

Digital twins: estratégias para redução de tempo de resposta utilizando computação em borda

Digital twins: strategies for reducing response time using edging computing

Engenharia de Computação

Fábio Henrique Cabrini (pro7799@cefsa.edu.br)

Doutor em Engenharia Elétrica pela Universidade de São Paulo (USP) e professor da Faculdade Engenheiro Salvador Arena e da FIAP

Caio Rodrigues Fernandes Santos (caio.2003.rfs@gmail.com)

Bacharel em Engenharia de Computação pela Faculdade Engenheiro Salvador Arena

Carlos Eduardo Vieira Santos (cadu.vs@hotmail.com)

Bacharel em Engenharia de Computação pela Faculdade Engenheiro Salvador Arena

Guilherme Dias Lima Turtera (081200007@faculdade.cefsa.edu.br)

Bacharel em Engenharia de Computação pela Faculdade Engenheiro Salvador Arena

Nathan Vilela de Souza (nathanvilela18@gmail.com)

Bacharel em Engenharia de Computação pela Faculdade Engenheiro Salvador Arena

FTT Journal of Engineering and Business

• SÃO BERNARDO DO CAMPO, SP JUN. 2025

• ISSN 2525-8729

Submissão: 13 dez. 2024 Aceitação: 28 maio 2025

Sistema de avaliação: às cegas dupla (double blind review)

FACULDADE ENGENHEIRO SALVADOR ARENA, p. 56 - 75

FTT JOURNAL
of Engineering and Business



Resumo

Este estudo investiga estratégias para reduzir o tempo de resposta na arquitetura de comunicação de uma prova de conceito de gêmeos digitais, desenvolvida para sincronizar um braço robótico com sua versão virtual. O projeto original foi reproduzido de forma fidedigna, e observou-se que ele apresentava um tempo de resposta elevado durante a comunicação entre os gêmeos digitais, limitando sua aplicabilidade em contextos nos quais a velocidade da resposta é um fator determinante. Para mitigar esse desafio, foram implementadas duas modificações: a reformulação do fluxo de comunicação entre os gêmeos, reduzindo atrasos na transmissão, e a realocação do servidor da nuvem para a borda da rede. Os testes seguiram duas abordagens distintas: uma quantitativa, utilizando o tempo de ida e volta como métrica principal, e uma qualitativa, com a aplicação de um teste A/B para medir a percepção dos usuários quanto às melhorias. Os resultados indicaram que a arquitetura baseada em borda proporciona tempos de resposta significativamente mais rápidos, demonstrando o impacto positivo da localização do servidor na eficiência do sistema. Assim, o trabalho contribui para a crescente pesquisa sobre computação de borda, abrindo caminho para aplicações mais responsivas da Internet Tátil, especificamente em sistemas de gêmeos digitais.

Palavras-chave: Gêmeos Digitais, Computação em Borda. Tempo de Resposta, Internet Tátil.

Abstract

This study investigates strategies to reduce response time in the communication architecture of a digital twin proof of concept, developed to synchronize a robotic arm with its virtual version. The original project was faithfully reproduced, and it was observed that it exhibited a high response time during communication between the digital twins, limiting its applicability in contexts where response speed is a critical factor. To address this challenge, two modifications were implemented: a restructuring of the communication flow between the twins to reduce transmission delays, and the relocation of the server from the cloud to the network edge. The tests followed two distinct approaches: a quantitative approach, using round-trip time as the primary metric, and a qualitative approach, applying an A/B test to assess user perception of the improvements. The results indicate that the edge-based architecture provides significantly faster response times, highlighting the positive impact of server location on system efficiency. Thus, this work contributes to the growing research on edge computing, paving the way for more responsive applications of the Tactile Internet, specifically in digital twin systems.

Keywords Digital Twins. Edge Computing. Response Time. Tactile Internet.

Introdução

A crescente tendência à transformação digital tem reconfigurado o panorama das indústrias, causando mudanças profundas e permanentes. A internet e a digitalização de processos têm se consolidado como catalisadores centrais para inovações nas áreas de manufatura e produção. Mais do que simplesmente aprimorar operações já existentes, essas tecnologias pavimentam o caminho para novos métodos de produção e estratégias de gestão industrial. Esse movimento é significativo a ponto de ser considerado a quarta Revolução Industrial, caracterizada principalmente pelo advento das fábricas inteligentes (Moraes, 2020).

Essa nova fase, também conhecida como Indústria 4.0, é impulsionada por avanços tecnológicos em múltiplos setores, em especial nas áreas de Engenharia e Tecnologia da Informação. Entre as principais inovações que sustentam essa revolução, destaca-se a Internet das Coisas (IoT, do inglês *Internet of Things*), que possibilita que dispositivos atuem de forma autônoma e descentralizada, aumentando a eficiência de processos produtivos e gerando dados para embasar decisões estratégicas da gestão (Sátyro *et al.*, 2018).

Nesse contexto de fusão entre o digital e o físico, surge o conceito de gêmeos digitais (DT, do inglês *Digital Twins*): réplicas de elementos físicos que simulam suas condições e comportamentos de forma simultânea. Com essa tecnologia, empresas podem antecipar problemas e experimentar novas ideias e estratégias em um ambiente controlado e seguro, sem interromper os processos existentes e reduzindo custos materiais decorrentes desse processo (Tao *et al.*, 2019).

Embora apresentem um potencial transformador, muitas dessas aplicações possuem requisitos rigorosos de tempo de resposta, especialmente em cenários como fábricas inteligentes, cirurgia robótica e operações médicas remotas. Nesses cenários, atrasos podem comprometer a segurança, a confiança e a eficiência das operações. Para mitigar esses desafios, surgiram movimentos como a Internet Tátil, que visa minimizar a latência por meio de tecnologias como a computação em borda (do inglês, *edge computing*). Assim, busca-se aprimorar a confiabilidade e a reatividade de sistemas críticos em diferentes setores (Almasan *et al.*, 2022; Huynh *et al.*, 2022; Ma *et al.*, 2019).

Nesse contexto, o presente estudo aborda o trabalho realizado por Rodrigues *et al.* (2024), que desenvolveram uma prova de conceito (PoC, do inglês *Proof of Concept*) de gêmeos digitais, com o

objetivo de estabelecer um sistema de comunicação bidirecional entre um braço robótico e um modelo computacional tridimensional capaz de simular seus movimentos. Apesar de essa meta ter sido alcançada, foi identificado um atraso visível na comunicação entre os gêmeos, evidenciando a inviabilidade do sincronismo desejado para esse tipo de aplicação.

Assim, o objetivo deste trabalho foi analisar propostas de alteração da PoC que têm potencial de reduzir o tempo de resposta. Foram analisados os impactos decorrentes da implementação de um novo fluxo de comunicação entre gêmeos digitais, aliada à mudança de localidade do servidor no qual essa comunicação se baseia para a borda da rede, em substituição à sua posição original na nuvem.

Para alcançar tal propósito, foi seguida uma sequência estruturada de objetivos específicos que culminou no resultado deste estudo. Inicialmente, a PoC de Rodrigues *et al.* (2024) foi reproduzida de modo a assegurar a fidelidade do ambiente experimental. Posteriormente, foi implementado o novo fluxo supracitado. Por fim, foi implantado um novo servidor em borda com as mesmas configurações do servidor em nuvem, para que fosse possível a realização de uma comparação justa entre as soluções. Essa comparação foi realizada através de uma análise estatística dos tempos de ida e volta (RTT, do inglês *round-trip time*) das mensagens trocadas durante a execução do sistema, e de um teste A/B, a fim de determinar qual abordagem apresentou o menor tempo de resposta nas condições apresentadas.

Referencial teórico

Este capítulo apresenta os conceitos que sustentaram o desenvolvimento da PoC de gêmeos digitais abordada, incluindo Internet das Coisas e Gêmeos Digitais. Também explora a computação em borda como estratégia para reduzir o tempo de resposta, e as tecnologias utilizadas nos componentes do sistema, como o Unity, o FIWARE e o protocolo MQTT.

Internet das coisas

A IoT é resultado da convergência de progressos significativos em múltiplas áreas tecnológicas, representando uma evolução notável na maneira como o mundo físico interage com o digital. Ela pode ser descrita como um ecossistema no qual objetos de uso cotidiano (*e.g.* aparelhos eletrodomésticos, veículos, câmeras de segurança) e máquinas industriais são integrados com processadores, sensores e atuadores, conectando-se através de uma rede (Sátyro *et al.*, 2018).

Gêmeos digitais

Com a rápida expansão da IoT, os gêmeos digitais podem ser definidos como representações virtuais precisas de elementos do mundo real (*e.g.* máquinas, ambientes, pessoas *etc.*), conectadas continuamente a seus equivalentes físicos. Essa conexão permite refletir com precisão o estado atual de seus equivalentes do mundo real por meio de um fluxo sincronizado de dados (Madni; Madni; Lucero, 2019; Tao *et al.*, 2019).

Segundo Fuller *et al.* (2020), um gêmeo digital deve garantir a interatividade contínua e bidirecional entre as entidades física e virtual, de forma que mudanças em uma afetem a outra automaticamente. Essa característica o diferencia de modelos digitais estáticos, nos quais não há relação dinâmica entre as entidades.

A aplicabilidade de gêmeos digitais destaca-se em áreas como saúde, manufatura e manutenção. Eles podem ser usados para simular modelos personalizados de pacientes, auxiliando tratamentos, ou para modelar linhas de produção, prever falhas e otimizar processos produtivos (Benedictis *et al.*, 2023; Bjelland *et al.*, 2022; Fuller *et al.*, 2020).

Internet Tátil

Algumas aplicações de IoT exigem tempos de resposta extremamente baixos, destacando a relevância da Internet Tátil, uma abordagem de comunicação em rede focada em minimizar a latência e garantir alta confiabilidade na comunicação entre dispositivos. Esse conceito busca viabilizar transmissões rápidas o suficiente para simular a sensação de toque físico, encontrando aplicações em áreas como medicina remota e indústria (Aijaz; Sooriyabandara, 2019; Qiao *et al.*, 2020).

O principal desafio para sua implementação é a necessidade de latência inferior a milissegundos, um padrão ainda além das capacidades de sistemas atuais. O avanço dessa tecnologia depende da integração de diferentes soluções, muitas delas ainda em fase de pesquisa acadêmica (Berg *et al.*, 2017).

Latência e tempo de resposta

Neste contexto, o tempo de resposta é definido como o intervalo entre a interação do usuário e a reação do sistema (Shneiderman, 1984). Quando esse processo envolve a comunicação entre dispositivos, esse tempo

é impactado pela latência, que corresponde ao período necessário para que um pacote de dados percorra o caminho do ponto de origem ao destino (Khan *et al.*, 2020).

Dentre os fatores que influenciam a latência, destacam-se a distância física entre os pontos de troca, a quantidade de dispositivos de rede envolvidos e o meio de transmissão utilizado (*e.g.* fibra óptica, cabos de cobre *etc.*). O atraso de propagação é condicionado pelo tempo de processamento de cada dispositivo (*e.g.* switches, roteadores *etc.*) que estiver participando da transmissão, e pela velocidade dos sinais elétricos ou luminosos a depender do meio utilizado (Briscoe *et al.*, 2016).

Computação em borda

A mitigação do efeito da distância física é a premissa central de uma das tecnologias habilitadoras da Internet Tátil: a computação em borda. Propõe-se que parte ou todo o processamento de um sistema que, em um primeiro momento, esteja sendo realizado em um ambiente distante dos demais dispositivos (*e.g.* servidor em nuvem) passe a ser realizado por um servidor geograficamente mais próximo, no que se denomina borda da rede (Mouradian *et al.*, 2018; Shi *et al.*, 2016).

Message queing telemetry transport

Tanto na PoC de Rodrigues *et al.* (2024) quanto neste estudo, a comunicação foi implementada com o protocolo MQTT (sigla para *Message Queuing Telemetry Transport*), devido à sua eficiência e simplicidade, características adequadas para cenários de teste e demonstração (MQTT, 2024). O *broker* MQTT, *software* responsável por intermediar a comunicação aplicando o protocolo, foi o Eclipse Mosquitto, uma solução de código aberto, leve e de fácil implementação (Eclipse Foundation, 2024).

O protocolo opera na camada de aplicação do modelo OSI (sigla para *Open System Interconnection*), sobre o TCP (sigla para *Transmission Control Protocol*). Sua comunicação segue o paradigma de publicação e inscrição (Pub/Sub, do inglês *publish/subscribe*), em que dispositivos publicam dados em tópicos que servem como canais de transmissão. Após a publicação, o *broker* distribui as mensagens aos dispositivos inscritos nos tópicos correspondentes (Bender *et al.*, 2021).

Neste trabalho, foi utilizada a versão 3.1.1 do MQTT com o nível 0 de qualidade de serviço (QoS, do inglês *Quality of Service*). O protocolo oferece três níveis de QoS, sendo o 0 conhecido como “envie e esqueça”: nesse nível, não há nenhum mecanismo que garanta a entrega das mensagens, priorizando o desempenho e a rapidez na transmissão em detrimento da confiabilidade (Oasis, 2014).

FIWARE

No contexto de IoT e DTs, uma infraestrutura robusta se faz essencial para garantir a comunicação eficiente, o armazenamento e a orquestração dos dados. O FIWARE é uma plataforma aberta que oferece ferramentas voltadas para o desenvolvimento de soluções inteligentes, com foco na interoperabilidade e na adoção de padrões abertos. Projetada para facilitar a criação de ecossistemas conectados, a plataforma permite integrar dispositivos, coletar e processar dados, além de agregar serviços em áreas como infraestrutura urbana, agricultura, indústria e energia (FIWARE, 2024a).

Sua arquitetura é baseada em habilitadores genéricos (GE, do inglês *Generic Enablers*): componentes modulares que oferecem funcionalidades essenciais, como gerenciamento de dados, segurança, processamento de contexto e conectividade com IoT. Esses componentes simplificam o desenvolvimento ao eliminar a necessidade de reimplementar mecanismos básicos (FIWARE, 2023).

Entre os principais padrões suportados pelo FIWARE, destaca-se o NGSI (sigla para *Next Generation Services Interface*), que viabiliza a troca interoperável de informações com sistemas heterogêneos. Sua versão atualizada, NGSIv2, desenvolvida pela FIWARE Foundation, introduz uma API (sigla para *Application Programming Interface*), que padroniza a comunicação entre aplicações, facilita a integração de novos componentes e amplia a usabilidade do sistema (Fonseca *et al.*, 2019).

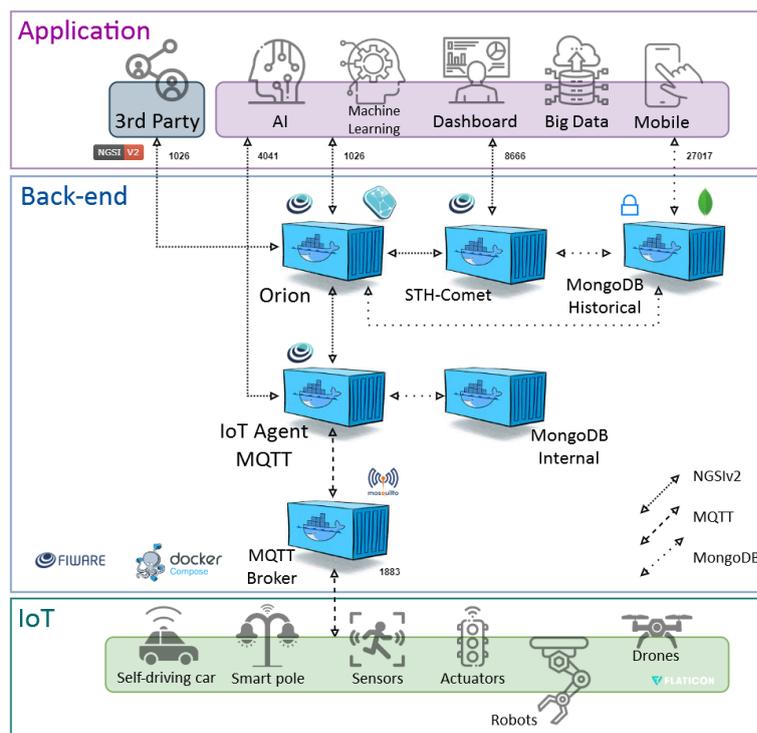
No centro da plataforma está o Orion Context Broker, que, utilizando o NGSIv2, é responsável por integrar diferentes elementos do ecossistema. Ele administra o ciclo de vida das informações de contexto, permitindo operações como criação, leitura, atualização e exclusão dos dados. Além disso, viabiliza a comunicação eficiente entre componentes, coordenando ações e promovendo o gerenciamento centralizado de dados em ambientes inteligentes (FIWARE, 2023; FIWARE, 2024b).

FIWARE descomplicado

O FIWARE descomplicado é uma ferramenta que simplifica o uso da plataforma FIWARE ao empregar tecnologias de agregação, como o Docker, garantindo a portabilidade e replicabilidade das soluções. Essa abordagem amplia sua aplicabilidade, permitindo o funcionamento em servidores com baixo poder computacional (Cabrin, 2024).

A Figura 1 ilustra a arquitetura da solução, destacando o Orion Context Broker como núcleo do gerenciamento de eventos. Ele se integra ao MQTT e ao armazenamento histórico utilizando MongoDB, um banco de dados não relacional para dados não estruturados, com suporte do *middleware* STH-Comet para registrar notificações no formato NGSIv2 (Cabrini, 2024; FIWARE, 2024c).

Figura 1. Arquitetura do FIWARE descomplicado.



Fonte: Cabrini (2024).

Os sensores e atuadores enviam informações ao Orion via MQTT, utilizando o IoT Agent para converter os dados ao padrão NGSIv2. Essa configuração elimina a necessidade de os dispositivos conhecerem o padrão, garantindo a interoperabilidade entre os componentes (Cabrini, 2024; FIWARE, 2024c).

Prova de conceito original

A seguir, serão descritos os componentes que integram a PoC. A entidade física, abordada na seção 2.8.1, é composta por um braço robótico controlado por um Arduino Uno R3 com Sensor Shield V5, enquanto a entidade virtual, abordada em 2.8.2, é um programa desenvolvido com o Blender (para modelagem) e o Unity (para lógica e simulação) (Rodrigues *et al.*, 2024).

Entidade física

O componente físico escolhido foi o RoboArm (Figura 2), um produto da empresa Robocore projetado para uso didático, oferecendo um método prático para se aprender conceitos de programação e mecânica. Ele se movimenta através de quatro servomotores, controlados por uma placa auxiliar integrada que deve ser escolhida pelo usuário (Robocore, 2024).

Figura 2. Braço robótico RoboArm.



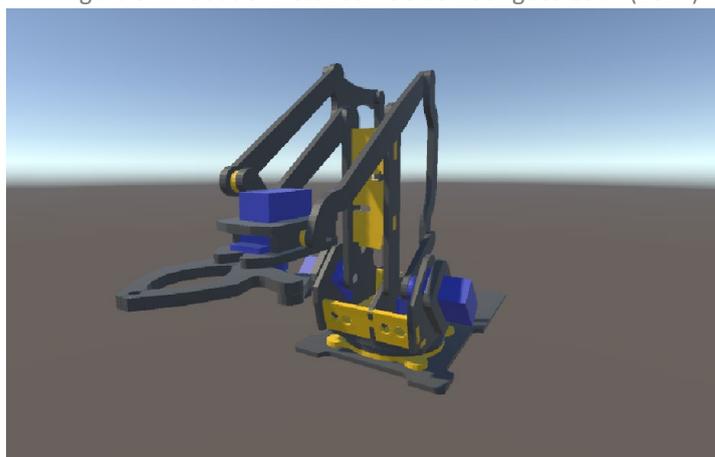
Fonte: Robocore (2024).

Foi utilizado o Arduino Uno R3 equipado com uma *shield* especializada capaz de controlar os motores presentes no RoboArm, criado para simplificar a prototipação de projetos eletrônicos. Seu elemento central é uma placa de circuito contendo um microcontrolador, como o ATmega328P, capaz de processar dados de diversas fontes externas (*e.g.* sensores, botões e potenciômetros) e, como resposta, acionar dispositivos e atuadores (Arduino, 2022).

Entidade digital

Entidade digital é um *software* de simulação que se comunica com a entidade física. A aplicação foi desenvolvida pelos autores da PoC, e conta com um modelo digital interativo do RoboArm, observado na Figura 3. O modelo foi criado com a ferramenta Blender, um *software* de modelagem tridimensional gratuito e de código aberto (Blender, 2024), e a aplicação que o manipula foi desenvolvida por intermédio da plataforma Unity (Rodrigues *et al.*, 2024).

Figura 3. Entidade virtual da PoC de Rodrigues et al. (2024).



Fonte: elaboração dos autores (2024).

O Unity é um motor de jogo multiplataforma utilizado para a criação de simulações e, principalmente, em jogos digitais. Projetado para facilitar o desenvolvimento independente, ele oferece uma gama de ferramentas para a criação e a integração de elementos essenciais em jogos, como gráficos, efeitos visuais, áudio e simulações de Física. Além dos jogos eletrônicos, a ferramenta é utilizada na produção de animações, aplicativos de realidade virtual, *softwares* de treinamento e simulações (Unity, 2024; Singh; Kaur, 2022).

Metodologia

Este estudo se baseou em uma abordagem experimental para avaliar o impacto de duas alterações propostas na implementação da PoC de Rodrigues *et al.* (2024), envolvendo o fluxo de comunicação e a localidade do servidor. Buscou-se reduzir o tempo de resposta na comunicação entre os gêmeos digitais.

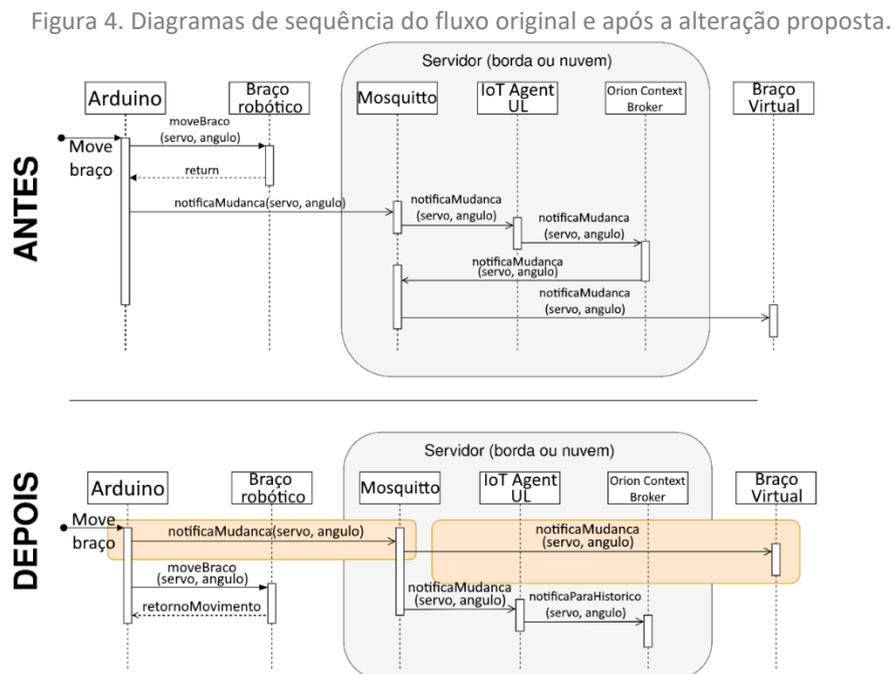
Inicialmente, a PoC foi reproduzida de forma fidedigna para servir como referência na avaliação de desempenho. Em seguida, as alterações descritas na seção 3.1 foram implementadas, e testes quantitativos e qualitativos foram realizados para verificar possíveis reduções no atraso de comunicação, conforme detalhado na seção 3.2. Os testes focaram no envio de notificações da entidade física para a digital, considerando-se suficiente avaliar apenas uma direção, com base na premissa de que os testes apresentariam resultados análogos caso fosse considerado o fluxo inverso.

Proposta de mudança

Após a reconstrução da PoC, foram identificadas limitações que comprometiam o sincronismo entre as entidades. Para resolver essas questões e maximizar a redução do tempo de resposta, foram implementadas mudanças no fluxo de comunicação antes do teste de mudança de localidade do servidor.

O fluxo original envolvia múltiplas etapas intermediárias. A notificação do comando era enviada ao *broker* MQTT somente após o robô físico concluir o movimento, passando pelo IoT Agent e pelo Orion Context Broker antes de alcançar a entidade virtual, o que impedia o sincronismo dos movimentos.

Como ilustrado na Figura 4, duas alterações foram realizadas para simplificar o fluxo. Primeiro, a notificação passou a ser enviada ao *broker* MQTT assim que o Arduino recebeu o comando, sem aguardar a execução do movimento. Em seguida, o Orion passou a atuar apenas no armazenamento histórico, enquanto a comunicação entre as entidades passou a ocorrer diretamente pelo *broker* MQTT.



Fonte: elaboração dos autores (2024).

As alterações propostas reduzem a quantidade de etapas e simplificam o fluxo de comunicação, com o intuito de reduzir o tempo de resposta do sistema. Os códigos e fluxos de comunicação atualizados foram disponibilizados em um repositório do GitHub (Turtera *et al.*, 2024).

Localização do servidor

A segunda proposta de alteração consistiu em transferir o *broker* MQTT e o FIWARE para um servidor localizado na borda da rede. Originalmente, ambos estavam hospedados em uma máquina virtual da Amazon Web Services (AWS), situada na região Norte da Virgínia, Estados Unidos, enquanto a PoC operava fisicamente em São Bernardo do Campo, São Paulo, Brasil.

Foi criada uma máquina virtual local, utilizando o *software* Oracle Virtualbox, com as mesmas especificações técnicas da máquina na nuvem, porém hospedada em um computador localizado em São Bernardo do Campo, conectado à mesma rede local que a PoC.

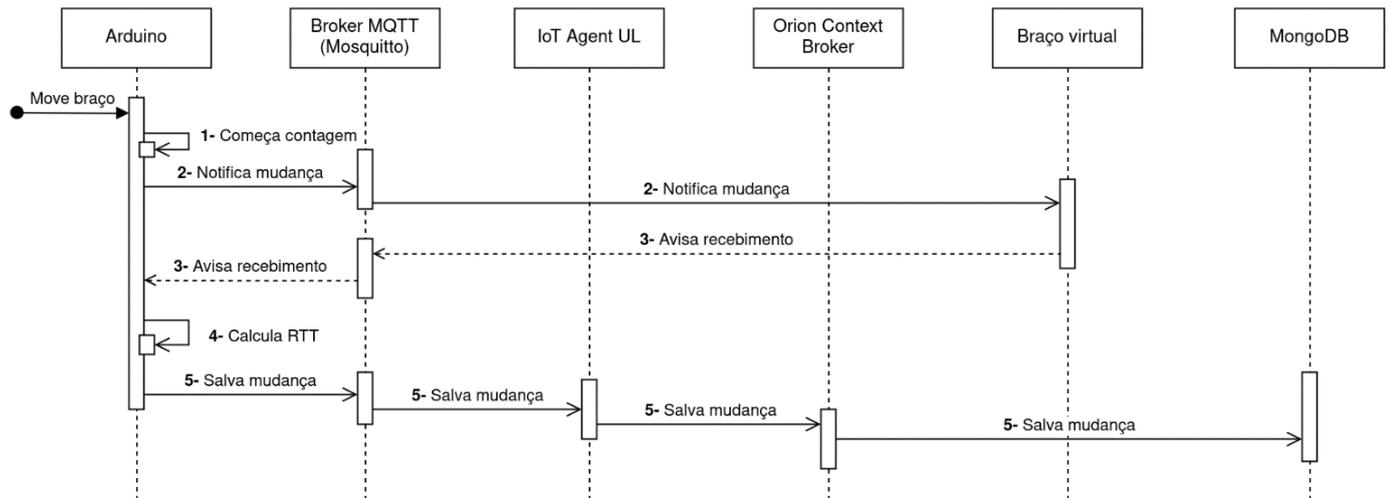
Estudo comparativo

Para avaliar o desempenho da nova arquitetura frente às mudanças propostas, adotou-se uma abordagem que integra métodos quantitativos e qualitativos. Aferiu-se a diferença significativa proporcionada pela implementação em borda utilizando o RTT como métrica. De forma complementar, foi realizado um teste A/B para confirmar se essa diferença também é significativa do ponto de vista do usuário.

Coleta de dados

No cenário dos testes quantitativos, foi realizada uma lógica de envio de comandos de forma contínua e repetitiva: dentro de um laço de repetição, o Arduino foi programado para enviar uma mensagem ao *broker* MQTT e aguardar o seu recebimento. Os comandos eram aleatórios e continham ângulos entre 0° e 360° para os servomotores. Para evitar que o equipamento fosse avariado ao tentar realizar movimentos inconsistentes, a lógica de movimentação foi desativada para esse teste. Esse fato não interfere nos resultados, uma vez que o tempo avaliado é influenciado unicamente pela comunicação. A Figura 5 ilustra o fluxo realizado durante as medições.

Figura 5. Diagrama de sequência do fluxo de coleta de valores de RTT.



Fonte: elaboração dos autores (2024).

Foram realizadas três baterias de testes nos dias 25 e 26 de outubro de 2024, entre 1h e 4h da manhã, com o intuito de minimizar a interferência do tráfego de rede. Cada bateria consistiu na coleta de 1000 amostras em cada localidade de servidor (nuvem e borda).

Análise quantitativa e ferramentas

A análise foi conduzida em duas etapas: primeiro, foi realizada uma análise exploratória dos dados utilizando métricas de estatística descritiva, a fim de inferir conclusões preliminares. Depois, para confirmar essas suposições, foi utilizado o teste U de Mann-Whitney para verificar se havia diferença entre as duas amostras de cada bateria.

Sob hipótese nula, o teste U de Mann-Whitney presume que a distribuição dos valores de duas amostras não apresenta diferença estatisticamente significativa. Esse teste foi escolhido em decorrência da natureza das distribuições dos dados coletados: conforme será evidenciado no capítulo referente aos resultados, após a análise visual por meio de histogramas e a aplicação do teste de normalidade de Shapiro-Wilk, concluiu-se que nenhuma das distribuições é normal, o que exigiu um teste não paramétrico (Mann; Whitney, 1947; Shapiro; Wilk, 1965).

Tanto o teste de Shapiro-Wilk quanto o teste U de Mann-Whitney fornecem um valor de p, que expressa a probabilidade de a hipótese nula ser verdadeira para o conjunto de dados analisado. Quando o valor de p é inferior a um limite predefinido, entende-se que há evidências suficientes para rejeitar a hipótese

nula. Embora não haja um único valor ideal para esse limite, níveis como 0,05 ou 0,005 são utilizados convencionalmente a depender do rigor exigido pelo estudo (Benjamin *et al.*, 2017).

Todos os valores foram obtidos a partir o *software* IBM SPSS Statistics. Os gráficos, por sua vez, foram gerados utilizando a linguagem de programação Python, especificamente com o uso das bibliotecas Matplotlib, Seaborn e Pandas.

Análise qualitativa

Para avaliar a percepção dos usuários em relação à diferença de sincronismo em cada implementação de localidade de servidor, foi aplicado um teste A/B. Os voluntários foram expostos, em ordem aleatória, a uma sequência de interações com a aplicação hospedada na nuvem (A) e na borda (B). A seguir, foram orientados a avaliar, com uma nota de 1 a 5, a percepção de sincronia entre o movimento realizado no controle e a resposta da entidade digital. É importante ressaltar que todos os voluntários eram alunos do curso superior de Engenharia de Computação, e estavam familiarizados com o tema, o que lhes permitiu observar nuances no sincronismo com maior precisão.

Especificações das ferramentas utilizadas

Conforme citado na seção 3.1.2, buscou-se isolar o impacto da localidade do servidor sobre o tempo de resposta para avaliar a proposta de implementação de um servidor em borda, exigindo, portanto, que a máquina virtual tivesse as mesmas especificações técnicas em ambas as situações. Essas especificações podem ser encontradas no Quadro 1.

Quadro 1 – Especificações do ambiente de execução do FIWARE e do Mosquitto.

Componente	Especificação
Sistema operacional	Ubuntu 20.04.4
vCPUs	1
Memória (GiB)	1
Armazenamento (GB)	30

Fonte: elaboração dos autores (2024).

Diversas ferramentas foram necessárias para o desenvolvimento da PoC e a realização dos testes. O Quadro 2 contém as versões de cada *software* e biblioteca utilizados neste trabalho.

Quadro 2 – Versões dos softwares e bibliotecas utilizados.

Software/Biblioteca	Versão
Eclipse Mosquitto	2.0.18
IBM SPSS Statistics	30.0.0.0 (Avaliação)
IoT Agent	3.5.0
Matplotlib	3.9.2
MongoDB	4.4
Oracle VM VirtualBox	7.0.20
Orion Context Broker	3.11
Pandas	2.2.2
Python	3.12.5
Seaborn	0.13.2
STH-Comet	2.11.0
Unity	2022.3.12f1

Fonte: elaboração dos autores (2024).

Análise dos resultados

A seguir, serão apresentados os resultados dos testes realizados para avaliar o impacto das alterações propostas no fluxo de comunicação e na localidade do servidor.

Resultado da alteração estrutural do fluxo de comunicação

Conforme explicado anteriormente, o fluxo de trabalho original não permitia que os gêmeos digitais atuassem em sincronia, uma vez que o código do Arduino fora programado para não enviar a notificação do comando ao braço virtual até que o movimento do braço físico fosse finalizado. Ao remover essa restrição, a diferença de tempo de resposta pôde ser percebida de forma qualitativa, ainda que sem a medição exata do valor apresentado no caso anterior. Por isso, os testes foram aplicados somente para a alteração na localidade do servidor.

Resultado da alteração da localização do servidor

A Tabela 1 apresenta as métricas descritivas dos dados coletados. Percebe-se que as três baterias apresentaram valores consistentes entre si, validando a confiabilidade das amostras para análises estatísticas. Entre os cenários avaliados, os dados revelam que a média do RTT na borda é cerca de 21 vezes menor do que na nuvem, demonstrando uma significativa redução no tempo de resposta. O alto coeficiente de variação na borda sugere que a solução é mais instável, porém, considerando que 90% dos valores registrados estão próximos à média, pode-se inferir que a razão do aumento dessa medida seja o grau de extremidade de valores atípicos que não representam o grupo de dados estudados.

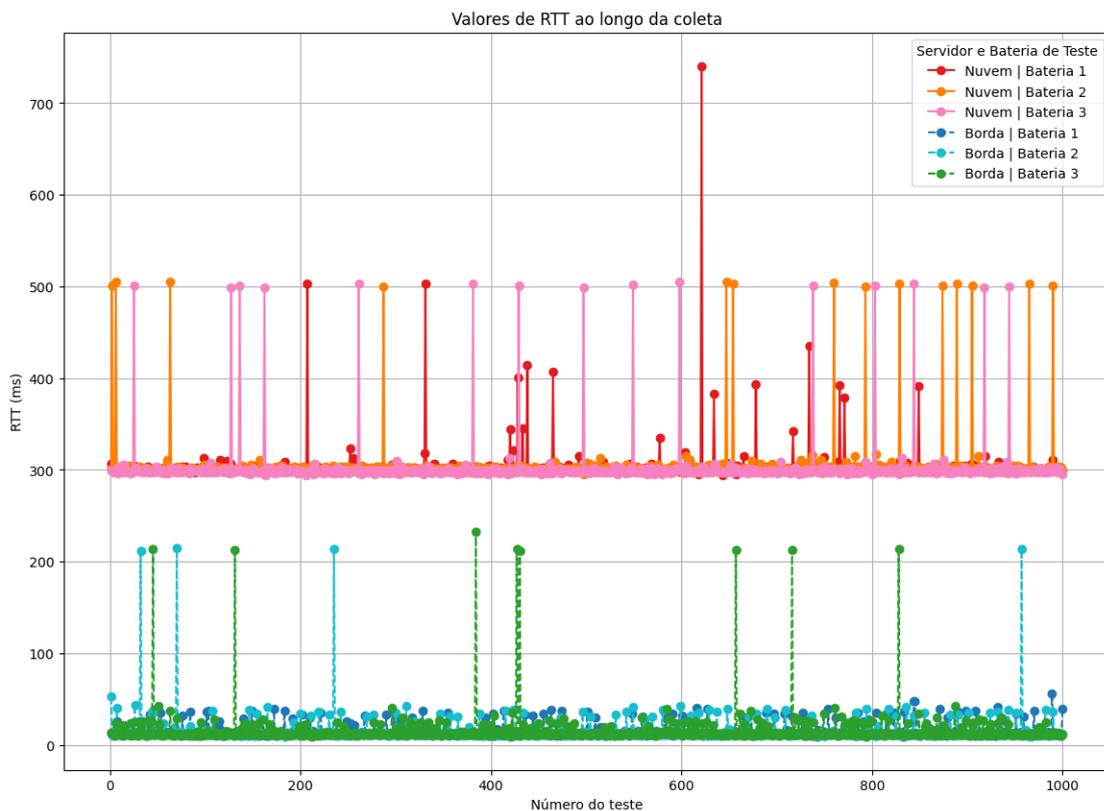
Tabela 1 – Métricas descritivas dos grupos de valores de RTT coletados.

Servidor	Nuvem			Borda			
	Bateria	1	2	3	1	2	3
Média (ms)		302,28	304,08	301,79	13,70	14,38	15,12
Desvio padrão (ms)		19,40	23,76	24,68	6,11	14,01	18,62
Coeficiente de variação (%)		6,42	7,81	8,18	44,57	97,45	123,17
Mínimo (ms)		294,64	295,21	294,58	9,52	9,60	9,42
Percentil 90 (ms)		302,71	303,31	301,02	14,98	14,22	20,98
Máximo (ms)		740,45	504,98	504,79	55,87	214,69	233,12

Fonte: elaboração dos autores (2024).

Além disso, observou-se que todos os valores obtidos na borda, incluindo os atípicos mais altos, permaneceram inferiores ao menor tempo registrado na nuvem. Essa separação clara entre os grupos, ilustrada na Figura 6, confirma a diferença nos tempos de resposta entre as abordagens, com a borda apresentando valores consistentemente mais baixos.

Figura 6. Valores de RTT ao longo da coleta.



Fonte: elaboração dos autores (2024).

Para reforçar essa conclusão, foi aplicado um teste não paramétrico de heterogeneidade entre amostras distintas, confirmando que os valores observados são, do ponto de vista estatístico, significativamente diferentes. Optou-se pelo teste U de Mann-Whitney mediante o fato de que nenhuma das distribuições

era normal. A não normalidade foi confirmada por três métricas: curtose, assimetria e valor de p do teste de Shapiro-Wilk, os quais se encontram detalhados na Tabela 2.

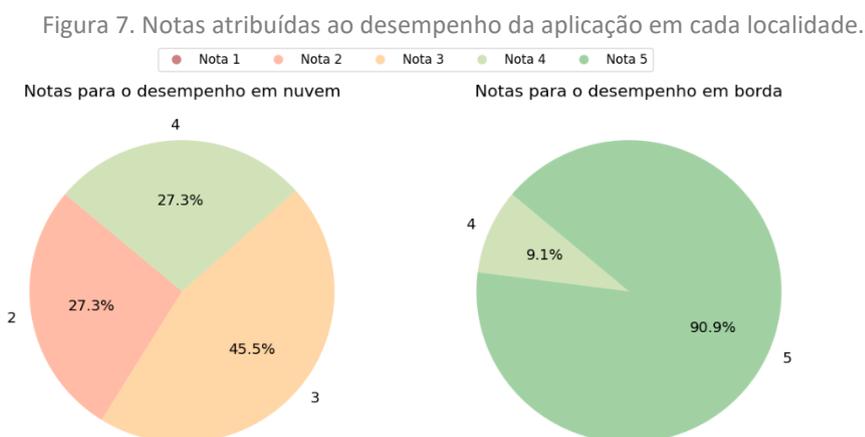
Tabela 2 – Métricas referentes à normalidade de cada distribuição.

Servidor	Bateria	Curtose	Assimetria	Valor de p (Shapiro-Wilk)
Nuvem	1	287,25	15,02	<0,001
	2	65,27	8,17	<0,001
	3	60,93	7,91	<0,001
Borda	1	11,10	3,38	<0,001
	2	161,79	11,79	<0,001
	3	105,43	10,06	<0,001

Fonte: elaboração dos autores (2024).

Os resultados do teste U de Mann-Whitney confirmaram, com alto grau de significância, a diferença estatística entre os tempos de resposta. Em todas as baterias analisadas, o valor de p obtido foi inferior a 0,001, possibilitando a rejeição da hipótese nula para os casos observados e ressaltando que essa proposta de alteração proporcionou uma redução expressiva do tempo de resposta.

Por fim, para o teste A/B foram coletadas 11 amostras, e todos os voluntários relataram que o tempo de resposta da abordagem em borda parecia mais baixo em comparação à nuvem, reforçando a conclusão anterior de forma qualitativa. A Figura 7 ilustra essa percepção, mostrando que 90,9% dos participantes atribuíram nota máxima ao desempenho do sistema em borda, enquanto nenhum deles deu uma nota maior do que 4 ao sistema em nuvem.



Fonte: elaboração dos autores (2024).

Considerações finais

Com o cumprimento das metas específicas, o objetivo principal deste trabalho foi alcançado. A replicação fiel da PoC descrita por Rodrigues *et al.* (2024) assegurou a confiabilidade do ambiente experimental, permitindo a implementação de um novo fluxo de comunicação e a mudança para um servidor em borda. Essas alterações eliminaram atrasos desnecessários, resultando em uma redução expressiva no tempo de resposta, comprovada pelos testes e pela análise estatística.

Apesar desse êxito, algumas limitações foram identificadas, como as restrições do protocolo MQTT e do *hardware* da placa Arduino, além do uso de um ambiente controlado que pode não refletir cenários mais complexos. Trabalhos futuros podem investigar alternativas, como protocolos mais rápidos, placas de maior capacidade e testes em condições variadas, a fim de ampliar a compreensão sobre os benefícios da computação em borda no contexto estudado.

Referências

- AIJAZ, A.; SOORIYABANDARA, M. The tactile internet for industries: a review. **Proceedings Of The IEEE**, [S.L.], v. 107, n. 2, p. 414-435, fev. 2019. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/jproc.2018.2878265>.
- ALMASAN, P. *et al.* Network digital twin: context, enabling technologies, and opportunities. **IEEE Communications Magazine**, [S.L.], v. 60, n. 11, p. 22-27, nov. 2022. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/mcom.001.2200012>.
- ARDUINO. **What is Arduino?** 2022. Disponível em: <https://docs.arduino.cc/learn/starting-guide/whats-arduino/>. Acesso em: 21 abr. 2024.
- BENDER, M. *et al.* Open-source MQTT evaluation. **2021 IEEE 18Th Annual Consumer Communications & Networking Conference (CCNC)**, [S.L.], p. 1-4, 9 jan. 2021. IEEE. DOI: <http://dx.doi.org/10.1109/ccnc49032.2021.9369499>.
- BENEDICTIS, A. *et al.* Digital twins in healthcare: an architectural proposal and its application in a social distancing case study. **IEEE Journal Of Biomedical And Health Informatics**, [S.L.], v. 27, n. 10, p. 5143-5154, out. 2023. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/jbhi.2022.3205506>.
- BENJAMIN, D. *et al.* Redefine statistical significance. **Nature Human Behaviour**, [S.L.], v. 2, n. 1, p. 6-10, 1 set. 2017. Springer Science and Business Media LLC. DOI: <http://dx.doi.org/10.1038/s41562-017-0189-z>.
- BERG, D. *et al.* Challenges in haptic communications over the tactile internet. **IEEE Access**, [S.L.], v. 5, p. 23502-23518, 2017. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/access.2017.2764181>.
- BJELLAND, O. *et al.* Toward a digital twin for arthroscopic knee surgery: a systematic review. **IEEE Access**, [S.L.], v. 10, p. 45029-45052, 2022. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/access.2022.3170108>.
- BLENDER. Introduction. **Docs Blender**, 2024. Disponível em: https://docs.blender.org/manual/en/latest/getting_started/about/introduction.html. Acesso em: 21 ago. 2024.

BRISCOE, B. *et al.* Reducing internet latency: a survey of techniques and their merits. **IEEE Communications Surveys & Tutorials**, [S.L.], v. 18, n. 3, p. 2149-2196, 2016. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/comst.2014.2375213>

CABRINI, F. **FIWARE Descomplicado**. 2024. Disponível em: <https://github.com/fabiocabrini/fiware>. Acesso em: 24 abr. 2024.

ECLIPSE FOUNDATION. **Eclipse Mosquitto**. 2024. Disponível em: <https://mosquitto.org>. Acesso em: 9 dez. 2024.

FIWARE. **FIWARE Catalogue**. 2023. Disponível em: <https://www.fiware.org/catalogue/>. Acesso em: 25 abr. 2024.

FIWARE. **FIWARE: Open APIs for Open Minds**. 2024a. Disponível em: <https://www.fiware.org>. Acesso em: 25 abr. 2024.

FIWARE. **Fiware-Orion**. 2024b. Disponível em: <https://fiware-orion.readthedocs.io/en/master/>. Acesso em: 28 abr. 2024.

FIWARE. **Fiware-STH-Comet**. 2024c. Disponível em: <https://fiware-sth-comet.readthedocs.io/en/latest/>. Acesso em: 27 set. 2024.

FONSECA, J. *et al.* **FIWARE-NGSI v2 Specification**. 2019. Disponível em: <https://fiware.github.io/specifications/ngsiv2/stable/>. Acesso em: 28 abr. 2024.

FULLER, A. *et al.* Digital twin: enabling technologies, challenges and open research. **IEEE Access**, [S.L.], v. 8, p. 108952-108971, 2020. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/access.2020.2998358>

HUYNH, D. *et al.* Edge intelligence-based ultra-reliable and low-latency communications for digital twin-enabled metaverse. **IEEE Wireless Communications Letters**, [S.L.], v. 11, n. 8, p. 1733-1737, ago. 2022. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/lwc.2022.3179207>.

IBM. **O que é MongoDB**. 2024. Disponível em: <https://www.ibm.com/br-pt/topics/mongodb>. Acesso em: 27 set. 2024.

KHAN, U. *et al.* Latency reduction in optical metro networks. **2020 IEEE 23Rd International Multitopic Conference (INMIC)**, [S.L.], p. 1-5, 5 nov. 2020. IEEE. DOI: <http://dx.doi.org/10.1109/inmic50486.2020.9318214>.

MA, Z. *et al.* High-reliability and low-latency wireless communication for internet of things: challenges, fundamentals, and enabling technologies. **IEEE Internet Of Things Journal**, [S.L.], v. 6, n. 5, p. 7946-7970, out. 2019. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/jiot.2019.2907245>.

MADNI, A.; MADNI, C.; LUCERO, S. Leveraging digital twin technology in model-based systems engineering. **Systems**, [S.L.], v. 7, n. 1, p. 7, 30 jan. 2019. MDPI AG. DOI: <http://dx.doi.org/10.3390/systems7010007>.

MANN, H.; WHITNEY, D. On a test of whether one of two random variables is stochastically larger than the other. **The Annals Of Mathematical Statistics**, [S.L.], v. 18, n. 1, p. 50-60, mar. 1947. Institute of Mathematical Statistics. DOI: <http://dx.doi.org/10.1214/aoms/1177730491>.

MORAES, R. **Indústria 4.0: impactos sociais e profissionais**. 1. ed. São Paulo: Blucher, 2020. E-book. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 14 abr. 2024.

MOURADIAN, C. *et al.* A comprehensive survey on fog computing: state-of-the-art and research challenges. **IEEE Communications Surveys & Tutorials**, [S.L.], v. 20, n. 1, p. 416-464, 2018. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/comst.2017.2771153>.

MQTT. **MQTT: The standard for IoT messaging**. Disponível em: <https://mqtt.org>. Acesso em: 24 abr. 2024.

OPEN MOBILE ALLIANCE. **Next Generation Service Interfaces Architecture**. 2012. Disponível em: http://www.openmobilealliance.org/release/NGSI/V1_0-20120529-A/OMA-AD-NGSI-V1_0-20120529-A.pdf. Acesso em: 20 maio 2024.

QIAO, Y. *et al.* Haptic communication: toward 5G tactile internet. **2020 Cross Strait Radio Science & Wireless Technology Conference (CSRSWTC)**, [S.L.], p. 1-3, 13 dez. 2020. IEEE. DOI: <http://dx.doi.org/10.1109/csrswtc50769.2020.9372659>.

ROBOCORE. **Braço Robótico RoboARM**. Disponível em: <https://www.robocore.net/robotica-robocore/braco-robotico-roboarm>. Acesso em: 21 abr. 2024.

RODRIGUES, Y. *et al.* Digital twins: controle dos movimentos de um braço robótico utilizando MQTT e NGSI. **FTT Journal Of Engineering And Business**, São Bernardo do Campo, v. 1, n. 9, p. 41-62, jun. 2024. Disponível em: <https://saijournal.cefsa.org.br/>. Acesso em: 11 jul. 2024.

SÁTYRO, W. *et al.* (org.). **Indústria 4.0: conceitos e fundamentos**. 1. ed. São Paulo, SP: Blucher, 2018. E-book. Disponível em: <https://plataforma.bvirtual.com.br>. Acesso em: 14 abr. 2024.

SHAPIRO, S.; WILK, M. An analysis of variance test for normality (complete samples). **Biometrika**, [S.L.], v. 52, n. 3-4, p. 591-611, 1 dez. 1965. Oxford University Press (OUP). DOI: <http://dx.doi.org/10.1093/biomet/52.3-4.591>.

SHI, W. *et al.* Edge computing: vision and challenges. **IEEE Internet Of Things Journal**, [S.L.], v. 3, n. 5, p. 637-646, out. 2016. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/jiot.2016.2579198>.

SHNEIDERMAN, B. Response time and display rate in human performance with computers. **ACM Computing Surveys (CSUR)**, v. 16, n. 3, p. 265-285, 1984.

SINGH, S.; KAUR, A. Game development using Unity game engine. **2022 3Rd International Conference On Computing, Analytics And Networks (ICAN)**, [S.L.], p. 1-3, 18 nov. 2022. IEEE. DOI: <http://dx.doi.org/10.1109/ican56228.2022.10007155>.

TAO, F. *et al.* Digital twin in industry: state-of-the-art. **IEEE Transactions On Industrial Informatics**, [S.L.], v. 15, n. 4, p. 2405-2415, abr. 2019. Institute of Electrical and Electronics Engineers (IEEE). DOI: <http://dx.doi.org/10.1109/tii.2018.2873186>.

TURTERA, G. *et al.* **TCC-DT-Robot**. 2024. Disponível em: <https://github.com/guitartera/TCC-DT-Robot>. Acesso em: 28 out. 2024.

UNITY. **Plataforma de desenvolvimento em tempo real do Unity | 3D, 2D, engine VR e AR**. Disponível em: <https://unity.com/pt>. Acesso em: 22 abr. 2024.