ed. Londres: Pearson, 2003.

FAESARELLA, I. S.; SACOMANO, J. B.; CARPINETTI, L. C. R. *Gestão da Qualidade: Conceitos e Ferramentas*. Escola de Engenharia de São Carlos da Universidade de São Paulo, São Paulo, 2006. Disponível em:
http://repositorio.eesc.usp.br/bitstream/handle/RIEESC/6212/Faesarellalvete_GestaoQualidade.pdf?sequence=1. Acesso em: 05 mai. 2023.

FALTA de pagamento de contas de água e luz bate recorde em março, aponta Serasa. *Correio do Povo*, 2022. Disponível em:

<https://www.correiadopovo.com.br/not%C3%ADcias/economia/falta-de-pagamento-de-contas-de-%C3%A1gua-e-luz-bate-recorde-em-mar%C3%A7o-aponta-serasa-1.812954>. Acesso em: 05 mai. 2023.

GRAÇA, P. *Sistema de aquisição de dados utilizando o módulo ESP8266 NodeMCU*. Universidade Estadual Paulista, São Paulo, 2017. Disponível em: <https://repositorio.unesp.br/handle/11449/156909>. Acesso em: 01 jul. 2023

GUBBI, J.; BUYYA, R.; MARUSIC, S.; PALANISWAMI, M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, v. 29, n. 7, p. 1645-1660, 2013. DOI: <https://doi.org/10.1016/j.future.2013.01.010>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0167739X13000241>. Acesso em: 05 maio 2023.

JUCÁ, S.; PEREIRA, R.. *Aplicações práticas de sistemas embarcados Linux utilizando Raspberry Pi*. 1. ed. Rio de Janeiro: PoD Editora, 2018.

KONCAGÜL, E.; CONNOR, R; STEFAN, U. Relatório Mundial das Nações Unidas sobre o Desenvolvimento dos Recursos Hídricos 2019: Não deixar ninguém para trás, resumo executivo. UNESCO World Water Assessment Programme, 2019. Disponível em: https://unesdoc.unesco.org/ark:/48223/pf0000367303_por. Acesso em: 05 mai. 2023.

MARTINS, I. R.; ZEM, J. L. Estudo dos protocolos de comunicação MQTT e COaP para aplicações machine-to-machine e Internet das coisas. *Revista Tecnológica da Fatec Americana, Americana*. v.3, n.1, p.64-87, 2015. Disponível em: http://www.fatec.edu.br/revista_ojs/index.php/RTecFatecAM/article/view/41/50. Acesso em 20 de maio de 2016.

MIERZWA, J. C.; HESPANHOL, I. *Água na Indústria: uso racional e reúso*. 1. ed. São Paulo: Oficina de Textos, 2002. https://books.google.com.br/books?hl=pt-BR&lr=&id=V1iXBAAQBAJ&oi=fnd&pg=PT10&dq=consumo+de+%C3%A1gua+na+industria&ots=ipgXTMgbdw&sig=Xh9NDyP2aSG-V_kJx0X7vdKRU_Q#v=onepage&q=consumo%20de%20%C3%A1gua%20na%20industria&f=false

PAULINO, L. S. NODEJS. *Revista de Gestão Educação e Tecnologia*, Gravataí, v. 4, n. 2, p. 3. dez. 2018. Disponível em: <https://refaqi.faqi.edu.br/index.php/refaqi/article/view/104>. Acesso em: 27 mai. 2023.

PFLEEGER, S. L.; ATLEE, J. M. *Software Engineering: Theory and Practice*. 4. ed. Hoboken: Prentice Hall, 2009.

PINHEIRO, M. C. *Rastreabilidade de Requisitos com o Enterprise Architect e RaQuest*. Universidade Federal de Minas Gerais, Belo Horizonte, 2011. Disponível em: <https://repositorio.ufmg.br/handle/1843/49189>. Acesso em: 05 mai. 2023.

PRESSMAN, R. S.; MAXIM, B. R. Engenharia de software: uma abordagem profissional. Tradução de Reginaldo Arakaki Julio Arakaki e Renato Manzan de Andrade. 8. ed. Porto Alegre: AMGH, 2016.

PUGA, S.; FRANÇA, E.; GOYA, M. Banco de Dados: Implementação em SQL, PL/SQL e Oracle 11g. 1. ed. São Paulo: Pearson, 2014. Disponível em: <https://plataforma.bvirtual.com.br/Leitor/Publicacao/3842/pdf/0>. Acesso em: 05 mai. 2023.

SABESP encerra rodízio de água em Franca, SP, após mais de 1 mês de racionamento. G1, 2021. Disponível em: <https://g1.globo.com/sp/ribeirao-preto-franca/noticia/2021/10/11/sabesp-encerra-rodizio-de-agua-em-franca-sp-apos-mais-de-1-mes-de-acionamento.gh.html>. Acesso em: 05 mai. 2023.

SENA, G. *Medidor de consumo de energia elétrica com acesso local e remoto usando plataforma ESP8266*, p. 113. Universidade Federal do Pampa, Alegrete, 2018. Disponível em: <https://dspace.unipampa.edu.br/handle/riu/3305>. Acesso em: 01 jul. 2023

SOMMERVILLE, I. Engenharia de software. Tradução de Luiz Claudio Queiroz. 10. ed. São Paulo: Pearson Addison-Wesley, 2011.

STATISTA. Cross-platform mobile frameworks used by software developers worldwide from 2019 to 2021. Universidade Federal de Ceará, Fortaleza, 2022. Disponível em: <https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/>. Acesso em: 27 mai. 2023

TYPESCRIPT. Typescript Lang, 2012. Disponível em: <https://www.typescriptlang.org/pt/>. Acesso em: 27 mai. 2023.

UML. Use Cases. Departamento de Sistemas e Computação da Universidade Federal de Campina Grande, Campina Grande, 2005. Disponível em: <http://www.dsc.ufcg.edu.br/~jacques/cursos/map/html/uml/diagramas/usecases/usecases.htm>. Acesso em: 05 mai. 2023.

VERIATO, M. K. L.; BARROS, H. M. M.; SOUZA, L. P.; CHICÓ, L. R.; BAROSI, K. X. L. Água: Escassez, crise e perspectivas para 2050. Revista Verde de Agroecologia e Desenvolvimento Sustentável, v. 10, n. 5, p. 17–22, 2015. DOI: 10.18378/rvads.v10i5.3869. Disponível em: <https://www.gvaa.com.br/revista/index.php/RVADS/article/view/3869>. Acesso em: 5 mai. 2023."