

Gleycykelly Syssy Indymayer Carnot Amaro

**Aplicação de realidade aumentada para monitoramento e manutenção da
infraestrutura computacional na indústria 4.0**

Goiânia
2024



UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TRABALHO DE CONCLUSÃO DE CURSO DE GRADUAÇÃO NO REPOSITÓRIO INSTITUCIONAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio do Repositório Institucional (RI/UFG), regulamentado pela Resolução CEPEC no 1240/2014, sem ressarcimento dos direitos autorais, de acordo com a Lei no 9.610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo dos Trabalhos de Conclusão dos Cursos de Graduação disponibilizado no RI/UFG é de responsabilidade exclusiva dos autores. Ao encaminhar(em) o produto final, o(s) autor(a)(es)(as) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do Trabalho de Conclusão de Curso de Graduação (TCCG)

Nome(s) completo(s) do(a)(s) autor(a)(es)(as): Gleycykelly Syssy Indymayer Carnot Amaro

Título do trabalho: Aplicação de realidade aumentada para monitoramento e manutenção da infraestrutura computacional na indústria 4.0

2. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador) Concorda com a liberação total do documento SIM NÃO¹

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante: a) consulta ao(à)(s) autor(a)(es)(as) e ao(à) orientador(a); b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo do TCCG. O documento não será disponibilizado durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro.

Obs.: Este termo deve ser assinado no SEI pelo orientador e pelo autor.



Documento assinado eletronicamente por **Marcelo Stehling De Castro, Professor do Magistério Superior**, em 12/12/2024, às 13:27, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Gleycykelly Syssy Indymayer Carnot Amaro, Discente**, em 12/12/2024, às 21:42, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5010171** e o código CRC **AA23AC97**.

Referência: Processo nº 23070.046482/2024-00

SEI nº 5010171

Glecykelly Syssy Indymayer Carnot Amaro

Aplicação de realidade aumentada para monitoramento e manutenção da infraestrutura computacional na indústria 4.0

Trabalho de conclusão de curso apresentado na Escola de Engenharia Elétrica, Mecânica e de Computação como requisito para a conclusão do curso de Engenharia de Computação e obtenção do título de Engenheiro de Computação.

Universidade Federal de Goiás – UFG
Escola de Engenharia Elétrica, Mecânica e de Computação (EMC)

Orientador: Prof. Dr. Marcelo Stehling de Castro

Goiânia
2024

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Amaro, Gleicykelly Syssy Indymayer Carnot
Aplicação de realidade aumentada para monitoramento e manutenção da infraestrutura computacional na indústria 4.0 [manuscrito] / Gleicykelly Syssy Indymayer Carnot Amaro. - 2024. xv, 15 f.

Orientador: Prof. Dr. Marcelo Stehling de Castro.
Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Goiás, Escola de Engenharia Elétrica, Mecânica e de Computação (EMC), Engenharia da Computação, Goiânia, 2024.
Bibliografia.
Inclui siglas, abreviaturas.

1. Realidade Aumentada. 2. Manutenção. 3. Indústria 4.0. 4. Tempo de Inatividade. I. Castro, Marcelo Stehling de, orient. II. Título.

CDU 62+004+005



UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

ATA DE DEFESA DE TRABALHO DE CONCLUSÃO DE CURSO

Ao(s) **12** dia(s) do mês de **dezembro** do ano de **2024** iniciou-se a sessão pública de defesa do Trabalho de Conclusão de Curso (TCC) intitulado “**Aplicação de realidade aumentada para monitoramento e manutenção da infraestrutura computacional na indústria 4.0**”, de autoria de **Gleycykelly Syssy Indymayer Carnot Amaro**, do curso de **Engenharia de Computação**, da **Escola de Engenharia Elétrica, Mecânica e de Computação (EMC)** da UFG. Os trabalhos foram instalados pelo Prof. Dr. **Marcelo Stehling de Castro** – EMC/UFG com a participação dos demais membros da Banca Examinadora: Prof. Dr. **Rodrigo Pinto Lemos** – EMC/UFG e Engenheiro Especialista **Gustavo Dias de Oliveira** – EMC/UFG. Após a apresentação, a banca examinadora realizou a arguição do(a) estudante. Posteriormente, de forma reservada, a Banca Examinadora atribuiu a nota final de **10,0**, tendo sido o TCC considerado **APROVADO**.

Proclamados os resultados, os trabalhos foram encerrados e, para constar, lavrou-se a presente ata que segue assinada pelos Membros da Banca Examinadora.



Documento assinado eletronicamente por **Marcelo Stehling De Castro, Professor do Magistério Superior**, em 12/12/2024, às 10:09, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Gustavo Dias De Oliveira, Técnico de Tecnologia da Informação**, em 12/12/2024, às 10:12, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Rodrigo Pinto Lemos, Professor do Magistério Superior**, em 12/12/2024, às 10:15, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **5010169** e o código CRC **DD2C0588**.

Aplicação de realidade aumentada para monitoramento e manutenção da infraestrutura computacional na indústria 4.0

Gleyckelly Syssy Indymayer Carnot Amaro¹, Marcelo Stehling de Castro²

Universidade Federal de Goiás (UFG) - Escola de Engenharia Elétrica, Mecânica e de Computação (EMC)
Goiânia, Goiás, Brasil, e-mails: ¹gleyckellysyssy@discente.ufg.br, ²mcastro@ufg.br

Resumo—A Indústria 4.0 busca aprimorar as cadeias produtivas por meio da integração de diferentes tecnologias, promovendo a otimização de processos e maior eficiência operacional. Entre essas tecnologias, destaca-se a Realidade Aumentada (RA), que permite sobrepor objetos digitais 3D em ambientes reais, possibilitando interação e visualização de dados em tempo real. Este trabalho tem como objetivo desenvolver um aplicativo Android para a gestão e manutenção de infraestruturas computacionais em indústrias inteligentes, utilizando recursos de RA e ferramentas como Unity e Vuforia. O aplicativo proporcionará a visualização de métricas e informações do computador, além de integrar um modelo 3D interativo, facilitando a manutenção e contribuindo para a redução do tempo de inatividade desnecessário.

Palavras-chave—realidade aumentada, manutenção, indústria 4.0, tempo de inatividade

Abstract—Industry 4.0 aims to enhance production chains through the integration of various technologies, promoting process optimization and greater operational efficiency. Among these technologies, Augmented Reality (AR) stands out, allowing the overlay of 3D digital objects onto real-world environments, enabling interaction and real-time data visualization. This work aims to develop an Android application for the management and maintenance of computational infrastructures in smart industries, utilizing AR resources and tools such as Unity and Vuforia. The application will provide the visualization of computer metrics and information, in addition to integrating an interactive 3D model, facilitating maintenance and contributing to the reduction of unnecessary downtime.

Index Terms—augmented reality, maintenance, industry 4.0, downtime

I. INTRODUÇÃO

A Indústria 4.0 está transformando a produção por meio da integração de tecnologias como Internet das Coisas (IoT), Inteligência Artificial (IA), Realidade Aumentada (RA), Big Data e Analytics[2] [26]. Estas tecnologias viabilizam a comunicação em tempo real entre máquinas, sensores e sistemas de controle, e facilitam a coleta e análise de grandes volumes de dados, fornecendo informações que facilitam a tomada de decisões mais inteligentes e precisas. Essa evolução resulta em cadeias produtivas mais eficientes, mas também gera novos desafios às infraestruturas computacionais, como servidores e data centers, que tornam-se cada vez mais complexos. A interrupção inesperada de uma infraestrutura de TI, dependendo do tamanho da organização, pode acarretar custos de até 1 milhão de dólares por hora [11], evidenciando a necessidade

de novas estratégias para mitigar o tempo de inatividade e garantir a continuidade dos processos industriais.

Neste contexto, a RA surge como uma solução inovadora para o monitoramento e diagnóstico de infraestruturas computacionais, permitindo a criação de modelos 3D interativos que funcionam como gêmeos digitais dos equipamentos, facilitando a inspeção sem interromper o funcionamento das máquinas.

Este trabalho propõe o desenvolvimento de um protótipo de RA para visualização e monitoramento de infraestruturas computacionais, utilizando modelos 3D interativos de computadores. A solução visa otimizar a manutenção dessas infraestruturas, contribuindo para a eficiência operacional nas indústrias.

O presente artigo está estruturado em cinco seções. Na seção II. Fundamentação Teórica, são apresentados os conceitos fundamentais e examina como a RA tem sido aplicada na Indústria 4.0, proporcionando a base teórica necessária para o desenvolvimento do protótipo. A seção III. Trabalhos Relacionados examina como a RA tem sido aplicada na Indústria 4.0, com foco nas aplicações voltadas para o monitoramento e manutenção de máquinas. Na seção IV. Desenvolvimento do Protótipo, é detalhado o processo de implementação do protótipo, incluindo as ferramentas utilizadas e as etapas do desenvolvimento. Por fim, a seção V. Conclusão apresenta as considerações finais e sugestões para futuros projetos.

II. FUNDAMENTAÇÃO TEÓRICA

Esta seção visa apresentar e definir os principais conceitos e tecnologias que serão explorados ao longo do trabalho, proporcionando o embasamento teórico necessário para compreender a solução proposta. Serão abordados temas como a Indústria 4.0, RA e suas aplicações, com foco em como essas tecnologias estão sendo implementadas no contexto industrial.

A. Indústria 4.0

Ao longo da história, a produção no chão de fábrica passou por várias transformações marcantes, conhecidas como revoluções industriais. A primeira revolução foi impulsionada pela introdução de máquinas movidas a vapor, enquanto a segunda revolução se caracterizou pela introdução da eletricidade e pela produção em massa. A terceira revolução surgiu com o

advento dos computadores e a automação dos processos industriais. Atualmente, estamos vivenciando a quarta revolução industrial, conhecida como Indústria 4.0, que é marcada pela integração de tecnologias avançadas, como Internet das Coisas (IoT), Inteligência Artificial (IA), RA, Big Data e computação em nuvem. Essas tecnologias estão transformando a maneira de operar no chão de fábrica, promovendo maior eficiência e conectividade entre os sistemas e equipamentos [13] [23].

No entanto, a transição para a Indústria 4.0 também apresenta desafios, como a gestão e os custos de infraestruturas computacionais mais complexas, a necessidade de expertise técnica e questões relacionadas à cibersegurança [25]. Este trabalho se insere nesse contexto, explorando como a RA pode ser aplicada para mitigar os impactos dessa crescente complexidade tecnológica, com foco em propor uma solução acessível para monitoramento e manutenção de infraestruturas computacionais.

B. Realidade aumentada (RA)

A RA é uma tecnologia da computação gráfica que combina elementos virtuais tridimensionais com o mundo real, permitindo a interação com informações digitais sobrepostas ao ambiente físico [8]. Dispositivos como smartphones, tablets e óculos de RA, como o MS HoloLens 2, viabilizam a aplicação dessa tecnologia em diversos contextos, incluindo a Indústria 4.0.

Estudos recentes têm explorado o uso da RA para otimizar processos industriais. Em [7], foi desenvolvido um aplicativo Android que sobrepõe componentes digitais a elementos físicos em uma linha de montagem, facilitando o treinamento de trabalhadores e estudantes. A RA melhora a visualização e o entendimento dos componentes, agilizando o processo de aprendizado.

De maneira semelhante, descreve-se em [8] o desenvolvimento de uma aplicação que exibe, na tela inicial do aplicativo, imagens tridimensionais das peças ou matérias-primas a serem posicionadas em uma máquina específica. O objetivo é orientar o operador, por meio de setas, sobre a posição ideal para evitar erros. Após realizar testes com operadores com e sem experiência prévia na operação da máquina, concluiu-se que a adoção da RA resultou em uma redução significativa nos erros e no tempo de execução, quando comparado às instruções tradicionais em papel.

Outra aplicação importante da RA é descrita em [3], onde a tecnologia é utilizada para teleoperação e monitoramento da trajetória de robôs. O sistema desenvolvido permite a plotagem de erros durante o processo, contribuindo para a melhoria da eficiência operacional e a redução de falhas. Complementando esse conceito, [27] apresenta uma interface de RA que sobrepõe informações sobre os movimentos de um robô colaborativo, facilitando a interação humano-robô (HRC) e, assim, ajudando a reduzir os riscos de colisão em linhas de montagem. Essas soluções demonstram como a RA pode não apenas otimizar processos, mas também aumentar a segurança nas operações industriais.

Além disso, outro estudo com foco na melhoria da experiência de aprendizado foi implementado em [30], visando

aprimorar o entendimento do processo de corte de fio EDM. Através de um modelo interativo em 3D, os usuários podem interagir com a máquina e compreender melhor sua estrutura e funcionamento. Embora este exemplo tenha um foco mais educacional, ele compartilha o mesmo princípio de aplicar a RA para otimizar a compreensão e a interação com sistemas complexos.

Esses exemplos evidenciam o vasto potencial da RA para resolver diversos desafios na indústria, desde a melhoria da segurança e da eficiência operacional até o aprimoramento do treinamento e aprendizado de operadores. Este trabalho se baseia em abordagens semelhantes, adaptando-as ao contexto do monitoramento e diagnóstico de infraestruturas computacionais, com o objetivo de promover uma manutenção mais ágil e sem a necessidade de interrupção, contribuindo para a mitigação do tempo de inatividade das máquinas.

C. Realidade virtual (RV)

A RV é uma tecnologia que substitui os estímulos sensoriais do usuário por dados gerados por computador, criando a ilusão de estar em um ambiente totalmente simulado. Esse ambiente, conhecido como mundo virtual, pode ser modelado a partir de cenários imaginários ou representações alteradas da realidade. Um dos aspectos centrais da RV é a imersão, que permite ao usuário experimentar uma sensação de presença no ambiente virtual, variando entre um mundo fictício completamente imaginado ou uma versão transformada do mundo real. [1] O artigo [22] apresenta uma aplicação prática de RV voltada para adolescentes autistas na Malásia, com o objetivo de auxiliar no desenvolvimento de habilidades sociais e emocionais. A solução consiste em um ambiente virtual que simula situações do dia a dia, como compras em supermercados e interações sociais básicas, permitindo que os adolescentes pratiquem comportamentos e respostas apropriadas. Essa implementação evidencia o potencial da RV na área da saúde e bem-estar, oferecendo uma ferramenta acessível e eficaz para auxiliar adolescentes autistas a desenvolverem habilidades essenciais para a vida cotidiana.

D. Realidade mista (RM)

A RM é uma tecnologia que combina elementos de RA e RV criando experiências mais imersivas e interativas. Na RM, objetos do mundo real e virtual coexistem e interagem em um mesmo ambiente, permitindo que usuários manipulem objetos virtuais enquanto mantêm a percepção de seu ambiente físico. É considerada uma evolução contínua das tecnologias de RA e RV e é amplamente utilizada em áreas como jogos, educação, treinamento saúde e arquitetura.[14] Para estudar como o uso de RM pode melhorar a experiência do usuário em exposições de produtos e interações virtuais, o artigo [14] implementou uma solução de RM envolvendo a criação de um ambiente de RM para melhorar a experiência do usuário em interações com produtos virtuais, diferenciando-se da RV e da RA ao combinar elementos físicos e digitais que coexistem e interagem no mesmo espaço. Utilizando ferramentas como Autodesk 3ds Max e Unreal Engine, o projeto modelou um espaço virtual integrado a um laboratório físico equipado com

o HTC VIVE PRO 2, oferecendo uma experiência imersiva e interativa. Diferentemente da RV, que cria um ambiente completamente virtual, e da RA, que apenas sobrepõe elementos digitais ao mundo real, a RM permite a interação simultânea e integrada entre o ambiente físico e os objetos virtuais. Os participantes realizaram tarefas que simulavam a interação com uma máquina de café expresso, como localizar, configurar e operar o produto, com avaliações que destacaram a alta satisfação e a experiência realista, apesar dos desafios enfrentados por usuários inexperientes. Essa solução demonstra o potencial da RM em superar limitações de exibições físicas e online, com aplicações em varejo, treinamento e design.

Outra aplicação de RM foi descrita no artigo [10], que apresentou um protótipo projetado para a manipulação simultânea de objetos físicos e virtuais, integrando soluções táteis e visuais em um único dispositivo. O protótipo utiliza uma estrutura semelhante a um alicate, equipada com um motor servo para fornecer feedback tátil realista, simulando a resistência e o comportamento de objetos reais ou digitais durante a interação. Por exemplo, ao apertar ou desapertar uma porca, o usuário sente a mesma resistência que teria ao manipular um objeto físico, mesmo que a porca seja virtual. Essa integração permite uma transição mais natural entre os mundos físico e digital, trazendo cenários mais realistas em que o feedback tátil pode simular condições reais de uso. Tal abordagem demonstra o potencial da RM em enriquecer interações e criar experiências mais imersivas.

E. Comparativo entre RA, RV e RM

A RA consiste na sobreposição de objetos virtuais tridimensionais ao ambiente físico real, proporcionando uma interação que enriquece a percepção do mundo ao redor do usuário. Por outro lado, a RV cria uma simulação tridimensional completamente imersiva, permitindo que os usuários interajam em um ambiente digital que pode replicar o mundo real ou criar cenários totalmente fictícios. Para isso, a RV exige o uso de equipamentos especializados, como sensores, óculos de realidade virtual e outros dispositivos que aprimoram a experiência. Já a RM combina elementos da RA e da RV, criando um ambiente híbrido onde os mundos físico e virtual coexistem e interagem de forma dinâmica e natural. No que diz respeito ao nível de imersão, a RA oferece a menor imersão entre as três tecnologias, uma vez que mantém o foco no ambiente físico com acréscimos virtuais. A RV e a RM compartilham níveis semelhantes de imersão; no entanto, a RM destaca-se por proporcionar uma interação mais intuitiva e integrada entre os mundos real e virtual, tornando a experiência mais natural para o usuário.[20]

F. O Papel dos Marcadores na RA

Marcadores em RA desempenham um papel essencial na integração entre o mundo real e o virtual, funcionando como referências para rastrear, posicionar e interagir com objetos virtuais no ambiente físico. Eles podem ser de diferentes tipos, como imagens, objetos 3D ou superfícies planas, e são fundamentais para permitir a sobreposição de informações digitais de maneira precisa e contextualizada. [5]

O artigo [15] apresenta uma aplicação de RA que utiliza dois tipos principais de marcadores: marcadores de imagem e marcadores de plano. Os marcadores de imagem são associados a objetos ou espaços virtuais específicos, permitindo que, ao serem detectados pela câmera, os elementos digitais correspondentes sejam exibidos no ambiente real. Já os marcadores de plano são empregados para identificar superfícies físicas, como mesas ou pisos, possibilitando o posicionamento de objetos virtuais em locais adequados e contextuais, respeitando as características do ambiente. Essa abordagem melhora a interação e a imersão do usuário, já que os objetos virtuais se adaptam ao espaço físico de maneira natural.

III. TRABALHOS RELACIONADOS

Esta seção apresenta trabalhos recentes sobre a aplicação de RA na Indústria 4.0, com ênfase em soluções voltadas para a manutenção e o monitoramento de máquinas e dispositivos.

Diversos estudos têm explorado o uso de RA para otimizar a manutenção, monitoramento e visualização de dados em tempo real na Indústria 4.0. Em [4], um aplicativo de RA foi desenvolvido para aprimorar a manutenção de ativos de produção em ambientes industriais. Utilizando tecnologias como o Vuforia SDK e a plataforma Unity, o aplicativo permite que, ao apontar a câmera de um dispositivo móvel para um marcador como mostrados na Fig. 1 — uma imagem colada em cada parte do componente — as informações relacionadas ao componente sejam exibidas, assim como ilustrações técnicas sobre como desmontá-lo. Essa solução foi integrada ao sistema ERP da empresa, permitindo que os técnicos acessem e modifiquem os recursos do estoque diretamente pelo aparelho móvel, tornando a ferramenta ainda mais eficiente.

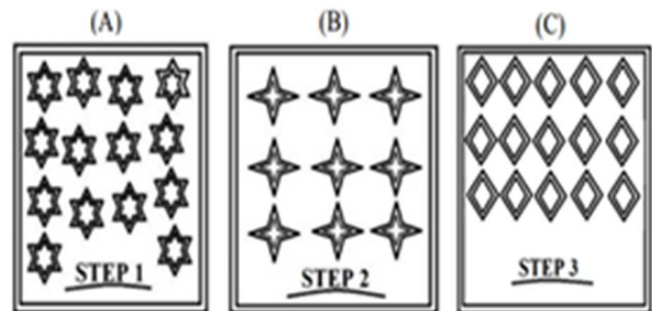


Fig. 1. (A) Marcador usado para exibir as ampliações no Passo 1. (B) Marcador usado para exibir as ampliações no Passo 2. (C). Reproduzido de [4].

No contexto de visualização de dados sobrepostos ao mundo real, o artigo [16] descreve o desenvolvimento de uma aplicação de RA para monitoramento de dados em tempo real em linhas de montagem conforme Fig.2. Essa aplicação permite que gerentes visualizem o desempenho geral e os detalhes específicos da linha, melhorando a eficiência operacional. Para sua criação, os autores utilizaram ferramentas como Unity e Mixed Reality Toolkit (MRTK), que acelera o desenvolvimento de aplicativos de realidade mista. Uma API Rest foi implementada, atualizando os dados a cada cinco segundos, garantindo informações sempre atualizadas.

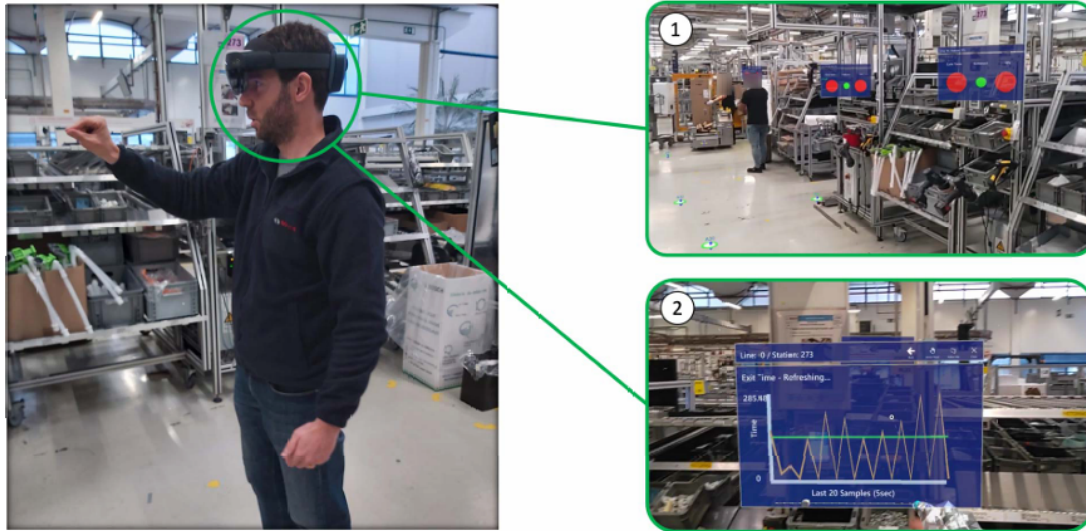


Fig. 2. Cenário industrial, com um gerente de linha de montagem visualizando dados em tempo real de toda a linha (1), bem como informações detalhadas para uma estação específica (2) por meio de RA. Reproduzido de [16].

Em [12], foi desenvolvida uma aplicação de RA voltada especificamente para manutenção industrial em refinarias de petróleo conforme pode ser observada na Fig. 3. O aplicativo Android detecta ativos e exibe um modelo tridimensional do ativo identificado, auxiliando na detecção e gerenciamento de falhas.

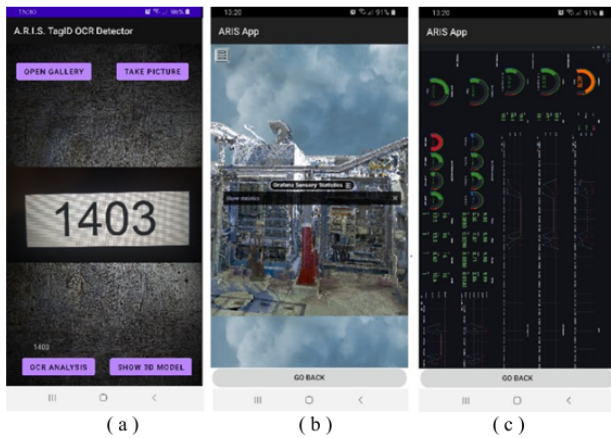


Fig. 3. a) Detecção de compressão de compressor usando aplicativo de telefone móvel e b) representação de gêmeos digitais usando links WebGL anotados para c) estatísticas em tempo real do compressor. Reproduzido de [12].

Outra aplicação relevante no campo do monitoramento foi proposta em [6], onde um sistema de monitoramento foi projetado para avaliar a saúde estrutural de pontes, como pode ser observado na Fig. 4 e na Fig. 5. Esse sistema permite o monitoramento em tempo real e a visualização dos dados relacionados à condição das pontes. A solução foi desenvolvida utilizando Unity e é personalizável para diferentes tipos de pontes, permitindo que o usuário interaja com sensores virtuais e acesse os dados.

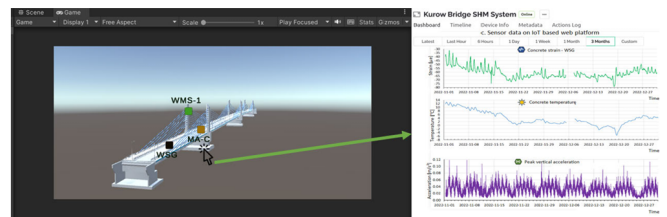


Fig. 4. Desenvolvimento de aplicativo no UNITY 3D e sua vinculação com a plataforma IoT. Reproduzido de [6].

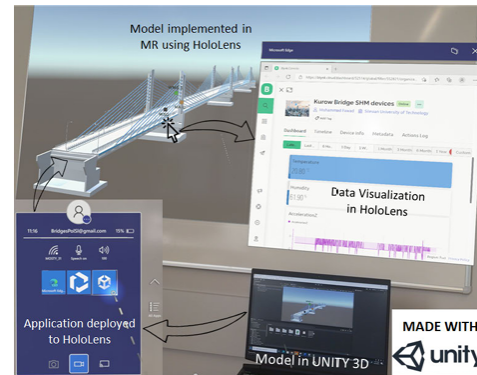


Fig. 5. Visualização de dados de SHM no HoloLens. Reproduzido de [6].

A revisão da literatura revelou que, embora muitas aplicações de RA na Indústria 4.0 sejam voltadas para o gerenciamento de dados operacionais e manutenção de maquinários industriais, ainda há uma lacuna no uso da RA para monitoramento e manutenção de dispositivos computacionais, como computadores, notebooks e servidores. Essa lacuna representa uma oportunidade promissora para futuras pesquisas e aplicações práticas na área.

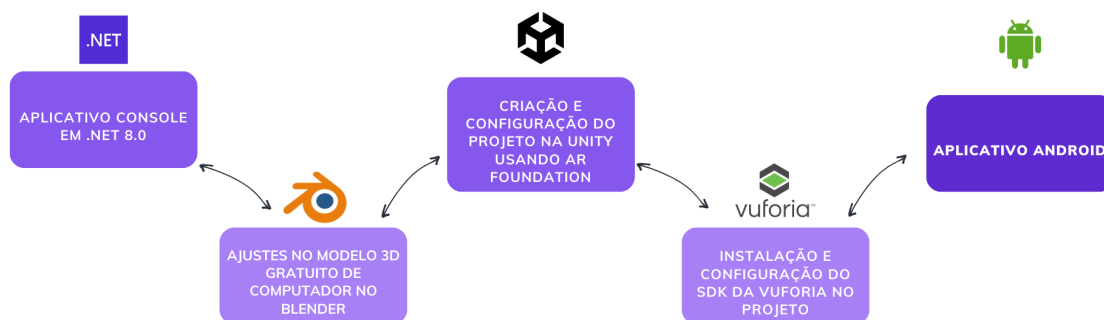


Fig. 6. Diagrama da implementação do protótipo. Fonte: Autoria própria.

IV. DESENVOLVIMENTO DO PROTÓTIPO

A. Objetivo do Protótipo

O protótipo foi desenvolvido com o objetivo de utilizar a RA para sobrepor métricas de computadores, facilitando o monitoramento e a obtenção de informações em tempo real. O usuário poderá acessar, por meio de um dispositivo móvel, um menu de RA que exibe dados relevantes sobre o hardware do computador, como informações sobre a memória RAM, GPU, placa-mãe e disponibilidade de *slots*, sem a necessidade de abrir ou manipular fisicamente a máquina. Além disso, o protótipo oferece a opção de interagir com um modelo 3D da máquina, permitindo ao operador explorar seus componentes de forma detalhada. Para acessar essas funcionalidades, o usuário precisará apenas escanear um QR Code, o que proporcionará uma experiência prática e eficiente diretamente em seu celular ou *tablet*.

B. Ferramentas e Tecnologias Utilizadas

Para a implementação deste protótipo, foi necessário o uso de diversas ferramentas e tecnologias que suportaram desde o desenvolvimento do aplicativo console até a criação e integração da aplicação de RA. Na Fig. 6 é possível analisar o passo a passo e as tecnologias e ferramentas relacionadas a cada passo para a concepção do protótipo. Cada ferramenta foi escolhida de acordo com sua adequação às necessidades do projeto, como a manipulação de métricas do computador, ajustes do modelo 3D e integração de funcionalidades de RA.

Nas seções seguintes, estão descritas as principais ferramentas e tecnologias utilizadas, juntamente com suas respectivas funções no desenvolvimento.

1) *Linguagem C#*: C# é uma linguagem de programação orientada a objetos, desenvolvida pela Microsoft, amplamente

utilizada no desenvolvimento de aplicações robustas e escaláveis, destacando pelas funcionalidades mostradas na Fig. 7. Reconhecida por sua versatilidade, C# é compatível com diferentes plataformas por meio do .NET, um *framework* que facilita a criação de software multiplataforma. Além disso, a linguagem combina características de alto nível, como abstração e tipagem estática, com recursos de baixo nível, que permitem o controle eficiente de memória e desempenho. Sua sintaxe intuitiva e extensibilidade tornam o C# uma escolha popular para o desenvolvimento de sistemas complexos, incluindo aplicativos de desktop, jogos (via Unity), APIs web e sistemas embarcados [19].



Fig. 7. Funcionalidades da Linguagem C#. Fonte: <https://www.tatvasoft.com/>

Atualmente a linguagem está na sua décima segunda versão, tendo o seu primeiro lançamento em janeiro de 2002, onde foi lançada junto com o visual Studio .NET 2002. [17]

2) *.NET*: O .NET é uma plataforma de desenvolvimento de software criada pela Microsoft que fornece um ecossistema robusto e versátil para criar uma ampla variedade de aplicações.

Ele suporta múltiplas linguagens de programação, incluindo C#, F# e Visual Basic, tornando-se uma escolha popular entre desenvolvedores pela integração mostrada na Fig. 8.

A plataforma .NET é cross-platform, permitindo o desenvolvimento de aplicações que podem ser executadas em diferentes sistemas operacionais, incluindo Windows, macOS e Linux. A versatilidade do .NET permite o desenvolvimento de diversos tipos de aplicações. É possível criar aplicações web dinâmicas e APIs RESTful utilizando o ASP.NET Core. Para dispositivos móveis, o .NET MAUI (Multi-platform App UI) permite o desenvolvimento de aplicativos nativos para Android, iOS, Windows e macOS a partir de uma única base de código. No contexto de aplicações *desktop*, a plataforma suporta tecnologias como WPF e Windows Forms para criar interfaces gráficas ricas. Também é amplamente utilizada no desenvolvimento de soluções cloud-native, com suporte nativo a serviços como Microsoft Azure. O .NET se destaca ainda no desenvolvimento de jogos, especialmente com o Unity, um dos *frameworks* mais populares no setor, que utiliza C#. Por fim, ele é aplicado em projetos de dispositivos IoT e sistemas embarcados, reforçando sua adaptabilidade a diferentes cenários de desenvolvimento [18].

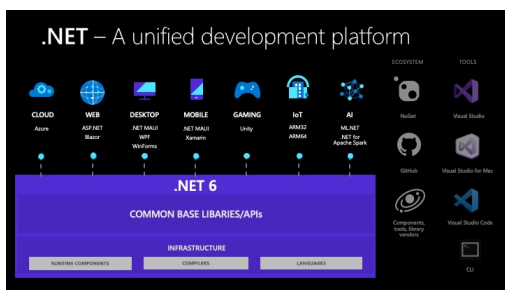


Fig. 8. Plataforma unificada de desenvolvimento. Fonte: <https://macoratti.net/21/11/net61.htm>

3) *Blender*: O Blender é um software multiplataforma amplamente utilizado para a criação de modelos 3D, sendo gratuito e de código aberto. Seu desenvolvimento teve início em 2 de janeiro de 1994 por Ton Roosendaal, mas a primeira versão pública, a 1.0, foi lançada em janeiro de 1995. Desde então, o Blender tem evoluído continuamente, tornando-se uma ferramenta robusta e amplamente adotada em áreas como modelagem, animação, renderização e edição de vídeo [9].



Fig. 9. Blender. Fonte: <https://www.blender.org>

4) *Unity*: A Unity é uma plataforma de desenvolvimento fundada em 2004 por David Helgason, Joachim Ante e Nicholas Francis, mostrada na Fig. 10. Ela oferece suporte para

a criação de projetos em 2D, 3D e também para aplicações de Realidade Virtual (VR) e RA [21].

Dentro desse contexto, a Unity disponibiliza o pacote AR Foundation, que facilita a criação de aplicativos de RA, fornecendo diversos *plug-ins* compatíveis com diferentes plataformas de destino [28]. Além disso, a Unity oferece o XR Interaction Toolkit, uma ferramenta que possibilita a interação entre objetos 3D e interfaces de usuário (UI) no projeto, utilizando eventos de entrada do Unity [29]



Fig. 10. Plataforma Unity. Fonte: <https://unity.com/pt>

5) *Vuforia*: A Vuforia Engine é uma plataforma poderosa para o desenvolvimento de aplicações de RA, mostrada na Fig. 11, oferecendo uma ampla gama de recursos e ferramentas para criar experiências imersivas em diversas plataformas. [24]



Fig. 11. Vuforia Engine. Fonte: <https://developer.vuforia.com/>

C. Desenvolvimento do sistema para obtenção de métricas

A criação do sistema para obtenção de métricas dos computadores foi o primeiro passo no desenvolvimento do protótipo. Este sistema foi implementado como uma aplicação console em .NET 8 e organizado em duas classes principais: `Program.cs` e `SystemMetrics.cs`, cada uma desempenhando papéis específicos e complementares na arquitetura.

1) *Classe Program.cs: Gerenciamento e Configuração da API*: A classe `Program.cs` contém quatro métodos principais, projetados para lidar com a identificação do computador, o preparo dos dados coletados, a configuração do servidor web e a inicialização do serviço. A seguir, os detalhes de cada método:

a) *Identificação Única de Computadores*: Um dos objetivos do sistema é garantir que cada máquina na rede possua uma identificação única. Para isso, foi desenvolvido um método específico que utiliza o endereço IPv4 local como identificador. Esse método aproveita a funcionalidade da classe `Dns donamespace System.Net`, que oferece ferramentas

para lidar com operações relacionadas ao DNS. Primeiramente, o método obtém o *host* da máquina. Em seguida, percorre os endereços retornados até encontrar o IPv4, que é então retornado como a chave identificadora única da máquina.

b) *Preparação dos Dados para a API*: Outro método essencial da classe é responsável por reunir os dados das métricas e prepará-los no formato JSON para exposição via API. Este método é implementado como *static* e *private*, sendo acessível apenas dentro da própria classe. Ele inicia instanciando a classe *SystemMetrics.cs*, a qual realiza o cálculo das métricas. Cada cálculo é executado individualmente e os valores retornados são armazenados em variáveis temporárias. Para converter esses valores em uma estrutura JSON, utiliza-se o *JsonSerializer* do namespace *System.Text.Json*. A opção *WriteIndented* é ativada para melhorar a legibilidade dos dados formatados.

c) *Configuração do Servidor Web*: A configuração do servidor web é realizada em outro método dedicado, que utiliza a interface *IHostBuilder*. Este método configura o servidor para escutar em uma URL específica, construída a partir do endereço IPv4 obtido e de uma porta fixa (5000). O *endpoint /metrics* é definido para expor as informações coletadas. Durante essa configuração, o método especifica que a aplicação deve ser executada como um serviço do Windows, permitindo sua execução em segundo plano. Após a construção do *host* com *Build()*, o servidor é iniciado com *Run()*, mantendo-se em execução contínua até que seja encerrado manualmente ou em caso de falha.

d) *Método Main*: O método *Main* é o ponto de entrada da aplicação. Ele combina os métodos previamente descritos para configurar o servidor e iniciar a aplicação. Este método chama o método de identificação para obter o IPv4 local, passa essa informação ao método de configuração do servidor e, em seguida, inicia a execução do *host* configurado.

2) *Classe SystemMetrics.cs: Coleta e Cálculo de Métricas*: A classe *SystemMetrics.cs* é responsável pela coleta das métricas de desempenho do sistema, implementando três métodos dedicados às métricas de CPU, memória e disco. A seguir, uma descrição detalhada:

a) *Uso de CPU*: Este método coleta o uso total de CPU, combinando todos os núcleos. A coleta é feita com a classe *PerformanceCounter* do namespace *System.Diagnostics*, passando os parâmetros "*Processor*", "*% Processor Time*" e "*_Total*". Para obter valores precisos, o método chama *NextValue()* duas vezes, com um intervalo de tempo entre as chamadas, e retorna o resultado formatado como *string* JSON com duas casas decimais.

b) *Uso de Memória*: Este método calcula o total de memória utilizada, disponível e o percentual de uso. Ele utiliza contadores específicos, como "*Committed Bytes*" e "*Available MBytes*". Após coletar os valores brutos, o método realiza cálculos para converter as informações para megabytes e calcula o percentual de uso com base nos valores obtidos. Os resultados são então formatados como JSON.

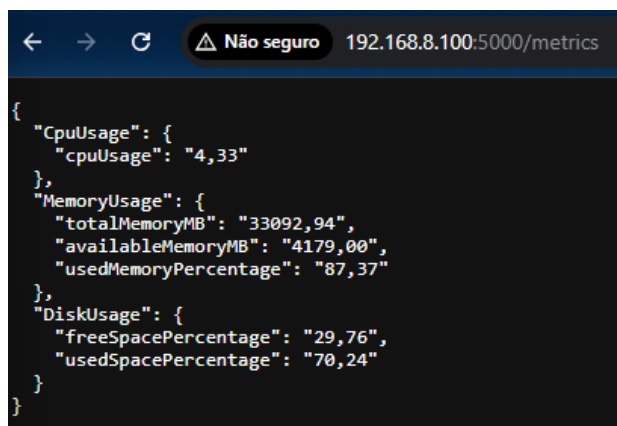
c) *Uso de Disco*: O uso de disco é calculado a partir do espaço livre disponível, utilizando os parâmetros "*Logical-Disk*", "*% Free Space*" e "*_Total*". O valor de espaço utilizado

é derivado da diferença entre 100% e a porcentagem de espaço livre. Os resultados são retornados em formato JSON.

3) *Registro como Serviço no Windows*: Para assegurar a inicialização automática do sistema, ele foi configurado como um serviço do Windows. O processo incluiu os seguintes passos:

- 1) Publicação do executável no Visual Studio, gerando o arquivo na pasta *publish*.
- 2) Registro do executável como serviço via linha de comando com *sc create*.
- 3) Configuração do serviço para inicialização automática com *sc config start= auto*.
- 4) Adição de uma regra personalizada ao *firewall* do Windows para liberar a porta 5000.

4) *Resultados e Demonstração*: A aplicação final expõe as métricas do sistema em tempo real por meio de um *endpoint* REST acessível na URL <http://<IPv4>:5000/metrics>. A Fig. 12 ilustra o JSON retornado pela API, destacando as informações coletadas e formatadas.



```
{
  "CpuUsage": {
    "cpuUsage": "4,33"
  },
  "MemoryUsage": {
    "totalMemoryMB": "33092,94",
    "availableMemoryMB": "4179,00",
    "usedMemoryPercentage": "87,37"
  },
  "DiskUsage": {
    "freeSpacePercentage": "29,76",
    "usedSpacePercentage": "70,24"
  }
}
```

Fig. 12. Resultado da requisição HTTP para a URL <http://192.168.8.100:5000/metrics>. Fonte: Autoria própria.

D. Desenvolvimento da aplicação de RA

O desenvolvimento da aplicação de RA foi realizado na plataforma Unity, exigindo um conjunto de configurações específicas para habilitar os recursos necessários. A seguir, cada etapa do processo é descrita com detalhamento técnico e justificativa das decisões tomadas.

1) *Criação do Projeto no Unity*: Para iniciar o desenvolvimento, foi necessário instalar o Unity Hub, uma ferramenta que permite gerenciar versões do Unity e projetos. O projeto foi criado na versão 6000.0.27f1, escolhida devido à sua compatibilidade com pacotes de RA modernos e estabilidade em sistemas Android. O template 3D foi selecionado, dado que proporciona maior controle sobre a renderização e elementos visuais tridimensionais.

2) *Instalação do Pacote AR Foundation*: O AR Foundation foi instalado por meio do Package Manager, sendo este o núcleo que habilita a funcionalidade de RA no Unity. Ele fornece uma camada de abstração sobre os sistemas de RA subjacentes, como ARCore e ARKit, permitindo maior interoperabilidade entre plataformas.

a) *Configuração da Plataforma para Android:* Para suportar dispositivos móveis, a plataforma do projeto foi alterada para Android em File > Build Settings. Essa troca é essencial, pois o Android requer configurações específicas para acesso ao hardware e bibliotecas de RA. A mudança garante que o aplicativo seja otimizado para dispositivos móveis, incluindo a compatibilidade com sensores e APIs de RA.

b) *Ativação do ARCore:* No Player Settings > XR Plug-in Management, o ARCore foi habilitado, pois ele é a principal API de RA suportada pelo Android. Essa etapa informa ao Unity qual provedor de RA utilizar, permitindo a integração com os sensores do dispositivo e funcionalidades avançadas, como detecção de superfícies e rastreamento de movimentos.

c) *Configuração do Backend de Renderização:* O backend de renderização foi ajustado para OpenGL ES 3.0, garantindo um desempenho otimizado em dispositivos móveis. O OpenGL ES 3.0 oferece suporte a recursos avançados de renderização com maior eficiência energética, crucial para aplicações de RA que exigem atualizações constantes do ambiente tridimensional.

d) *Definição do Minimal API Level:* O *Minimal API Level* foi configurado para 24 (Android 7.0 *Nougat*), pois versões anteriores não suportam adequadamente as bibliotecas de RA modernas. Essa configuração assegura que o aplicativo possa utilizar as últimas melhorias de hardware e software.

3) *Integração com Vuforia:* O Vuforia foi incorporado ao projeto para realizar a leitura de URLs em QR Codes. Como a integração nativa foi removida do Unity 2019.3, foi necessário baixar o Unity Package do site oficial da Vuforia e importá-lo como um Custom Package em Assets > Import Package > Custom Package. Essa integração adiciona suporte à câmera de RA e aos métodos de detecção de imagens.

a) *Configuração da Licença da Vuforia:* Para utilizar a Vuforia, foi criada uma conta no site oficial e gerada uma licença básica. A chave de licença (*App License Key*) foi adicionada na configuração do Vuforia Engine no Unity. Essa etapa é crucial, pois sem uma chave válida, os recursos da Vuforia não podem ser acessados, comprometendo a funcionalidade do projeto.

b) *Adição da AR Camera:* Após a configuração da licença, foi adicionada uma *AR Camera* ao projeto, disponível no pacote da Vuforia Engine. A *AR Camera* substitui a câmera padrão do Unity, fornecendo funcionalidades específicas de RA, como rastreamento de posição e integração com bibliotecas de detecção.

E. Hierarquia do Projeto de RA

A Fig. 13 ilustra a hierarquia da cena principal do projeto de RA, configurada no Unity. Cada elemento dessa hierarquia desempenha um papel específico na funcionalidade e na interação do sistema. Nesta seção, detalharemos cada componente e sua função, destacando as decisões técnicas que sustentam a estruturação da cena.

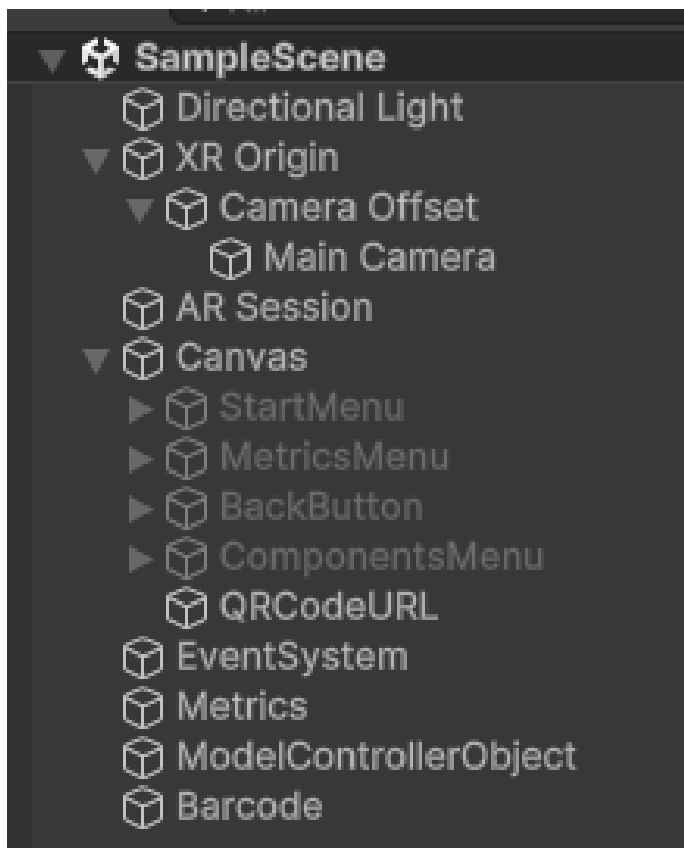


Fig. 13. Hierarquia do projeto de RA. Fonte: Autoria própria.

1) *Organização Geral da Hierarquia:* A estrutura hierárquica na Unity segue um modelo de árvore, onde objetos são organizados em uma relação pai-filho. Isso permite um controle eficiente e modular de cada elemento da cena, promovendo flexibilidade e escalabilidade do projeto.

A raiz da hierarquia é composta por objetos principais que agrupam funcionalmente os elementos da aplicação, como o *XR Origin*, o *AR Session*, os modelos 3D e os elementos de interface (UI).

a) *XR Origin:* O *XR Origin* é o elemento central para a interação entre o ambiente físico e virtual. Ele define a posição e orientação da câmera, que, por sua vez, rastreia os movimentos do usuário e garante o alinhamento correto dos objetos virtuais com o espaço real. A sua configuração inclui os seguintes subcomponentes:

- **Main Camera:** Responsável por renderizar a cena para o usuário. É configurada para captar imagens em tempo real do ambiente físico e sobrepor os objetos virtuais.
- **AR Camera Manager:** Gerencia as funções específicas de RA, como detecção de superfícies e âncoras para posicionamento dos objetos 3D.
- **AR Pose Driver:** Sincroniza os movimentos do dispositivo com os movimentos da câmera virtual, assegurando a precisão no rastreamento.

b) *Gerenciamento de Sessões - AR Session:* O componente *AR Session* é essencial para iniciar e controlar os recursos de RA. Ele gerencia a inicialização dos serviços de RA e a comunicação com os sistemas de rastreamento subjacentes, como o ARCore (Google) e o ARKit (Apple). Isso

inclui a detecção de superfícies, estimativa de luz ambiente e detecção de movimento.

c) **Iluminação Global - Directional Light:** A luz direcional, configurada no *Directional Light*, é crucial para criar uma experiência visualmente realista. Este componente simula uma fonte de luz distante (como o Sol), permitindo que sombras sejam projetadas de maneira consistente em todos os objetos da cena. As configurações de intensidade e cor foram ajustadas para criar um ambiente harmônico que realça os detalhes dos modelos 3D.

2) **Interface de Usuário (UI):** A interface de usuário é organizada no componente *Canvas*, que contém todos os elementos visuais interativos da aplicação. O *Canvas* está estruturado hierarquicamente para permitir a fácil manipulação e gerenciamento de menus, botões e textos.

a) **Menus e Controles:** Os menus principais incluem:

- **Menu Principal:** Oferece opções de navegação inicial, como acessar métricas do sistema ou manipular o modelo 3D.
- **Tela de Métricas:** Exibe os dados de uso da CPU, memória e disco, atualizados em tempo real.
- **Tela de Interação com o Modelo 3D:** Permite ao usuário explorar o modelo com funcionalidades de zoom, rotação e visualização detalhada de componentes.

b) **Eventos de Interação:** O *Event System* é o núcleo responsável por processar as interações do usuário, como cliques ou toques na tela. Ele garante que cada elemento interativo, como botões, responda de maneira precisa às ações realizadas.

3) **Objetos Virtuais e Scripts:** Além dos elementos descritos, a hierarquia inclui os objetos principais que contêm os modelos 3D e os scripts que governam sua funcionalidade:

- **Metrics Object:** Responsável pela coleta e exibição de dados do sistema monitorado.
- **ModelControllerObject:** Gerencia o posicionamento e a interação com o modelo 3D.
- **SimpleBarcodeScanner:** Integra o *Vuforia* para decodificar QR Codes e iniciar a comunicação com a aplicação console.

4) **Vantagens da Organização Estrutural:** A hierarquia projetada garante modularidade e clareza na organização dos elementos. Cada componente desempenha uma função bem definida, facilitando o desenvolvimento, teste e manutenção do projeto. Essa abordagem também permite a futura expansão da aplicação, seja para adicionar novos modelos, melhorar a interface ou integrar funcionalidades avançadas de RA.

F. Modelo 3D

O modelo 3D utilizado no protótipo foi obtido de uma plataforma de distribuição gratuita de modelos 3D, respeitando os termos de uso definidos pelo site. Após o *download*, foram realizados pequenos ajustes no *Blender* para adaptar o modelo às necessidades específicas do projeto, como a correção de proporções, aplicação de texturas ou alterações em detalhes estruturais. Esses ajustes garantiram uma melhor integração do modelo ao contexto da aplicação e sua funcionalidade dentro do ambiente de RA.

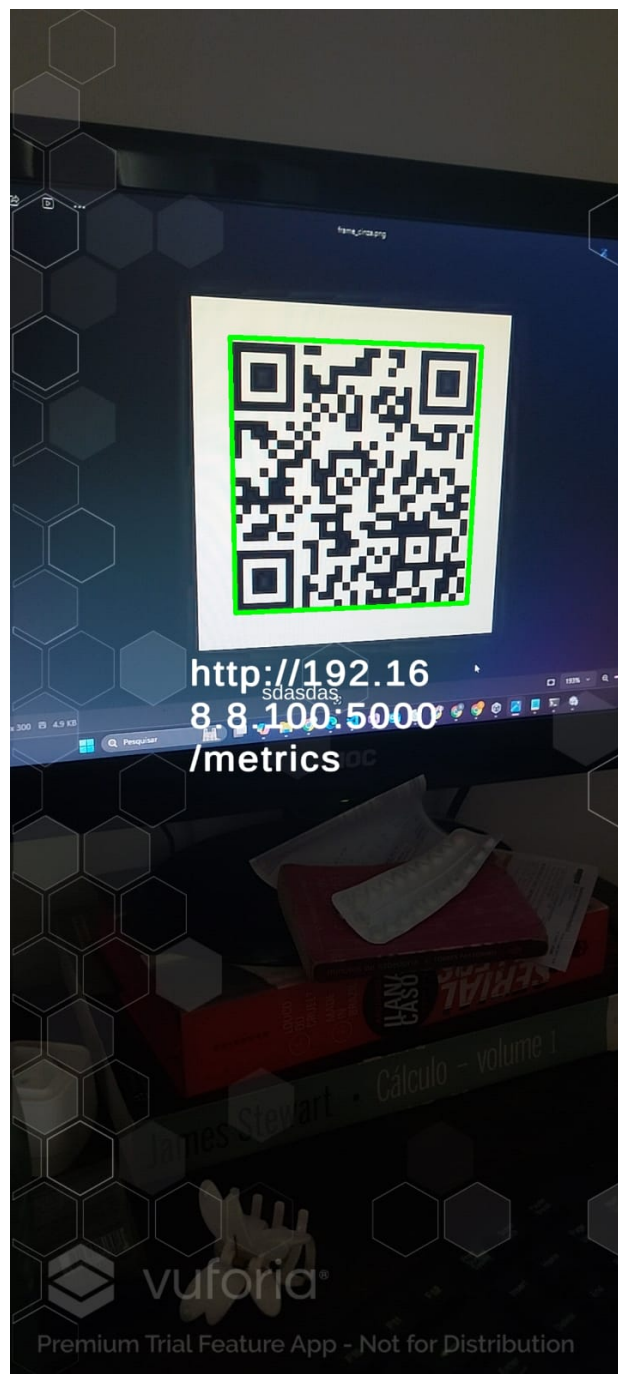


Fig. 14. Tela de Escaneamento do QR Code. Fonte: Autoria própria.

G. Interface de Usuário (UI)

A Interface de Usuário foi projetada para proporcionar uma navegação simples e intuitiva. A seguir, descreve-se o fluxo e os elementos presentes nas diferentes telas do protótipo, com suporte visual fornecido por imagens.

1) **Tela de Escaneamento do QR Code:** Descrição: Esta tela inicial apresenta uma interface limpa, centrada na funcionalidade de escaneamento de QR Codes como mostrado na Fig.14. A câmera é ativada automaticamente ao abrir a aplicação, e o layout destaca o espaço central para enquadrar o QR Code. Funcionalidade: Após o QR Code ser escaneado, o sistema



Fig. 15. Menu principal da aplicação. Fonte: Autoria própria.

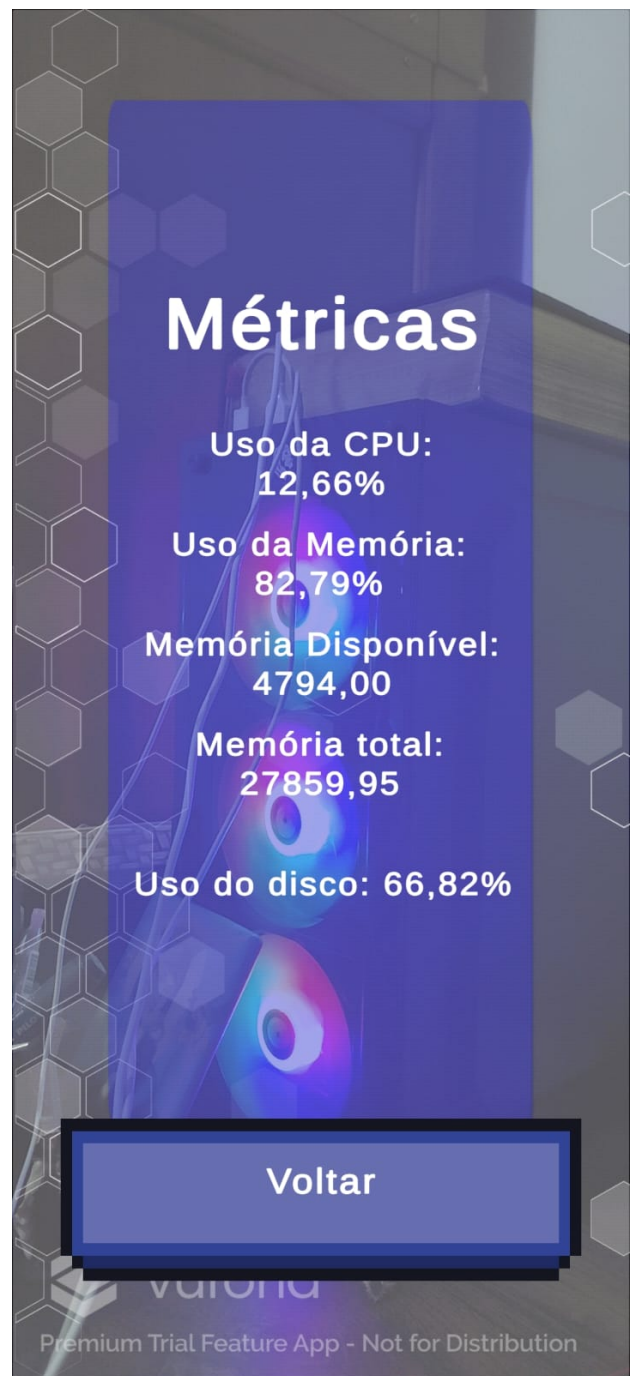


Fig. 16. Tela de Monitoramento. Fonte: Autoria própria.

extraí a *Uniform Resource Locator* (URL) e redireciona o usuário para a tela de menu principal.

2) *Tela de Menu Principal*: Descrição: Contém três botões principais: Monitorar: Redireciona para a tela de monitoramento de métricas. Modelo 3D: Leva à visualização interativa do modelo 3D do computador. Voltar: Retorna o usuário para a tela de escaneamento do QR Code. Estética: Os botões possuem cores contrastantes para fácil identificação e estão centralizados para facilitar a interação, especialmente em dispositivos móveis, conforme pode ser observado na Fig. 15.

3) *Tela de Monitoramento*: Descrição: Exibe um card com as métricas do sistema (como uso de CPU, memória e disco), apresentadas em formato legível. Funcionalidades: Um botão de Voltar, posicionado no canto inferior da tela, retorna ao menu principal. As métricas são atualizadas em tempo real a partir da URL obtida pelo QR Code. Design: O card é retangular e possui bordas levemente arredondadas conforme é possível visualizar na Fig. 16

4) *Tela de Modelo 3D Interativo*: Descrição: Apresenta o modelo 3D de um computador como pode ser visto na Fig.17, com opções interativas. O usuário pode movimentar, girar e

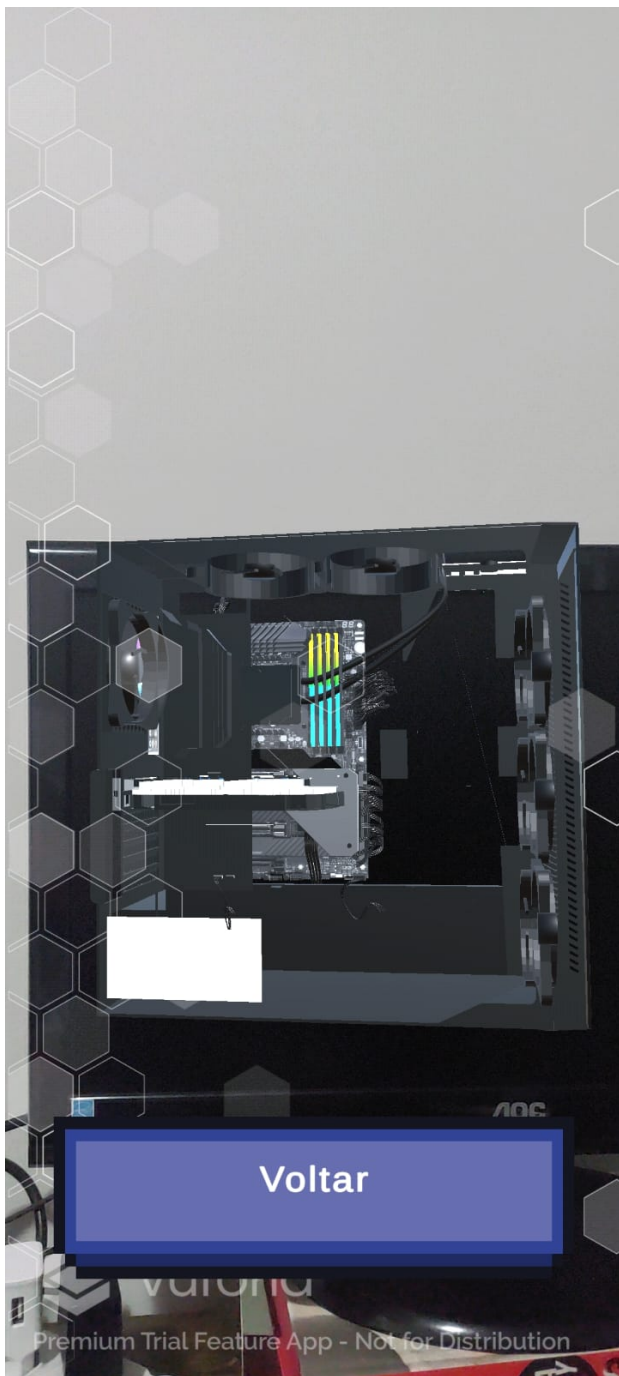


Fig. 17. Tela de Modelo 3D Interativo. Fonte: Autoria própria.



Fig. 18. Tela de dados da GPU. Fonte: Autoria própria.

aplicar zoom no modelo usando gestos comuns em dispositivos móveis. Ao clicar em componentes específicos do modelo (como a CPU ou GPU), abre-se um *card* com informações sobre o componente, incluindo nome e modelo, como por ser analisada pelas Fig.18, Fig.19 e Fig.20. Funcionalidades: O *card* informativo possui um botão verde com uma seta, que fecha o *card* e retorna à tela interativa do modelo 3D. Um botão de Voltar no canto inferior retorna o usuário ao menu principal. Design: A interface é minimalista, permitindo que o modelo 3D seja o foco principal. Os botões são grandes e visíveis para facilitar a manipulação.

H. Testes e Depuração da Aplicação

Os testes e a depuração da aplicação de RA foram conduzidos em três etapas distintas, garantindo a validação do funcionamento do protótipo em diferentes ambientes: a *Game View* da Unity, testes diretos em dispositivos Android via USB, e instalação manual do arquivo APK em dispositivos móveis.

1) *Testes na Unity via Game View*: A *Game View* foi utilizada como ponto de partida para validar funcionalidades iniciais da aplicação. Essa abordagem permitiu testar:

- O comportamento dos *scripts* em C# implementados.



Fig. 19. Tela de dados do processador. Fonte: Autoria própria.



Fig. 20. Tela de dados da memória RAM. Fonte: Autoria própria.

- A interação e disposição dos elementos da interface de usuário (UI).
- O posicionamento e comportamento de objetos 3D no ambiente virtual.
- A integração com as bibliotecas AR Foundation e Vuforia Engine.

Apesar de eficiente para identificar problemas em funcionalidades básicas, os testes na *Game View* não simulam completamente o comportamento da aplicação em dispositivos reais, como a interação com câmeras e sensores de movimento.

2) *Testes em Dispositivos Android via USB:* A segunda etapa envolveu a execução da aplicação diretamente em dispositivos Android conectados via USB. O processo incluiu:

- 1) Configuração da plataforma para Android nas *Build Settings* da Unity.
- 2) Habilitação do modo desenvolvedor e depuração USB no dispositivo.
- 3) Implantação da aplicação por meio da opção *Build and Run* ou *Unity Remote*.

Esses testes foram fundamentais para validar o comportamento da aplicação no hardware real, garantindo a:

- Detecção de QR Codes pela câmera do dispositivo.
- Fluidez na exibição de métricas em tempo real e interação com objetos 3D.
- Compatibilidade com sensores e APIs do Android.

3) *Testes via Instalação do APK*: Para simular as condições de uso final, a aplicação foi exportada como um arquivo APK, transferido e instalado manualmente em dispositivos móveis. As etapas para realizar esses testes foram:

- 1) Exportação do projeto na Unity utilizando a opção *Build Settings*.
- 2) Transferência do arquivo APK para o dispositivo via USB ou serviços de compartilhamento de arquivos.
- 3) Instalação do aplicativo após ativar a opção *Instalar aplicativos de fontes desconhecidas*.

Nessa fase, foram avaliados aspectos como:

- Estabilidade e inicialização do aplicativo.
- Responsividade da interface e adequação a diferentes resoluções de tela.
- Precisão na leitura de QR Codes e desempenho geral.

4) *Benefícios do Processo*: A combinação dos métodos de teste garantiu:

- Identificação ágil de erros em etapas iniciais, por meio da *Game View*.
- Validação completa das funcionalidades em hardware real.
- Melhoria contínua, com ajustes em tempo hábil, como calibragem da UI e otimização do desempenho.

5) *Considerações Finais*: Esse processo de testes foi essencial para assegurar a funcionalidade, estabilidade e usabilidade da aplicação, culminando em um protótipo confiável e robusto. A abordagem iterativa permitiu identificar e corrigir problemas em diferentes estágios do desenvolvimento, contribuindo para a qualidade final do sistema de RA.

V. LIMITAÇÕES E TRABALHOS FUTUROS

A. Limitações da Solução Atual

Embora o protótipo desenvolvido tenha demonstrado viabilidade e funcionalidade no contexto proposto, algumas limitações foram identificadas, as quais demandam futuras melhorias para ampliar a aplicabilidade da solução.

1) *Compatibilidade com Sistemas Operacionais*: Atualmente, o aplicativo console utilizado para coletar métricas de hardware está limitado ao sistema operacional Windows. Isso se deve à utilização de APIs específicas para este ambiente. Para estender a compatibilidade a outros sistemas operacionais, como Linux e macOS, será necessário:

- Reescrever parte do código para utilizar APIs multiplataforma ou específicas dos sistemas-alvo.
- Pesquisar métodos equivalentes para capturar métricas de desempenho em outros ambientes.
- Implementar soluções para iniciar o aplicativo automaticamente no momento da inicialização do computador, independentemente do sistema operacional.

2) *Compatibilidade com Outros Dispositivos Móveis*: A aplicação de RA foi desenvolvida exclusivamente para dispositivos Android, utilizando AR Foundation e Vuforia Engine, que atualmente possuem maior suporte para esta plataforma. Entretanto, a adaptação da solução para outros sistemas móveis, como iOS, é necessária para:

- Atingir um público mais amplo e diversificado.
- Garantir a funcionalidade em dispositivos de diferentes fabricantes e sistemas.
- Resolver problemas de compatibilidade, como APIs específicas e requisitos de hardware.

B. Trabalhos Futuros

Para o aprimoramento e evolução do projeto, destacam-se as seguintes sugestões de trabalhos futuros:

1) *Criação de Modelos 3D Realistas*: A inclusão de modelos tridimensionais que representem máquinas reais, além dos computadores atualmente utilizados, traria maior aplicabilidade ao protótipo no contexto de infraestruturas computacionais. Futuras expansões podem incluir:

- Computadores pessoais com maior detalhamento.
- Modelagem de notebooks, servidores e outros equipamentos utilizados em infraestruturas de TI.
- Adição de periféricos e dispositivos relacionados, como roteadores, *switches* e unidades de armazenamento.

2) *Aplicação Multiplataforma*: A adaptação da aplicação para outros dispositivos, como óculos de RA e *tablets*, pode melhorar a experiência do usuário e expandir as possibilidades de uso da solução. Isso inclui:

- Ajustes na interface para diferentes tamanhos e resoluções de tela.
- Testes de desempenho em dispositivos com hardware e capacidades gráficas variadas.
- Exploração de novos paradigmas de interação, como comandos por voz e gestos.

3) *Melhoria do Sistema de Monitoramento*: A integração de novas funcionalidades para análise e visualização de dados, como gráficos em tempo real e alertas automáticos, traria maior valor à aplicação para fins de diagnóstico e manutenção. Além disso, a expansão da coleta de métricas pode incluir:

- Monitoramento de temperaturas e níveis de consumo energético.
- Estatísticas de utilização por período.
- Integração com sistemas de gestão de infraestrutura existentes.

C. Considerações Finais

Essas limitações e propostas de melhorias destacam o potencial de crescimento e impacto da solução desenvolvida. Ao superar essas barreiras e implementar as sugestões apresentadas, a aplicação poderá atingir níveis mais altos de funcionalidade, eficiência e abrangência no contexto da Indústria 4.0.

VI. CONCLUSÃO

Este trabalho apresentou o desenvolvimento de um protótipo de RA voltado para o monitoramento e manutenção de infraestruturas computacionais, com foco na Indústria 4.0. A aplicação demonstrou como a integração de tecnologias de RA pode contribuir para a eficiência operacional, permitindo a sobreposição de informações em tempo real sobre o ambiente físico, a visualização interativa de modelos 3D e a interação direta com os componentes computacionais.

O uso da plataforma Unity, aliado a ferramentas como AR Foundation e Vuforia Engine, possibilitou a criação de uma solução funcional que abrange desde a leitura de métricas até a manipulação de modelos tridimensionais. A aplicação foi testada com sucesso em dispositivos móveis Android, com diferentes abordagens para depuração, incluindo a *Game View*, execução via *USB* e instalação direta do arquivo .apk em dispositivos móveis.

Apesar dos avanços, o projeto apresenta limitações que restringem sua aplicação a determinados ambientes, como a dependência do sistema operacional Windows para o aplicativo console e a compatibilidade exclusiva com dispositivos Android. Essas restrições apontam direções claras para trabalhos futuros, como a expansão para outros sistemas operacionais, a adaptação da aplicação para dispositivos iOS e a inclusão de modelos 3D que representem equipamentos reais, como notebooks, servidores e dispositivos de rede.

Os resultados obtidos reforçam o potencial da RA no contexto industrial, destacando sua capacidade de transformar o monitoramento e manutenção de infraestruturas computacionais. A continuidade deste projeto, com a superação das limitações identificadas e a implementação das propostas de trabalhos futuros, poderá consolidar ainda mais a utilidade desta solução e sua aplicabilidade em diferentes cenários da Indústria 4.0.

Em suma, este trabalho não apenas evidencia a viabilidade técnica da aplicação de RA neste domínio, mas também oferece uma base sólida para avanços que atendam às necessidades crescentes de integração e automação nas infraestruturas computacionais industriais.

REFERÊNCIAS

- [1] Aleksić, V. and Politis, D. (2020). The characteristics of virtual reality usage in educational systems. In *2020 International Conference on INnovations in Intelligent Systems and Applications (INISTA)*, pages 1–5.
- [2] Bajic, B., Rikalovic, A., Suzic, N., and Piuri, V. (2021). Industry 4.0 implementation challenges and opportunities: A managerial perspective. *IEEE Systems Journal*, 15(1):546–559.
- [3] Birais, P. H. and Rafikova, E. (2023). Augmented reality system for immersive mobile robot simulation and trajectory estimation. In *2023 Latin American Robotics Symposium (LARS), 2023 Brazilian Symposium on Robotics (SBR), and 2023 Workshop on Robotics in Education (WRE)*, pages 17–22.
- [4] Bondin, A. and Zammit, J. P. (2022). A new age for plant maintenance: Making use of augmented reality to improve maintenance of production assets. In *2022 IEEE International Conference on Technology Management, Operations and Decisions (ICTMOD)*, pages 1–6.
- [5] Choe, J. and Seo, S. (2020). A 3d real object recognition and localization on slam based augmented reality environment. In *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 745–746.
- [6] Fawad, M., Salamak, M., Hanif, M. U., Koris, K., Ahsan, M., Rahman, H., Gerges, M., and Salah, M. M. (2024). Integration of bridge health monitoring system with augmented reality application developed using 3d game engine—case study. *IEEE Access*, 12:16963–16974.
- [7] Fernández-Mireles, V., Martínez-Cruz, M., Cardenas-Juarez, M., Montalvo-Rodríguez, , and Alemán-Mendiola, V. (2023). Design and implementation of an augmented reality-based app for assembly lines in industry 4.0. In *2023 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, volume 7, pages 1–6.
- [8] Ferreira, M. R. (2021). Indústria 4.0: Análise da viabilidade do uso de métodos e ferramentas baseadas em ra como estratégia para diminuição do tempo de parada durante a configuração (set-up) de máquinas cnc e cmm na indústria manufatureira. Trabalho acadêmico. Detalhes adicionais podem ser adicionados caso necessários, como instituição ou local de publicação.
- [9] Foundation, B. (2024). Blender history. Accessed: 2024-12-06.
- [10] Gonsheer, I. and Lei, Z. (2021). Prototype of force feedback tool for mixed reality applications. In *2021 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pages 508–509.
- [11] Information Technology Intelligence Consulting (2023). 2023 Global Server Hardware, Server OS Reliability Report. Acesso em: 6 dez. 2024.
- [12] Karaiskos, V., Zinas, N., Gkamas, T., Karolos, I. A., Pikridas, C., Vrettos, N., Tsioukas, V., and Kontogiannis, S. (2022). Proposed industry 4.0 maintenance framework for critical and demanding infrastructures and processes. In *2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNM)*, pages 1–5.
- [13] Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., and Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, 6(4):239–242.
- [14] Liang, Y.-W. and Huang, Y.-H. (2023). Exploration of user experience in mixed reality for product virtual interaction and display. In *2023 IEEE 6th International Conference on Knowledge Innovation and Invention (IC-KII)*, pages 404–409.
- [15] Lu, Y. and Yihao, L. (2021). Interactive augmented reality application design based on mobile terminal. In *2021 IEEE 5th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, volume 5, pages 6–12.
- [16] Maio, R., Marques, B., Santos, A., Ramalho, P., Almeida, D., Dias, P., and Santos, B. S. (2023). Real-time data monitoring of an industry 4.0 assembly line using pervasive

- augmented reality: First impressions. In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 414–417.
- [17] Microsoft (2024a). Histórico de versões do c#. Acesso em: 6 dez. 2024.
- [18] Microsoft (2024b). Introdução ao .net. Acesso em: 6 dez. 2024.
- [19] Microsoft (2024c). Visão geral do c#. Acesso em: 6 dez. 2024.
- [20] Mirza, T., Tuli, N., and Mantri, A. (2022). Virtual reality, augmented reality, and mixed reality applications: Present scenario. In *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pages 1405–1412.
- [21] MMORPG, W. (2024). Unity development history and the influence of this game engine on the game development. *Medium*. Accessed: 2024-12-08.
- [22] Mustapha, S., Yang, Y. B., and Salahuddin, M. A. (2021). Virtual reality based application for autistic adolescents in malaysia. In *2021 IEEE 11th IEEE Symposium on Computer Applications Industrial Electronics (ISCAIE)*, pages 169–173.
- [23] Pooja and Sood, S. K. (2025). Quantum-inspired metaheuristic algorithms for industry 4.0: A scientometric analysis. *Engineering Applications of Artificial Intelligence*, 139. Cited by: 0.
- [24] PTC Inc. (2024). Vuforia engine developer library. Accessed: 2024-12-08.
- [25] Reddy Gurralla, K. and Choudhary, S. (2024). Analysis of industry 4.0 barriers: A fuzzy demantel approach. In *2024 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, pages 266–272.
- [26] Rikalovic, A., Suzic, N., Bajic, B., and Piuri, V. (2022). Industry 4.0 implementation challenges and opportunities: A technological perspective. *IEEE Systems Journal*, 16(2):2797–2810.
- [27] Rybalskii, I., Kruusamäe, K., Singh, A. K., and Schlund, S. (2024). An augmented reality interface for safer human-robot interaction in manufacturing. *IFAC-PapersOnLine*, 58(19):581–585. 18th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2024.
- [28] Technologies, U. (2024a). *AR Foundation Documentation*. Accessed: 2024-12-08.
- [29] Technologies, U. (2024b). *XR Interaction Toolkit Documentation*. Accessed: 2024-12-08.
- [30] Wahab, R. M., Taharudin, A. I., Hemdi, A. R., Noor, N. M. M., Zubair, A. F., and Azmi, N. N. (2023). Interactive visualization to enhance learning experienced on machining process of edm wire cut through augmented reality (ar). In *2023 IEEE 12th International Conference on Engineering Education (ICEED)*, pages 1–6.



Marcelo S. de Castro graduou-se em Engenharia Elétrica pela Universidade Federal de Juiz de Fora (1992), com mestrado em Engenharia Elétrica pela Universidade Estadual de Campinas (1995) e doutorado em Engenharia Elétrica pela UnB (2010). Docente Associado da Universidade Federal de Goiás, tendo ingressado em 1996. Possui experiência na área de engenharia de redes, BI e educação em engenharia.



Gleycykelly Syssy Indymayer Carnot Amaro é graduanda em Engenharia de Computação na Universidade Federal de Goiás. Trabalha atualmente como desenvolvedora full stack em Goiânia - GO. Possui experiência em desenvolvimento de software para sistemas web.