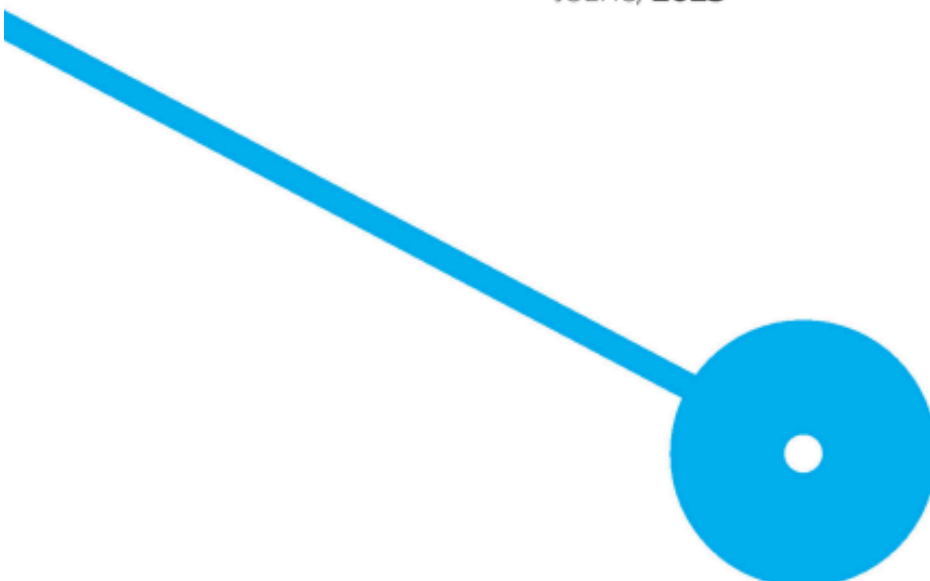


Atende.AI: Um Serious Game em Realidade Virtual com IA para Treinamento de Atendimento ao Público

Leandro Silva

JULHO/2025



Atende.AI: Um Serious Game em Realidade Virtual com IA para Treinamento de Atendimento ao Público

Leandro Silva
8240471

Orientador(es)

Doutor Fábio André Souto Da Silva

Relatório de Projeto apresentado para cumprimento dos requisitos necessários à obtenção do grau de Licenciado em Sistemas de informação para gestão pela Escola Superior de Tecnologia e Gestão do Instituto Politécnico do Porto.

JULHO/2025

Declaração de Integridade

Eu, Leandro Silva, estudante nº 8240471, da Licenciatura de Sistemas de informação para gestão da Escola Superior de Tecnologia e Gestão do Instituto Politécnico do Porto, declaro que não fiz plágio nem auto-plágio, pelo que o trabalho intitulado "Atende.AI: Um Serious Game em Realidade Virtual com IA para Treinamento de Atendimento ao Público" é original e da minha autoria, não tendo sido usado previamente para qualquer outro fim. Mais declaro que todas as fontes usadas estão citadas, no texto e na bibliografia final, segundo as regras de referência adotadas na instituição.

Resumo

Este trabalho detalha o desenvolvimento do "Atende.AI", um protótipo de Serious Game em Realidade Virtual (RV) com Inteligência Artificial (IA) para treinamento de atendimento ao público. O objetivo foi criar uma experiência imersiva onde o usuário aprimora habilidades de comunicação e procedimentos de atendimento ao interagir com um cliente virtual (NPC) cujos diálogos são gerados em tempo real por IA. A arquitetura da solução é do tipo cliente-servidor, com a aplicação em RV (cliente) desenvolvida em Unity e uma API em Python (servidor) que orquestra os modelos de IA para reconhecimento de voz (STT), geração de diálogo (LLM) e síntese de voz (TTS). Os resultados validam o protótipo como uma ferramenta eficaz, simulando diferentes cenários de atendimento e fornecendo um sistema de *feedback* que avalia o desempenho do usuário, demonstrando o potencial da sinergia entre RV e IA para a capacitação profissional.

Palavras-chave: *Serious Games*, Realidade Virtual, Inteligência Artificial, Treinamento Profissional, Atendimento ao Público.

Abstract

This work details the development of "Atende.AI," a prototype of a Virtual Reality (VR) Serious Game with Artificial Intelligence (AI) for public-facing customer service training. The objective was to create an immersive experience where the user improves communication skills and service procedures by interacting with a virtual customer (NPC) whose dialogues are generated in real-time by an AI system. The solution employs a client-server architecture, with the VR application (client) developed in Unity and a Python API (server) orchestrating AI models for speech-to-text (STT), dialogue generation (LLM), and text-to-speech (TTS). The results validate the prototype as an effective tool, simulating different customer service scenarios and providing a feedback system that evaluates user performance, demonstrating the potential of the synergy between VR and AI for professional training.

Keywords: Serious Games, Virtual Reality, Artificial Intelligence, Professional Training, Customer Service

Lista de figuras

Figura 1 -Diagrama de Arquitetura Geral da Solução Atende.AI.....	18
Figura 2 - Diagrama de sequência do fluxo de atendimento de alto nível.....	22
Figura 3 - Representação do armazém e o sistema de localização de encomendas.....	24
Figura 4 - Exemplo do Documento de Identificação e da Carta da Encomenda.....	25
Figura 5 - Integração das ferramentas interativas: leitor, balança e monitor de atendimento.....	26
Figura 6 - Balcão de atendimento com a barreira física e o sistema de intercomunicador implementados.....	31
Figura 7 - Fluxo de Dados na API Modular.....	33
Figura 8 - Trecho do histórico de conversa com o 'Cliente Colaborativo' em formato JSON	38
Figura 9 - Tela de feedback para o cenário 'Colaborativo'.....	38
Figura 10 - Trecho do histórico de conversa com o 'Cliente Inseguro' em formato JSON.	40
Figura 11 - Tela de feedback para o cenário 'Inseguro'	40
Figura 12 - Trecho do histórico de conversa com o 'Cliente Distraído' em formato JSON.....	42
Figura 13 - Avaliação de desempenho para o cenário 'Distraído'	42

Abreviaturas

API - *Application Programming Interface* (Interface de Programação de Aplicações)

CHA - Conhecimentos, Habilidades e Atitudes

CTT - Correios de Portugal

DNT - Diagnóstico de Necessidade de Treinamento

FSM - *Finite State Machine* (Máquina de Estados Finitos)

GPU - *Graphics Processing Unit* (Unidade de Processamento Gráfico)

HMD - *Head-Mounted Display*

HTTP - *Hypertext Transfer Protocol*

IA - Inteligência Artificial

ITS - *Intelligent Tutoring System* (Sistema de Tutoria Inteligente)

JSON - *JavaScript Object Notation*

LLM - *Large Language Model* (Grande Modelo de Linguagem)

LTS - *Long-Term Support*

NPC - *Non-Player Character* (Personagem não-jogável)

RV - Realidade Virtual

STT - *Speech-to-Text* (Reconhecimento de Fala)

TTS - *Text-to-Speech* (Síntese de Fala)

Sumário

Resumo.....	IV
Abstract.....	V
Lista de figuras.....	VI
Abreviaturas.....	VII
1. Introdução.....	1
1.1 Contextualização.....	1
1.2 Limitações das abordagens tradicionais.....	1
1.3 Objetivo.....	2
1.4 Metodologia de Trabalho.....	3
1.5 Plano de Trabalhos.....	3
2. Revisão da Literatura.....	5
2.1 Treinamento Profissional.....	5
2.1.1 Treinamento e Desenvolvimento de Pessoas.....	5
2.1.2 Diagnóstico de Necessidade de Treinamento (DNT).....	6
2.1.2.1 Níveis de Análise do DNT.....	6
2.1.2.2 Métodos e Técnicas para o DNT.....	7
2.1.3 Formação Contínua.....	8
2.1.4 Treinamento como Ferramenta Estratégica e de Desenvolvimento de Competências.....	9
2.1.4.1 Como o Treinamento Ajuda a Construir Competências.....	9
2.1.4.2 O Impacto do Treinamento.....	10
2.1.4.3 Treinamento como Investimento Estratégico.....	11
2.2 Tecnologias no Treinamento Profissional.....	11
2.2.1 Serious Games.....	11
2.2.2 Realidade Virtual (RV).....	12
2.2.3 Inteligência Artificial (IA).....	13
2.2.3.1 Aplicações da IA na Capacitação Profissional.....	13
2.2.3.2 A Evolução dos NPCs com IA.....	14
2.2.3.3 IA Generativa e a Conversação Dinâmica.....	14
2.2.3.4 Engenharia de Prompts.....	15
2.3 Sinergia Tecnológica Aplicada ao Treinamento de Atendimento.....	15
3. Trabalho Desenvolvido.....	17
3.1 Arquitetura da Solução.....	17
3.2 Escolha do modelos de IA.....	19
3.3 Desenvolvimento da API.....	20
3.4 Desenvolvimento do Serious Game.....	21
3.4.1 Planejamento e estrutura base.....	21
3.4.2 Definição do fluxo principal do jogo.....	22
3.4.3 Desenvolvimento do protótipo.....	23
3.4.3.1 Ambiente do Armazém e Gestão de Pacotes.....	23
3.4.3.2 Ferramentas e Objetos Interativos.....	25
3.4.3.3 Configuração do NPC.....	27

3.4.3.4 Gerenciadores e Lógica de Jogo.....	29
3.5 Melhorias e Refinamentos.....	30
3.5.1 Refinamento da Interação por Voz.....	30
3.5.2 Otimização da Latência e Modularização da API.....	32
3.5.3 Sistema de Feedback.....	34
4. Resultados Obtidos.....	36
4.1 Ambiente de Testes e Hardware Utilizado.....	36
4.2 Protótipo e a Experiência Imersiva.....	36
4.3 Conversação Dinâmica com IA.....	37
4.3.1 Cenário: Cliente Colaborativo.....	37
4.3.2 Cenário: Cliente Inseguro.....	39
4.3.3 Cenário: Cliente Distraído.....	41
4.4 Potencial do Sistema de Feedback.....	43
4.5 Otimização da Latência.....	43
5. Conclusão.....	44
6. Referências Bibliográficas.....	45
Anexos.....	49

1. Introdução

1.1 Contextualização

O treinamento profissional é um processo sistemático e contínuo que busca desenvolver conhecimentos, habilidades e atitudes para melhorar o desempenho individual e organizacional, especialmente no atual cenário de mercado dinâmico e altamente competitivo, onde a intensificação da concorrência e as rápidas mudanças tecnológicas exigem funcionários qualificados e bem preparados. Nessa perspectiva, o treinamento se consolida como uma ferramenta estratégica para o desenvolvimento organizacional projetado, essencial para preparar os funcionários para agirem diante de diversas situações, agregando valor às pessoas e enriquecendo o patrimônio da empresa ao aprimorar processos internos, qualidade e produtividade, diferenciando-se do desenvolvimento de pessoas por focar na construção imediata de habilidades e competências voltadas ao cargo atual do funcionário. (Esteves, 2009; Rodrigues, 2019)

Essa necessidade de avaliar desempenho e desenvolver treinamento nas organizações existiam desde o início do século XX, mas ganhou maior relevância principalmente após a Segunda Guerra Mundial com a transição para a era do conhecimento onde o conceito de treinamento se expandiu para o de "formação", focando na competência individual. (Rocha, 2015)

1.2 Limitações das abordagens tradicionais

Várias abordagens e ferramentas foram utilizadas para conseguir uma maior eficiência na capacitação, sempre visando aumentar o engajamento e retenção de informações por parte dos funcionários. Inicialmente, o treinamento profissional era frequentemente focado na transmissão de conhecimento teórico por meio de documentação em papel e métodos passivos, como aulas expositivas com uso de slides. Já a prática, era realizada diretamente no ambiente de trabalho real com outros colegas de trabalho mais experientes (Stoque et al., 2002). Porém, essas abordagens apresentavam limitações. O treinamento prático podia expor aprendizes a perigos em atividades de risco, além de que o conhecimento transmitido por colegas mais experientes pode gerar vícios técnicos e resistência a novas técnicas

mais eficazes. Os treinamentos internos eram considerados excessivamente teóricos, falhando em adaptar a linguagem ou promover engajamento, e com pouca efetividade na aplicação do conhecimento no trabalho. A dissociação entre a exposição teórica e a aplicação prática também limitava a consolidação do aprendizado. A avaliação, quando utilizada, frequentemente se limitava a exercícios ou questionários.(Lima, 2020; Mourão et al., 2014)

Diante de todas essas limitações dos métodos mais tradicionais, ficou mais evidente a demanda por abordagens que facilitem o processo de ensino-aprendizagem. Essas pendências deixadas pelo treinamento tradicional abriram margem para vários estudos de metodologias e tecnologias que promovem uma aprendizagem mais ativa, imersiva e interativa. Neste contexto, os *Serious Games*, Realidade Virtual (RV) e a Inteligência Artificial (IA) se destacam por serem tecnologias capazes de transformar a maneira como o treinamento profissional é conduzido, ao proporcionar ambientes seguros, imersivos e interativos que simulam cenários reais do trabalho, promovendo maior engajamento e retenção do conhecimento, além de permitirem abordagens personalizadas e adaptativas para os aprendizes.(A. P. D. Silva et al., 2024)

1.3 Objetivo

Desta forma, este trabalho visa propor e desenvolver um protótipo de um *Serious Game* em Realidade Virtual, com aplicação de Inteligência Artificial para geração de diálogos. O objetivo deste protótipo é oferecer uma experiência de treinamento profissional voltado para atendimento ao público, de forma que o usuário pratique habilidades de comunicação através de diálogos com clientes virtuais (NPCs) que são geridos em tempo real por um sistema de IA, e pratique procedimentos necessários na posição abordada no treinamento com uma maior imersão através da realidade virtual. Para alcançar este objetivo principal, foram definidos os seguintes sub-objetivos específicos:

- Desenvolver uma API robusta em Python capaz de orquestrar múltiplos modelos de Inteligência Artificial para processamento de voz e geração de diálogo.

- Integrar modelos de IA para realizar as funções de reconhecimento de voz (STT), geração de diálogo contextualizado (LLM) e síntese de voz (TTS).
- Construir um ambiente 3D imersivo em Realidade Virtual, que simula um cenário de atendimento real, como um balcão de atendimento, com objetos e ferramentas interativas.
- Implementar um sistema de diálogo dinâmico, permitindo que o NPC (cliente virtual) responda de forma coerente e execute ações no ambiente do jogo com base nas falas do jogador e no contexto da simulação.

1.4 Metodologia de Trabalho

A metodologia adotada para desenvolvimento deste trabalho foi baseada em um desenvolvimento com uma abordagem iterativa, utilizando a metodologia *Kanban* para a gestão visual do fluxo de desenvolvimento, focando na criação de um protótipo funcional que relaciona o conhecimento teórico com a prática. No primeiro momento foram feitas pesquisas de artigos com temas relacionados à proposta do projeto para solidificar o enquadramento teórico. Depois, o desenvolvimento da parte prática foi dividida em duas partes: Uma API para rodar os modelos de inteligência artificial e a aplicação em realidade virtual que faz o uso da API. O processo de desenvolvimento foi marcado por etapas de testes e refinamentos, que permitiram a identificação e solução de desafios técnicos, como a otimização da latência de resposta da IA e a modularização da API para maior eficiência e escalabilidade.

1.5 Plano de Trabalhos

O desenvolvimento do projeto seguiu uma estrutura de quatro etapas principais:

- Etapa 1 (Fundamentação e Requisitos): Levantamento e estudo do cenário atual das áreas de treinamento profissional, *Serious Games*, Realidade Virtual e Inteligência Artificial, resultando na redação da Revisão da Literatura. Nesta etapa, também foi realizado o levantamento de requisitos da aplicação, definindo a arquitetura base do projeto.
- Etapa 2 (Criação da API): Desenvolvimento da API que orquestra o fluxo de atendimento entre os modelos de IA. Nessa etapa foi criado um servidor básico em Python, utilizando a biblioteca *Flask* para implementar os *endpoints*

e a integração local dos modelos Whisper para transcrição da fala, Llama 3.1 para geração de diálogos e o XTTS para síntese de voz.

- Etapa 3 (Criação da aplicação RV): Desenvolvimento do jogo em realidade virtual que consome a api para simular um atendimento. Esta etapa envolveu a escolha do ambiente que seria representado, criação do cenário 3D de um ponto dos correios, implementação das mecânicas de interação do jogador com objetos e a programação do comportamento do NPC, incluindo a sua integração com a API de IA.
- Etapa 4 (Testes, correções e melhorias): Ciclo de testes após implementação para validar o fluxo de interação completo, seguido de correções e melhorias observadas nos testes. Nessa etapa testes ponta a ponta foram feitos, o que possibilitou otimizações essenciais no protótipo, como uma refatoração na API que possibilitou mesclar modelos de IAs locais e em nuvem para reduzir a latência e melhorar a experiência do jogador.

2. Revisão da Literatura

2.1 Treinamento Profissional

2.1.1 Treinamento e Desenvolvimento de Pessoas

O treinamento profissional é visto como um processo organizado de forma sistemática, onde o principal objetivo é oferecer aos funcionários os meios para aprender e aprimorar habilidades que tenham uma relação direta com a atividade que o funcionário exerce ou irá exercer. Este processo frequentemente é visto como um ensino de curto a médio prazo, já que o seu principal objetivo é fazer com que todos na empresa trabalhem de forma mais eficiente e padronizada através de objetivos bem definidos.(Stoque et al., 2002)

Já o desenvolvimento de pessoas é um conceito mais geral e voltado para o longo prazo. Seu principal objetivo é a capacitação contínua do funcionário, que diferente do treinamento profissional, foca também em desenvolver habilidades relevantes e exigidas para futuras funções que ele possa exercer em novos cargos. O desenvolvimento busca o aprendizado de habilidades de conhecimento mais geral e interpessoais, melhorando não só habilidades técnicas como também mudanças de atitude mais profundas e até um jeito mais abstrato de pensar e raciocinar. Quando a empresa se preocupa com o desenvolvimento dos funcionários, isso pode melhorar bastante o desempenho deles, mostrando que ela se importa com algo além do simples cumprimento das tarefas.(Stoque et al., 2002)

Tanto o treinamento quanto o desenvolvimento se baseiam no conceito de competência, envolvendo a integração de conhecimento, habilidades e atitudes (CHAs) necessários para a execução eficiente no trabalho. A principal diferença entre os dois é que o treinamento busca melhorar habilidades voltadas ao cargo atual para uma aplicação imediata, enquanto o desenvolvimento busca preparar o indivíduo para desafios futuros e promovendo seu crescimento contínuo, sendo um processo de longo prazo que abrange aspectos técnicos, conceituais e comportamentais. (Stoque et al., 2002)

2.1.2 Diagnóstico de Necessidade de Treinamento (DNT)

Para o planejamento e desenho das ações que serão trabalhadas na capacitação é essencial determinar “quem” precisa ser treinado, “o que” precisa ser aprendido e “porquê” o treinamento é necessário. Essa análise é conhecida como Diagnóstico de Necessidades de Treinamento (DNT) e constitui o ponto de partida essencial e a base para a concepção e implementação de programas de treinamento eficazes nas organizações. O papel dela é importante porque ajuda a direcionar os treinamentos para aquilo que a empresa realmente precisa, garantindo que o tempo e o dinheiro investidos tragam resultados práticos e ajudem no crescimento das pessoas que trabalham lá. (Mourão et al., 2014)

2.1.2.1 Níveis de Análise do DNT

Para entender as reais necessidades da empresa, o diagnóstico costuma ser realizado em três níveis de análise distintos, mas complementares: Organizacional, de Tarefas/Operações e Pessoal/Individual.

- **Análise Organizacional:** Examina a empresa como um todo, levando em consideração os objetivos de curto e longo prazo, quais são seus recursos, como está o ambiente ao redor (tanto social quanto tecnológico) e como é o clima interno. Sua contribuição principal é garantir que os treinamentos estejam alinhados com o que a empresa quer para o futuro e com o que acontece fora dela, auxiliando a definir o “porquê” dos treinamentos. O sucesso de um programa de treinamento está diretamente ligado à correta identificação das necessidades neste nível.
- **Análise das Tarefas/Operações:** Examina as atividades e responsabilidades de cada função para descobrir quais conhecimentos, habilidades e atitudes (CHAs) são realmente importantes para cada papel. Esta análise define com precisão os requisitos de competências demandados por uma função e ajuda a determinar "no que" treinar os funcionários.
- **Análise Pessoal/Individual:** Voltada diretamente para os funcionários da empresa. O objetivo dela é identificar quais indivíduos ou equipes necessitam desenvolver determinadas competências, analisando o que cada um já sabe

e onde estão as principais lacunas. Ferramentas como avaliação de desempenho ajudam muito nesse processo, mostrando onde estão as maiores necessidades. Esta análise responde à pergunta "quem" deve ser treinado e considera também a percepção dos próprios participantes sobre a necessidade do treinamento, fator que influencia o impacto do mesmo no trabalho.

Quando juntamos essas três análises conseguimos montar um diagnóstico certo e criar programas de treinamento que realmente fazem a diferença tanto para a empresa quanto para os funcionários. (Mourão et al., 2014)

2.1.2.2 Métodos e Técnicas para o DNT

Para realizar o DNT e definir quem precisa de treinamento e o que deve ser aprendido, existem várias formas e métodos que podem ser usados, muitas vezes juntos. O melhor caminho depende do contexto da empresa e dos objetivos do diagnóstico. (Stoque et al., 2002)

Entre as principais técnicas e métodos reconhecidos, estão:

- **Observação Direta:** O analista acompanha o trabalho acontecendo na prática, o que ajuda a perceber dificuldades, desvios de procedimentos ou necessidades não explicitadas pelos funcionários. (Stoque et al., 2002)
- **Entrevistas:** Podem ser realizadas individualmente ou em grupo, com funcionários, gestores e até mesmo com clientes. O objetivo é recolher informações detalhadas sobre o que precisa ser aprendido, percepções e sugestões. Entrevistas semiestruturadas, por exemplo, são úteis para o diagnóstico do público-alvo e levantamento de competências. (Lima, 2020; Yezzi & Barros Neto, 2025)
- **Questionários:** Forma prática de coleta massiva e padronizada de dados, investigando opiniões, percepções de necessidade, ou o domínio de certas competências. (Yezzi & Barros Neto, 2025)
- **Pesquisas Internas e com o Público-Alvo:** Recomendadas para um diagnóstico mais certo, já que permitem levantar diretamente com quem será treinado quais são as principais necessidades. (Ferreira et al., 2024)

- **Análise Documental e de Indicadores:** É levantado e realizado o estudo de documentos da empresa, como planejamento estratégico, descrições de cargos, avaliações de desempenho anteriores, e o levantamento de dados por meio de indicadores de resultados da empresa que podem sinalizar necessidades de melhoria ou despreparo profissional.(Stoque et al., 2002; Yezzi & Barros Neto, 2025)
- **Discussões em Grupo e Estudo de Casos:** Facilitam a troca de informações e a identificação de problemas comuns. Além disso, são ótimas para fixar conhecimento e ensinar como resolver situações reais.(Stoque et al., 2002)
- **Simulações e Role-Playing:** Úteis para treinar habilidades de relacionamento e atendimento, porque permitem vivenciar situações do dia a dia. Embora sejam também estratégias de treinamento, podem ser usadas no diagnóstico para avaliar o desempenho atual em cenários específicos.(A. P. D. Silva et al., 2024)

O uso destas técnicas de forma integrada, considerando diferentes níveis de análise (organizacional, tarefas e individual), resulta em um diagnóstico mais completo e preciso, embasando a elaboração de programas de treinamento que realmente ajudem a empresa a alcançar seus objetivos e promovam o desenvolvimento dos funcionários. (Ferreira et al., 2024; Stoque et al., 2002)

2.1.3 Formação Contínua

O mercado de trabalho atual é marcado por rápidas mudanças e pela constante evolução do conhecimento, impondo a necessidade de um conceito conhecido como aprendizagem ao longo da vida. Esse conceito se refere ao processo contínuo de adquirir novos conhecimentos e aprimorar habilidades, acompanhando toda a trajetória profissional.(Barilli & Cunha, 2003; Dutra & Eboli, 2022)

A formação continuada traz diversos benefícios tanto para o funcionário quanto para a empresa. Para a empresa, funcionários que estão sempre aprendendo trazem mais inovação, melhoram a qualidade do que é entregue, aumentam a produtividade, fortalecem a cultura da empresa e ajudam a reter talentos. Já para o profissional, há um desenvolvimento de novas habilidades, mais confiança, mais

chances de crescer na carreira e mais facilidade para lidar com mudanças.(Sá et al., 2022; Yezzi & Barros Neto, 2025)

Por isso, deve ser de interesse e responsabilidade de ambas as partes o desenvolvimento contínuo. A empresa tem o papel de facilitar e incentivar esse processo, através de investimentos em treinamentos, cursos e outras formas de desenvolvimento que estejam de acordo com o que a empresa precisa e que sejam acessíveis para todos, resultando em um ambiente onde aprender faz parte do dia a dia. Em contrapartida, o funcionário deve estar disposto a receber e procurar novos conhecimentos, desenvolver as habilidades que precisa para crescer na carreira e cuidar do próprio desenvolvimento.(Barilli & Cunha, 2003; Sá et al., 2022)

2.1.4 Treinamento como Ferramenta Estratégica e de Desenvolvimento de Competências

O treinamento na prática é uma ferramenta para desenvolver habilidades que a empresa precisa e deve ser visto como um investimento estratégico, não só como um custo a mais para a empresa.

2.1.4.1 Como o Treinamento Ajuda a Construir Competências

Como mencionado antes (2.1.1), a competência envolve Conhecimentos, Habilidades e Atitudes (CHA). O treinamento profissional tem um papel direto em desenvolver cada uma dessas áreas:

- **Conhecimentos (o Saber):** O treinamento é uma ferramenta útil para transmissão de informações e conhecimentos aplicados no trabalho do dia a dia, contextualizando sua aplicação prática e relevância para as tarefas do funcionário e para os objetivos da empresa.
- **Habilidades (o Saber Fazer):** Neste ponto, o foco do treinamento é converter o que foi aprendido em ação. Ou seja, desenvolver a capacidade de executar tarefas e aplicar técnicas, seja no manuseio de uma ferramenta, na condução de uma negociação ou na interação com um cliente. As habilidades se desenvolvem principalmente com prática, simulações e *feedbacks*, ajudando o funcionário a ganhar confiança e agilidade.

- **Atitudes (o Querer Fazer/Saber Ser):** Este ponto tende a ser o mais desafiador, já que lida com fatores internos do funcionário, como sua atitude, conduta e postura profissional. O papel do treinamento aqui é provocar reflexão, mostrar na prática a importância de atitudes como proatividade, empatia no atendimento e trabalho em equipe, além de reforçar um ambiente que valorize esse tipo de comportamento. O objetivo é alinhar o jeito de agir das pessoas com a cultura e os valores da empresa, o que faz toda diferença no clima e no trabalho em equipe.

Portanto, um treinamento bem planejado não só passa informação, mas ajuda a desenvolver o CHA de forma equilibrada, preparando o time para encarar os desafios do dia a dia com muito mais preparo e confiança. (Lima, 2020; Stoque et al., 2002)

2.1.4.2 O Impacto do Treinamento

O treinamento eficaz, ao desenvolver os Conhecimentos, Habilidades e Atitudes (CHA), gera benefícios significativos tanto para as organizações quanto para os funcionários envolvidos. (Stoque et al., 2002)

Do ponto de vista organizacional, os benefícios podem ser divididos em dois grupos, os tangíveis e os não tangíveis. Os tangíveis são frequentemente os mais perceptíveis, a qualidade dos serviços e produtos melhora e os erros, retrabalho e desperdícios diminuem. Um treinamento bem direcionado, como o de atendimento ao cliente, por exemplo, faz com que os clientes fiquem mais satisfeitos e fiéis, o que pode até aumentar o lucro. Já os intangíveis impactam no clima organizacional, o fortalecimento dos relacionamentos interpessoais e da confiança entre as equipes, fatores fundamentais para a construção de uma cultura organizacional sólida e para a retenção de talentos. (Lacerda & Casagrande, 2022)

Do ponto de vista do funcionário, não apenas novas competências são adquiridas, mas também é notado melhorias imediatas em sua eficiência e agilidade nas tarefas cotidianas e na resolução de problemas, contribuindo com o aumento da satisfação com o próprio trabalho. O treinamento também ajuda a aumentar a autoconfiança e a motivação, além de dar uma visão mais clara das regras e valores da empresa. (Sá et al., 2022)

2.1.4.3 Treinamento como Investimento Estratégico

Com todos os benefícios apresentados que o treinamento proporciona, fica mais evidente que a capacitação de um time é um investimento estratégico e não um custo. (Stoque et al., 2002)

Olhar para o treinamento como estratégia é entender que as pessoas são o maior patrimônio da empresa. Em um mercado competitivo e em constante mudança, as empresas que investem no desenvolvimento contínuo de seus funcionários estão mais preparadas para se adaptar, inovar e enfrentar novos desafios. Esse tipo de investimento cria uma vantagem real: equipes mais preparadas, motivadas e alinhadas com os objetivos da empresa, o que impacta diretamente na satisfação dos clientes e no sucesso do negócio a longo prazo. (Sá et al., 2022)

Fica claro então que o treinamento profissional é um dos pilares para o sucesso das empresas, sendo um investimento fundamental no desenvolvimento das competências (CHA) que realmente fazem diferença nos resultados. Contudo, conforme apontado na introdução deste trabalho, a eficácia deste processo é frequentemente limitada pelos métodos tradicionais, que lutam para promover o engajamento e a aplicação efetiva do conhecimento. Por isso, buscar formas mais interativas e imersivas de aprender virou uma necessidade. Nesse cenário, os *Serious Games* aparecem como uma solução tecnológica inovadora, com potencial para transformar a experiência de aprendizagem e atender essas necessidades.

2.2 Tecnologias no Treinamento Profissional

O uso de novas tecnologias é essencial para superar as limitações dos métodos mais tradicionais de treinamento, tornando o aprendizado mais imersivo, interativo e eficiente. Limitações como baixo engajamento, falta de realismo nas simulações, dificuldade em treinar habilidades interpessoais e ausência de personalização são exemplos de problemas que tendem a serem contornadas por ferramentas baseadas em *Serious Games*, Realidade Virtual (RV) e Inteligência Artificial (IA). (A. P. D. Silva et al., 2024)

2.2.1 *Serious Games*

Serious Games pode ser definido como um jogo desenvolvido com objetivos que vão além de unicamente entretenimento, podendo ser ensinar, simular situações do mundo real ou treinar habilidades. Eles usam as mesmas tecnologias dos jogos tradicionais, se diferenciando principalmente na sua finalidade e na sua forma de medir sucesso, que ao invés de ser determinado pela popularidade e diversão, os *Serious Games* são avaliados pela sua eficácia em atingir os objetivos de aprendizagem e promover o desenvolvimento de competências.(Buoro et al., 2021)

Quando se trata de treinamento profissional, o *Serious Games* se destaca como uma ferramenta que facilita tanto o ensino de habilidades técnicas quanto não-técnicas. Por simular ambiente e situações reais, o funcionário consegue se sentir imersivo e entender de forma mais clara como são executados os processos na empresa, de forma que no dia a dia ele consiga entender melhor como deve atuar perante certos problemas que podem aparecer. Entre os benefícios desta ferramenta estão a possibilidade de testar estratégias em ambientes seguros, receber *feedbacks* instantâneos, aumentar a motivação dos participantes e permitir uma mensuração objetiva da performance dos participantes.(A. P. D. Silva et al., 2024)

2.2.2 Realidade Virtual (RV)

A Realidade Virtual refere-se a um ambiente 3D gerado por computador que busca criar uma experiência altamente interativa e imersiva que o usuário tem a percepção de estar presente em um mundo diferente do físico. Para tal feito, é necessário apresentar estímulos sensoriais como visão, audição e tato de forma convincente e que respondam às suas ações e movimentos em tempo real. Essa interação imediata é crucial para sustentar a ilusão de presença, fazendo com que o mundo simulado pareça tão real quanto o ambiente físico.(Almeida, 2019)

A Realidade Virtual se baseia em três pilares:

- **Imersão:** Capacidade de envolver os sentidos do usuário, normalmente usando óculos especiais (*Head-Mounted Display* ou HMDs), para dar uma visão completa do ambiente virtual.(Almeida, 2019)

- **Interação:** O sistema responde na hora ao que o usuário faz, muitas vezes com dispositivos que simulam toque e força, permitindo manipular objetos virtuais.(Almeida, 2019)
- **Envolvimento:** Nível de motivação e engajamento do usuário com as atividades dentro do ambiente virtual.(Almeida, 2019)

Um dos grandes diferenciais da Realidade Virtual, especialmente no treinamento profissional, é permitir a prática realista de tarefas complexas ou perigosas em um ambiente seguro, sem riscos reais e sem o consumo de materiais. A natureza simulada do ambiente virtual possibilita repetir os exercícios quantas vezes for necessário, o que é fundamental para a fixação do conhecimento e facilita a aplicação prática no futuro.(A. P. D. Silva et al., 2024).

2.2.3 Inteligência Artificial (IA)

Inteligência Artificial (IA) pode ser definida como o campo de estudo na área da ciência da computação que busca desenvolver sistemas capazes de simular a inteligência humana e executar tarefas que normalmente exigiriam intervenção humana. (P. H. R. Silva, 2019)

Esse campo teve sua fundação em 1956 e passou por várias fases de estagnação, já que barreiras como a necessidade de um investimento pesado, poder computacional limitado, baixa disponibilidade de dados para treino e algoritmos pouco sofisticados acabavam impossibilitando o avanço da área. Porém, nos últimos anos, muitas dessas barreiras foram superadas. Graças a um mundo mais conectado resultando em uma vasta disponibilidade de dados gerados, um avanço exponencial do poder computacional, algoritmos mais complexos e eficientes e um alto investimento permitiu que a IA se tornasse uma tecnologia transformadora em inúmeros setores. (Souza et al., 2024)

2.2.3.1 Aplicações da IA na Capacitação Profissional

No âmbito da capacitação, a IA é aplicada como uma ferramenta para otimizar o processo de aprendizagem e desenvolver competências, atuando principalmente em duas frentes: a personalização da aprendizagem e a criação de comportamentos inteligentes para personagens não-jogáveis (NPCs) em simulações.

Cada pessoa tem seu próprio ritmo e necessidade de aprendizado, e a IA tem potencial de prover uma personalização que melhor se adequa ao indivíduo (Mendanha et al., 2025). O Sistema de Tutoria Inteligente (*Intelligent Tutoring System* - ITS) é um exemplo dessa abordagem, que faz uso de inteligência computacional para simular o acompanhamento individual de um tutor humano. Esses sistemas analisam o desempenho do usuário, identificam onde ele tem mais dificuldade e ajustam o conteúdo e o nível das tarefas conforme necessário. Isso só é possível porque o ITS utiliza um contexto geral de informações, como dados sobre o assunto, o nível de conhecimento do usuário e estratégias pedagógicas que melhor se adequa às suas necessidades. (Barbato, 2016)

2.2.3.2 A Evolução dos NPCs com IA

Já nas simulações, a IA é a ferramenta que traz naturalidade aos personagens não-jogáveis (NPCs), criando comportamentos que respondem e se adaptam ao que acontece. Tradicionalmente, esses personagens possuem suas ações e diálogos pré-definidos e gerenciados por uma Máquinas de Estados Finitos (*Finite State Machines*), que apenas define qual ação ou fala será executada perante tal cenário, tornando os comportamentos muitas vezes previsíveis. Com o avanço da IA, surgiram novas abordagens para ter um comportamento do NPC mais natural, como o uso do Aprendizado de Máquina (Machine Learning), que permite criar NPCs com comportamentos complexos e menos previsíveis. Outro exemplo de abordagem é o Aprendizado por Reforço (*Reinforcement Learning*), na qual o NPC é treinado a tomar decisões com base em recompensas e punições, fazendo com que ele aprenda sozinho a atingir objetivos, de forma mais parecida com o comportamento humano. (Rolim, 2023)

2.2.3.3 IA Generativa e a Conversação Dinâmica

O avanço mais disruptivo para o propósito deste trabalho é a IA Generativa, especialmente os Grandes Modelos de Linguagem (LLMs), que são capazes de reconhecer, resumir e gerar linguagem humana com base no conjunto de dados em que ele foi treinado. Para esse tipo de IA, geralmente a arquitetura se baseia em *Transformers*, que utiliza mecanismos de “atenção” para focar seletivamente em diferentes partes do texto de entrada, o que lhes confere uma capacidade sofisticada de lidar com o contexto. Este treinamento extensivo permite que os modelos

compreendam padrões complexos e relações entre palavras. Com esta capacidade, a IA consegue superar a limitação dos diálogos pré-definidos, onde ao invés de seguir um roteiro fixo, ela entende o que o usuário diz e gera respostas dinâmicas e relevantes na hora. Em treinos de atendimento ao público, isso se torna extremamente valioso, pois permite simular a diferentes personalidades (clientes irritados, confusos, etc.) e situações inesperadas encontradas no mundo real, tornando a prática de comunicação, empatia e resolução de problemas de forma muito mais realista. (Rolim, 2023)

Porém, o modelo LLM por si só não tem capacidade de se comunicar por voz, sendo necessário se aliar com duas outras tecnologias cruciais. A primeira é o Reconhecimento de Fala (*Speech-to-Text* ou STT), que consiste no processo de converter o sinal de fala do usuário numa sequência de palavras (F. R. D. da Silva, 2022). A segunda é a Síntese de Fala (*Text-to-Speech* ou TTS), a tecnologia que converte texto escrito em áudio falado que simula a voz humana (Souza Figueiredo Junior et al., 2024). Ao conciliar a funcionalidade das três ferramentas, o LLM passa a ter capacidade de entender a fala do usuário e responder por voz, proporcionando uma conversa mais natural e imersiva ao aplicar dentro de uma simulação.

2.2.3.4 Engenharia de *Prompts*

Para garantir que os diálogos gerados pela IA realmente ajudem no aprendizado, é fundamental usar a Engenharia de *Prompts*. Essa técnica consiste em criar instruções detalhadas para orientar a IA e garantir que as respostas sejam coerentes e úteis. Um bom *prompt* descreve claramente a tarefa, o perfil e o comportamento do NPC, além de explicar o contexto da simulação, ajudando a manter as interações alinhadas com os objetivos do treinamento. (Rolim, 2023)

2.3 Sinergia Tecnológica Aplicada ao Treinamento de Atendimento

As três tecnologias abordadas apresentam seus benefícios ao serem aplicadas em treinamentos. Porém, a combinação delas representa um avanço significativo na eficácia do treinamento, especialmente para competências interpessoais, ao oferecer uma experiência estruturada, imersiva, realista e adaptável ao mesmo tempo.

Nessa combinação, cada tecnologia desempenha funções específicas e complementares. O *Serious Game* fornece a estrutura pedagógica: os objetivos de aprendizagem, as regras e os desafios (Buoro et al., 2021). A Realidade Virtual cria o ambiente tridimensional, que aumenta a sensação de presença e permite praticar em cenários realistas (Almeida, 2019). Por fim, a IA atribui vida aos NPCs, permitindo diálogos dinâmicos e imprevisíveis, semelhantes às interações humanas reais (Rolim, 2023). Desta forma, o participante não apenas recebe informações teóricas sobre o atendimento, mas vivencia diretamente o atendimento em contexto prático.

Entretanto, a implementação desta integração de tecnologias apresenta um desafio técnico a ser superado: a latência. A Latência é o atraso entre uma ação e a resposta a essa ação. Como a RV depende de interações em tempo real para sustentar a imersão (Almeida, 2019), qualquer atraso significativo na resposta da IA quebra a naturalidade da conversa e a sensação de presença do usuário. Esse atraso pode ser causado tanto pela complexidade do modelo de IA (latência computacional) quanto pelo tempo de comunicação com servidores externos (latência de rede). Por isso, controlar a latência é fundamental para garantir que o treinamento seja fluido e eficiente. (Rolim, 2023)

Para lidar com essa latência, são implementadas estratégias técnicas e de design. Do lado técnico, busca-se reduzir o atraso usando modelos de IA mais leves e rápidos ou entregando as respostas de forma incremental (*streaming*), sem esperar a frase completa. (Rolim, 2023) Do lado de design, a estratégia é disfarçar a percepção desses atrasos utilizando gestos ou animações sugestivas de pensamento, e frases breves de preenchimento, como exemplo 'Hmm, deixe-me ver...', mantendo assim a fluidez e a naturalidade da interação, um conceito que se alinha com as estratégias de '*misdirection*' e '*benevolent deception*' para gerenciar a experiência do usuário durante períodos de latência. (Antognoli & Fisher, 2023)

3. Trabalho Desenvolvido

Baseado no conteúdo do tópico anterior, foi desenvolvido o protótipo do *Serious Game* “Atende.AI”. O desenvolvimento foi feito em etapas, começando com a criação de um protótipo funcional que englobava o fluxo principal do atendimento. A partir dessa base, foram realizadas melhorias incrementais, adicionando novas funcionalidades e corrigindo débitos técnicos identificados ao longo dos ciclos de desenvolvimento e teste.

A ideia desse projeto partiu de uma perspectiva futura de um projeto de iniciação tecnológica chamado “*VocationalLab: Uma Experiência Imersiva com Realidade Virtual para Apoio à Orientação Vocacional*”(L. Silva & Natale, 2024). Neste projeto foi trabalhado o uso de *Serious Game* e Realidade Virtual para apoio ao processo vocacional, na qual durante seu desenvolvimento foi levantado a ideia de incluir uso de IA para conversa com *NPC*, porém, acabou não sendo aplicado por conta da tecnologia na época não estar madura o suficiente como está hoje. Agora, com o avanço da tecnologia, está sendo cada vez mais acessível e satisfatório os resultados.

3.1 Arquitetura da Solução

Inicialmente, a configuração do servidor de IA foi focada em rodar os modelos localmente para evitar custos com APIs durante o desenvolvimento. Além disso, o fator latência também foi considerado nessa escolha, já que havia grande chances do processamento e o consumo em rede local de um modelo mais simples, ser mais rápido do que depender de uma API externa.

Motores gráficos como a Unity até possibilitam configurar modelos de IA diretamente na aplicação com a ferramenta *sentis*. Porém, utilizar isso demandaria de muito poder de processamento gráfico e memória de vídeo disponível para manter o jogo em uma taxa de quadro constante ao mesmo tempo que processa os modelos..Por isso, foi adotado a arquitetura cliente-servidor: *Serious Game* (cliente) faz pedidos HTTP para uma API de IA (servidor). Esta separação de responsabilidades é ilustrada na arquitetura geral da solução, apresentada na Figura 1.

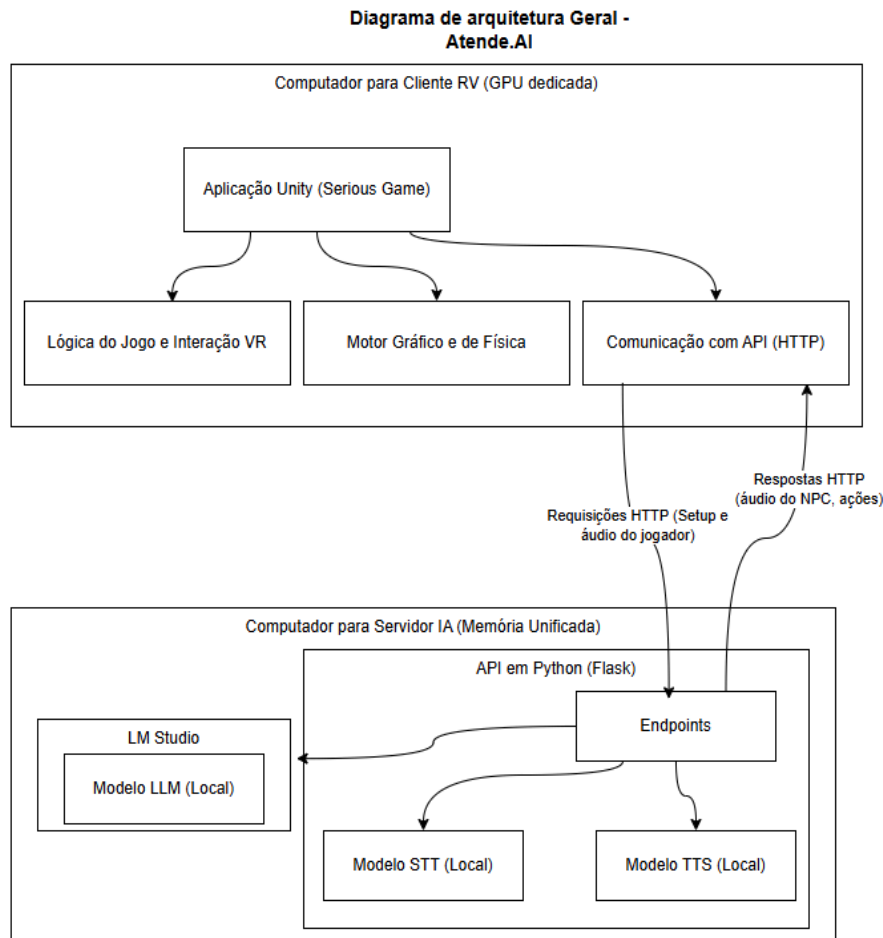


Figura 1: Diagrama de Arquitetura Geral da Solução Atende.AI.

Conforme o diagrama, a arquitetura é distribuída em dois componentes principais, cada um executado em um hardware com características distintas para otimizar o desempenho. Para o componente do cliente (o jogo em RV), foi utilizado um computador equipado com uma GPU dedicada potente o suficiente para renderizar a aplicação em realidade virtual com fluidez. Já para o componente do servidor (a API de IA), foi utilizado um segundo computador com uma arquitetura de memória unificada, ideal para alocar a grande quantidade de memória necessária para executar múltiplos modelos de IA simultaneamente de forma eficiente.

O fluxo de conversa entre o jogador e o NPC exige três tipos distintos de modelos de IA: um modelo para transcrever o áudio da fala do jogador em texto (*Speech-To-Text* ou STT), um modelo de linguagem para interpretar esse texto e, com base nele e nas instruções configuradas, gerar uma resposta textual coerente (Large Language Model ou LLM), e finalmente um modelo para transformar o texto gerado pelo LLM em áudio da fala do NPC (*Text-To-Speech* ou TTS).

3.2 Escolha do modelos de IA

Para as tarefas de conversão de voz em texto (STT) e de texto em voz (TTS), foram escolhidos, respectivamente, os modelos Whisper (small) e XTTS-V2, ambos integrados em uma API desenvolvida em Python com o framework *Flask*. O modelo do LLM escolhido foi o LLAMA, na qual foi executado por meio do *LM Studio*, utilizando o formato MLX.

A escolha dos modelos Whisper-small e XTTS se deu devido a popularidade, facilidade de implementação e o resultado satisfatório em testes realizados. Porém, mesmo que menos exigentes em questão de consumo de memória de vídeo comparado a outros modelos mais sofisticados, ainda consomem uma quantidade considerável da máquina utilizada, memória essa que também precisa ser compartilhada com outras tarefas do sistema operacional. Por isso, a escolha do LLM foi feita com base em critérios específicos: compatibilidade com o formato MLX para uso da GPU, disponibilidade no *LM Studio* e consumo máximo de memória de até 7 GB.

Seguindo esses critérios, quatro opções foram inicialmente separadas para avaliação:

- LLama 3.1 8B instruct 4bit (4,53 GB)
- Mistral Nemo Instruct 2407 4bit (6,91 GB)
- Phi 3 mini 4k instruct 4bit (2,15 GB)
- LLama 3.2 3B Instruct (1,82 GB)

A avaliação foi realizada por meio de testes interativos, utilizando *prompts* genéricos, porém alinhados ao contexto da aplicação, como comandos relacionados à entrega de documentos e manipulação de objetos.

Durante os testes, os modelos LLama 3.1 e Mistral apresentaram desempenho satisfatório, respondendo adequadamente às instruções. O modelo Phi 3 mini, por outro lado, gerou respostas muito curtas e ignorou várias vezes comandos importantes, sendo descartado. O LLama 3.2, apesar de inicialmente promissor, apresentando bons resultados no início do teste, se comportou de forma inconsistente, onde várias vezes, mesmo utilizando as mesmas instruções e

configurações que os outros modelos, começava a gerar como resposta a fala do npc e a do jogador ao mesmo tempo. Por isso foi desqualificado.

Ao final das avaliações, o LLama 3.1 foi escolhido em vez do Mistral, pois ambos apresentaram desempenho semelhante, mas o LLama 3.1 se destacou pelo menor consumo de memória, atendendo de forma eficiente e prática às restrições e necessidades técnicas do projeto.

3.3 Desenvolvimento da API

A linguagem e o framework escolhidos para rodar o servidor foi o Python com *Flask*. O motivo dessa escolha se deu a familiaridade e a facilidade, agilizando o desenvolvimento. Esse servidor, a princípio, foi criado em apenas um arquivo, na qual carrega os modelos de STT e TTS, se comunica com a API do *LM Studio* (Ferramenta utilizada para configurar e executar o LLM de forma simplificada), configurar as rotas consumidas e gerencia os fluxos de dados entre os modelos e as requisições.

As seguintes rotas na API foram definidas para garantir a interação completa entre cliente e servidor:

- POST */setup_npc*: Rota de inicialização de uma nova sessão de diálogo. É responsável por configurar o *prompt* de sistema do NPC, definindo sua personalidade, contexto, objetivos e o arquivo de áudio com a voz que será clonada e utilizada pelo npc.
- POST */chat*: Rota principal de interação. Recebe como entrada um arquivo de áudio com a fala do jogador, passa pelos modelos de IA e é criado um arquivo de áudio correspondente à resposta gerada pelo LLM. Como resposta da requisição, é retornado o nome do arquivo de áudio criado.
- GET */audio*: Rota para requisitar o arquivo de áudio da fala do npc gerado na etapa */chat*
- POST */add_context*: Rota para adicionar dados de contexto no histórico de conversa com a IA.
- POST */remove_context*: Rota para remover dados de contexto que foram adicionados na rota */add_context*

3.4 Desenvolvimento do *Serious Game*

Para o desenvolvimento da aplicação, foi escolhido o Unity (versão LTS 6000.0.3f1), principalmente por ser leve, ótima para jogos em RV e possuir uma comunidade de desenvolvedores bastante engajada.

3.4.1 Planejamento e estrutura base

Antes de ter exatamente qual seria o tipo de loja a ser representada, alguns pontos gerais já começaram a ser trabalhados na estrutura base da aplicação. Entre eles:

- Cenário: Para o cenário criado um ambiente simples e genérico, de forma que independente do tipo de loja o cenário poderia ser facilmente adaptado para se adequar a escolha.
- Controles: Para as interações em RV, inicialmente foi escolhido o sistema padrão de controles RV. Porém, logo em seguida foi substituído pelo asset *AutoHand*, asset este que traz vários sistemas prontos para uso como locomoção, interação com objetos, animação da mão com base no formato dos objetos e um sistema de *placepoints* para posicionar itens, o que acelerou bastante o desenvolvimento.
- Arquitetura de Software: Para garantir a manutenibilidade e escalabilidade do código, foram definidos padrões de arquitetura:
 - *Namespaces*: Foram utilizados para organizar os *componentes* em domínios lógicos distintos, facilitando a futura implementação de *Assembly Definitions* para otimizar a compilação.
 - *Singleton*: Adotado para os *managers* de cada sistema principal (ex: áudio, interação, comunicação), garantindo um ponto de acesso global e uma única instância de cada gestor.
 - *Data Container*: Foi utilizado o conceito de *ScriptableObjects* para armazenar configurações, como os dados de conexão da API. Isso permite alterar valores, como o endereço do servidor, diretamente no editor da Unity, sem a necessidade de recompilar o código.
- Sistema principal de comunicação com a API: Este sistema é composto pelos seguintes componentes:

- Configuração da API: *ScriptableObject* que guarda os dados de conexão (URL, porta e se é uma conexão *HTTPS*), facilitando a troca caso o servidor altere de endereço e ao mesmo tempo possibilitando consumir a API de forma remota.
- Cliente da API: Classe que realiza o serviço de comunicação direta com a API, formatando e enviando as requisições HTTP e recebendo as respostas.
- Orquestrador de Comunicação com a API: Classe que atua como uma camada de abstração, orquestrando o fluxo de dados. Este componente gerencia as informações a serem enviadas (como o *prompt* inicial e o áudio gravado do jogador) e direciona as respostas recebidas para os sistemas correspondentes, como a reprodução do áudio do NPC ou o acionamento de animações.

3.4.2 Definição do fluxo principal do jogo

O cenário de treinamento e a função que o jogador irá executar na aplicação escolhida, em conjunto com o orientador Fábio, foi a de atendimento em um ponto dos correios para o serviço de levantamento de encomendas. Esse cenário foi escolhido por ser simples e fácil de adaptar para outros contextos logísticos, como almoxarifados ou bibliotecas, já que o fluxo é bem parecido entre eles, seguindo de forma geral a mesma sequência representada na figura 2.

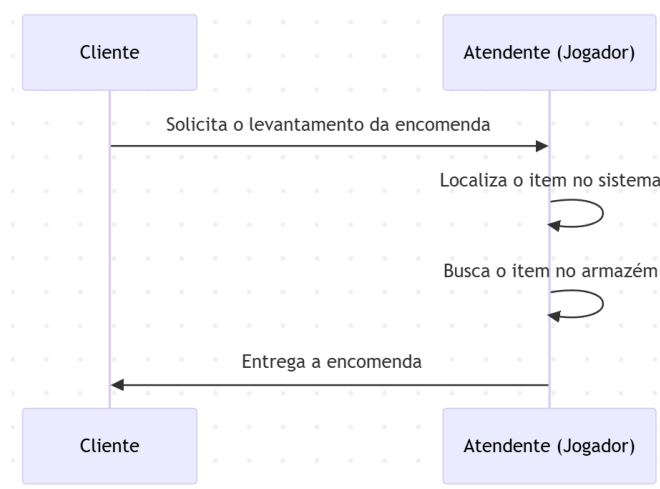


Figura 2: Diagrama de sequência do fluxo de atendimento de alto nível.

Para melhor representar este cenário, o serviço de levantamento de encomenda da empresa CTT serviram como referência, levantando informações dos procedimentos através de pesquisa na internet e até mesmo visitando uma unidade para observar e perguntar sobre os procedimentos. O fluxo levantado pode ser observado no anexo C. Esse passo a passo serviu de base para criar as tarefas e interações do *Serious Game*.

3.4.3 Desenvolvimento do protótipo

Após definir o fluxo de atendimento, o desenvolvimento na Unity foi retomado. O foco principal foi o planejamento e desenvolvimento de elementos visuais, como melhoria do cenário e escolha dos modelos 3D, assim como elementos com a lógica do jogo, como a criação de sistemas modulares e interconectados.

3.4.3.1 Ambiente do Armazém e Gestão de Pacotes

A partir do cenário básico criado anteriormente, foi adicionado uma nova sala para representar o galpão com as encomendas armazenadas. A sala do galpão foi criada com 14 prateleiras, cada uma composta por três andares e três espaços para pacotes em cada andar, totalizando 378 posições de armazenamento (Figura 3). Cada prateleira recebe uma letra (A-N) e cada andar e posição é identificado por um número (1-3), facilitando a localização dos itens.



Figura 3: Representação do armazém e o sistema de localização de encomendas.

Os pacotes são representados por *prefabs* (Objeto reutilizável com o modelo 3D e componentes já configurados, possibilitando instanciar este mesmo item várias vezes) controlados por um componente de gestão do pacote, que associa o objeto físico a um *ScriptableObject* contendo os detalhes do pacote. Esse objeto armazena informações como peso, código de rastreio, data limite para levantamento e a posição no armazém. O componente de gestão do pacote também ativa um componente de alvo de escaneamento, permitindo que o pacote seja lido pelo leitor de código de barras.

Ao iniciar a cena, *prefabs* dos pacotes com dados aleatórios são colocados em cada espaço das prateleiras. No início de cada atendimento, um dos pacotes no armazém é escolhido, de forma aleatória, para armazenar as informações configuradas do pacote do cliente. O gestor do armazém é o componente responsável por estas ações e por gerir tudo relacionado com o armazém.

3.4.3.2 Ferramentas e Objetos Interativos

Para representar alguns procedimentos do atendimento, foram desenvolvidas diversas ferramentas e objetos que o jogador pode interagir diretamente através dos controles de realidade virtual proporcionados pelo asset *Autohand*.

- Leitor de Código de Barras: Objeto representado por um modelo 3D de um scanner baixado na internet. Este objeto possui um componente que controla o dispositivo de leitura, que usa a função *Raycast* (uma linha invisível que detecta colisões) para identificar objetos que implementam a interface de objetos escaneáveis. Ao detectar tais objetos, o componente dispara um evento com as informações do objeto, que são enviadas para outros sistemas, como a interface do monitor. Os principais objetos escaneáveis que possuem a interface implementada são:
 - Documentos: Tanto o documento de identificação quanto a carta da encomenda são objetos 3D que o cliente pode apresentar durante o atendimento, representando, respectivamente, um “Cartão de cidadão” com foto e dados pessoais e uma carta com o aviso de chegada de uma encomenda. Cada um desses itens possui componentes específicos para o escaneamento e gerenciamento das informações:

ao escanear o documento de identificação, os dados do cliente são exibidos no monitor para o jogador, enquanto ao escanear a carta da encomenda, são apresentadas as informações relativas à encomenda do cliente. A Figura 4 ilustra a representação visual destes dois documentos.



Figura 4: Exemplo do Documento de Identificação e da Carta da Encomenda.

- Pacote: O prefab do pacote possui uma imagem com um código de barras. Essa imagem tem o componente de alvo de escaneamento, que possibilita utilizar o scanner para exibir as informações atribuídas a ele no monitor. Ao fazer o scan, um *feedback* sonoro indicando se o pacote coletado é o correto ou não.
- Monitor de Atendimento: Tela que apresenta informações contextuais ao jogador, o gestor do monitor de atendimento é o componente responsável por possibilitar a exibição das informações do cliente em atendimento e da encomenda a ser levantada. Também é possível navegar entre as telas através de botões na tela do monitor.
- Balança: Objeto com um *placepoint* que aceita apenas pacotes. Ao colocar um pacote neste *placepoint*, o peso é exibido em uma tela acoplada, simulando a pesagem. O componente da balança é responsável por ler os dados do pacote e mostrar o peso.

A Figura 5 demonstra a integração destas ferramentas em pleno uso: o jogador utiliza o leitor de código de barras num pacote posicionado sobre a balança, enquanto o monitor de atendimento exibe em tempo real as informações correspondentes àquela interação.

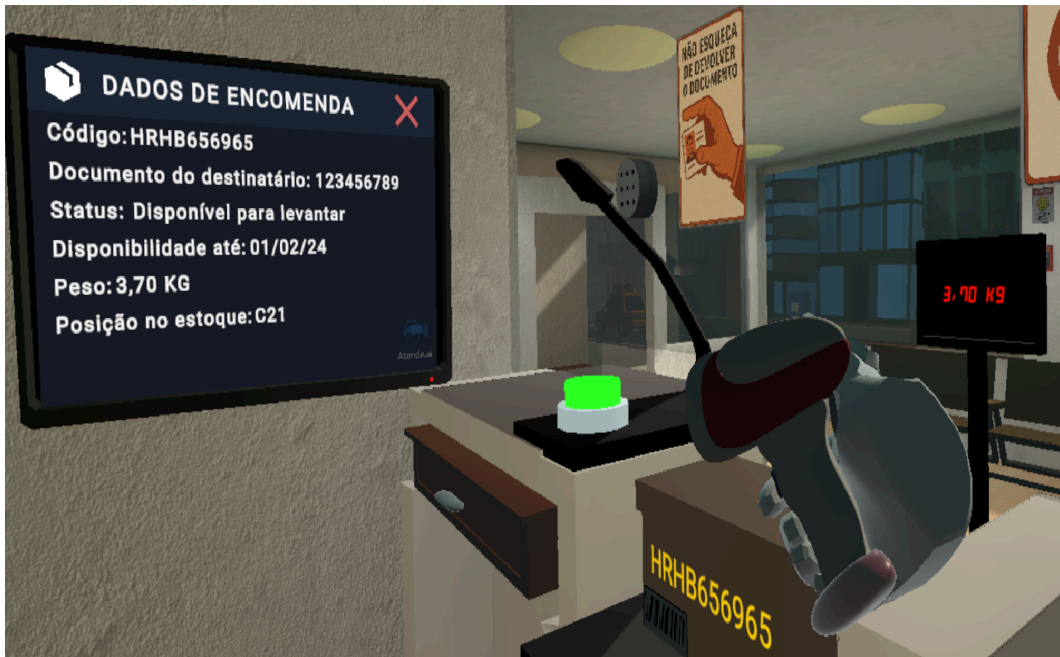


Figura 5: Integração das ferramentas interativas: leitor, balança e monitor de atendimento.

3.4.3.3 Configuração do NPC

O NPC representa o cliente que vai receber o atendimento, por isso é uma das implementações mais importantes do projeto. O sistema foi pensado para ser modular e flexível, juntando um modelo visual, sistemas de animação e movimento, uma estrutura de dados para definir a personalidade e um processador verifica se o LLM executou alguma das ações de interação pré definidas.

O avatar 3D do cliente foi criado utilizando o serviço “*Ready Player Me*” (Plataforma que permite criar avatares 3D personalizados para usar em jogos). Esse avatar já tem por padrão o *rig* (estrutura de controle que permite animar um modelo 3D) configurado, agilizando o desenvolvimento. Além disso, ele já inclui um sistema de *lip sync* integrado, facilitando a sincronização labial automaticamente com a fala do NPC.

A maior parte das animações foram originalmente criadas para este projeto, de forma que o NPC consiga representar visualmente ações como entrega de documentos, assinar a carta, coletar objetos e etc. O controle destas animações é realizado pelo componente controlador de animações do NPC, que usa o asset *Animancer* para ter um controle maior por código da reprodução das animações.

Esse componente gerencia quais animações devem ser reproduzidas com base no estado e nas ações do NPC.

O controlador de movimento do NPC, é o componente responsável pelo seu posicionamento, tem a função de definir em que momento e para onde o personagem deve se movimentar. Ele usa o *Navmesh* da *Unity* (Sistema de navegação que permite movimentação automática por rotas inteligentes no cenário), para guiar o NPC até os pontos importantes, como o balcão de atendimento, o local de entrega de encomendas e a saída da loja. Com isso, o NPC sempre se posiciona da forma correta em cada etapa do atendimento.

As informações e personalidade do NPC são definidas através de uma estrutura de dados parametrizável, um *ScriptableObject* que contém os detalhes do cliente. Nesta estrutura, são definidos tanto os dados de exibição (como nome e documento) quanto os parâmetros que afetam diretamente o comportamento do cliente, como o temperamento, paciência e familiaridade com o serviço solicitado. Cada uma dessas características comportamentais tem um texto descritivo associado, que é posicionado em um espaço específico dentro de um *prompt* modelo. Esse *prompt*, gerido por um componente de modelos de *prompt*, reúne o contexto e as instruções gerais que orientam o LLM. Tanto essas instruções quanto os textos descritivos passaram por testes e ajustes, com o objetivo de encontrar o texto que melhor instrui o comportamento do modelo. Um exemplo desses ajustes foi a necessidade de repetir a instrução de que o LLM deve agir como cliente, já que nos testes iniciais ele costumava assumir o papel do atendente, quebrando a imersão.

Para transformar a resposta do LLM em ações que o NPC deve executar dentro do jogo, foi criado o processador de ações do NPC. Ele funciona como a ponte final entre a decisão do LLM e a execução no jogo, analisando a resposta recebida em busca de tags de ação pré-definidas, como [AÇÃO:assinar_carta] ou [AÇÃO:recolher_documento_devolvido]. Quando encontra uma dessas tags, o processador dispara o evento correspondente, por exemplo, iniciar uma animação específica no controlador de animações do NPC ou interagir com objetos do cenário. O ciclo completo, desde a fala do jogador até a interpretação e execução de uma ação física pelo NPC, como a entrega de um documento, é detalhado na sequência de interações apresentada no Anexo A.

3.4.3.4 Gerenciadores e Lógica de Jogo

Com todos os elementos interativos e o NPC configurados, foi necessário desenvolver componentes para orquestrar todos sistemas, garantindo que as regras do atendimento fossem seguidas para proporcionar uma experiência mais dinâmica e responsiva às ações do jogador. Esta lógica foi centralizada nos seguintes sistemas:

- **Gestor da Sessão de Atendimento:** Esse sistema é implementado através do gestor da sessão de atendimento, que é responsável por definir em que etapa do atendimento o jogador está. Para isso, foi utilizada uma Máquina de Estados Finitos (FSM), onde cada etapa do serviço é um estado único (implementado a partir de uma interface de estado de serviço). Ele transita entre estados como *AwaitingDocsState* (Aguardando Documentos) ou *PackageLocatedState* (Pacote Localizado) apenas quando as ações corretas do jogador são validadas. Desta forma, o sistema garante que o fluxo do atendimento seja seguido corretamente, evitando saltos ou inconsistências.
- **Orquestrador da Interação:** O orquestrador de interações é o componente responsável por representar em cena o estado lógico atual que é definido pelo gestor de sessão. Enquanto o gestor da sessão de atendimento lida com a lógica abstrata, o orquestrador traduz essa lógica em eventos e ações visíveis no jogo. Ele comanda, por exemplo, o controlador de movimento do NPC para que o NPC se mova para os locais corretos em cada fase do atendimento, como o balcão ou o ponto de saída. Além disso, gerencia a ativação e desativação de interações, como habilitar o *PlacePoint* para a assinatura da carta apenas no momento apropriado, garantindo que as interações disponíveis para o jogador sejam coerentes com o estado atual.
- **Sistema de contextualização para o LLM:** Esse sistema tem como objetivo contextualizar informações dinâmicas sobre o estado do atendimento para o LLM, resultando em respostas que façam mais sentido com o cenário. Implementada através de um componente de enriquecimento de *prompt*, esta funcionalidade monitora os gestores do jogo para recolher dados em tempo real, como a etapa atual do atendimento ou os documentos que o jogador ainda não apresentou. Essas informações são então formatadas e enviadas para a API, conferindo ao LLM uma "consciência situacional". Com isso, o

NPC pode, por exemplo, gerar uma resposta para lembrar o jogador de um procedimento que foi esquecido, em vez de seguir um diálogo linear, o que eleva significativamente a imersão do jogador.

O funcionamento em conjunto de todos estes sistemas compõem a arquitetura completa do lado do cliente (*Client-Side*) da aplicação. A implementação na Unity em conjunto com a implementação da API, possibilita a execução de ponta a ponta do serviço de atendimento para levantamento de encomendas.

3.5 Melhorias e Refinamentos

Após o desenvolvimento do protótipo e ter o fluxo de atendimento completo implementado, foi iniciada uma fase de testes e refinamentos. Executar o mesmo fluxo várias vezes possibilitou a identificação de possíveis melhorias e vários problemas que podem comprometer a experiência do jogador. Essas observações durante a fase de teste motivaram o desenvolvimento de melhorias que visam aumentar a imersão, otimizar a performance e facilitar os ciclos de teste e desenvolvimento.

3.5.1 Refinamento da Interação por Voz

Inicialmente, a forma com que se iniciava e parava a gravação da fala do jogador era através de um simples clique no botão do controle do óculos RV. É uma abordagem funcional, mas nada imersiva e muito menos interativa. Para melhorar essa função, essa mecânica de gravação passou a ser executada por um objeto no cenário que simula um intercomunicador.

O modelo 3D desse intercomunicador se baseia em dois objetos, ambos modelados a partir do *ProBuilder* (Ferramenta integrada na unity que possibilita modelar objetos 3D partindo de formas primitivas), que representam a unidade interna e externa. O objeto da unidade externa é apenas um item decorativo, mas que visualmente complementa o da unidade interna, representando onde o cliente ouviria a fala do jogador e contribuindo para a coerência visual do ambiente. O objeto da unidade interna possui um grande botão, que o jogador interage ao pressionar utilizando o controle, na qual muda de cor para fornecer um *feedback* visual claro sobre o estado da interação. O comportamento desse botão é controlado pelo gestor de entrada de

voz, que se comunica com outros sistemas para definir quais dos seguintes estados o botão vai estar:

- Indisponível (Cor Cinza): O botão muda para esta cor quando não é possível iniciar uma nova gravação de voz para a api. Os dois motivos para isso ocorrer são: Enquanto o NPC está falando ou durante o período em que a fala do jogador ainda está em processamento na API. Este estado evita interrupções e o envio de múltiplos pedidos simultâneos.
- Disponível (Cor Verde): A cor verde indica que o sistema está pronto e que o jogador pode pressioná-lo para começar a falar com o NPC.
- Gravando (Cor Azul): Quando pressionar o botão que estava com a cor verde, é alterado para azul, sinalizando que a fala do jogador já está sendo gravada. A gravação é realizada enquanto o botão estiver sendo pressionado, parando e enviando para a API apenas ao soltar. Porém, há uma regra implementada para prevenir toques acidentais no botão, na qual a gravação e envio para API são cancelados se o botão for pressionado por menos de 1 segundo.

Para fazer sentido ter um intercomunicador, foi necessário adicionar no cenário um vidro no balcão de atendimento que separa o jogador e o NPC, já que a unidade externa foi posicionada no vidro. Isso também necessitou de alteração da forma que o NPC coletava o pacote, já que a entrega em mãos do NPC não era mais possível por conta do vidro. Para resolver isso, foi criada uma abertura no vidro com uma pequena porta que se abre quando o cliente vai coletar a encomenda. Essa solução necessitou algumas alterações nos outros sistemas, já que o NPC agora precisa entender quando deve se movimentar para o ponto de coleta. A nova disposição do balcão, que inclui a barreira de vidro e o microfone do intercomunicador, é ilustrada na Figura 6.



Figura 6: Balcão de atendimento com a barreira física e o sistema de intercomunicador implementados.

3.5.2 Otimização da Latência e Modularização da API

A latência da API foi um dos problemas mais perceptíveis durante os testes, afetando diretamente a experiência do jogo. Em média o tempo entre a fala do jogador e a reprodução do áudio com a fala do NPC era de aproximadamente 13 segundos. Esse tempo impacta diretamente na imersão, já que prejudica muito a naturalidade da conversa. Para entender qual etapa era o gargalo da API, foi adicionado indicadores de tempo para a entrada e saída de cada modelo, na qual o STT e TTS apresentaram tempos muito maiores em comparação ao LLM. Os modelos destas etapas (Whisper e XTTS-V2) estavam sendo executados, por configuração errada, inteiramente na CPU ao invés de usar a GPU.

Várias tentativas de correção dos modelos foram feitas para usar a GPU. Porém, por conta de vários problemas de compatibilidade de bibliotecas e hardware encontrados neste processo, optou-se por modificar a arquitetura do servidor. Essa alteração permitirá a seleção do modelo que será utilizado em cada etapa, além de facilitar a adição de um novo modelo para determinada etapa. Para isso, o servidor deixou de ser um único arquivo python, que executava todas as funções do servidor, para ser dividido em três camadas principais:

- Camada de Roteamento (server.py): Ponto de entrada da aplicação, responsável por definir os *endpoints*, gerir o ciclo de vida das requisições e orquestrar a comunicação entre os diferentes módulos de lógica.
- Camada de Lógica de Negócio (*Handlers*): Módulos que contêm a lógica específica para cada serviço de IA (STT, LLM, TTS) e para o gerenciamento de sessões e cache de arquivos.
- Camada de Modelos de IA (*Models*): Classes que encapsulam e abstraem a complexidade da interação com os diferentes modelos de IA e serviços externos, sejam eles locais ou na nuvem. A nova arquitetura de componentes da API, agora modular, é apresentada no Anexo B.

Esta modularização permitiu a substituição dos modelos locais por alternativas em nuvem, de forma que ainda seja possível alterar para execução local facilmente. Foi integrado o Google Cloud para o STT e TTS, utilizando os serviços via API, de forma gratuita por conta dos créditos para estudantes. O uso destes modelos, mesmo em nuvem, resultou em uma queda drástica do tempo de resposta, passando a ser em média 7 segundos, valor um pouco alto ainda, mas muito mais aceitável em comparação com o anterior. A Figura 7 detalha o fluxo de dados dentro desta nova arquitetura durante uma requisição de chat.

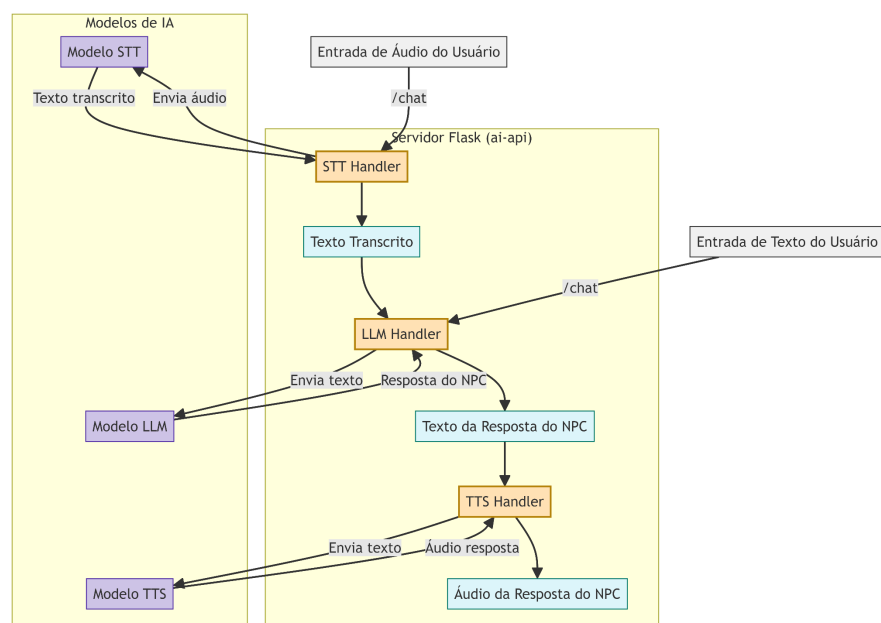


Figura 7: Fluxo de Dados na API Modular. Ilustra o pipeline de processamento da informação, desde a entrada do usuário até a resposta final, passando pelos Handlers STT, LLM e TTS e seus respectivos modelos de IA.

Essa nova arquitetura também permite utilizar modelos de LLM via API. Testes foram realizados considerando esta mudança e substituindo o LLama 3.1 local pelo modelo *Gemini 2.5 Flash* em nuvem. Porém, como os *prompts* de sistemas foram planejados e testados para o Llama, o gemini não apresentou bons resultados, gerando respostas muito longas e “alucinando” dados e eventos fora do contexto fornecido. Por isso, foi escolhido manter o LLama 3.1 localmente e alterar apenas o STT e TTS.

3.5.3 Sistema de *Feedback*

O *feedback* é um ponto importante no processo de treinamento. Por isso é uma funcionalidade interessante de ser adicionada na aplicação, onde ao final do atendimento, seria avaliado a interação que o jogador fez com o NPC e apresentar uma pontuação, juntamente com uma justificativa.

Para isso, foi desenvolvido a nova rota */score* na API. Esta rota foi projetada para ser chamada no final do atendimento, enviando para o LLM o histórico completo da conversa da sessão e um *prompt* de instruções, solicitando a avaliação da conversa com determinados critérios e formatos para a resposta. No Unity, o orquestrado de interações foi modificado para, no final do atendimento, chamar este *endpoint* e, com o resultado obtido, acionar um novo gestor criado para a exibição da pontuação no monitor, mostrando a avaliação com a nota e a justificativa no monitor.

Embora implementada a infraestrutura de *score*, tanto na API quanto na Unity, essa feature não pode ser considerada finalizada. Poucos testes foram realizados, além de que precisaria de um processo de refinamento do *prompt* de instrução para ter um *feedback* bem feito por parte do LLM. Devido às limitações de tempo do cronograma, e para priorizar a estabilidade da experiência central da simulação, esta tarefa de calibração e ajuste fino foi definida como um trabalho futuro. A arquitetura implementada é robusta e modular, permitindo que esta funcionalidade seja facilmente retomada e aprimorada em um trabalho futuro.

4. Resultados Obtidos

O protótipo desenvolvido “Atende.AI” atinge o objetivo apresentado na introdução de desenvolver um *Serious Game* em Realidade Virtual, utilizando Inteligência Artificial para criar diálogos focados no treinamento de atendimento ao público. Os resultados indicam que foi possível desenvolver uma solução funcional e interativa, que simula de forma imersiva um ambiente de atendimento.

4.1 Ambiente de Testes e Hardware Utilizado

Seguindo as especificações de hardware definidas na seção 3.1, a aplicação foi desenvolvida e testada no seguinte ambiente:

- Componente cliente (a aplicação em Realidade Virtual): Foi utilizado um notebook Dell G15 5530, equipado com um processador Intel i5-13450HX, 32GB de RAM DDR5 e uma GPU NVIDIA RTX 4050 com 6GB de VRAM. A escolha deste equipamento foi motivada pela sua potente GPU, necessária para garantir uma experiência de RV fluida e com altas taxas de quadros. Para a imersão e interação no ambiente virtual, foi utilizado um HMD (Head-Mounted Display) Meta Quest 3s, juntamente com os seus controles Meta Quest Touch Plus. A conexão com o computador foi estabelecida através de um cabo, utilizando a tecnologia Meta Quest Link, o que assegura uma comunicação de dados com a largura de banda e a baixa latência necessárias para a experiência em RV.
- Componente servidor (a API de IA): Um MacBook Air com o chip M3 e 16GB de memória unificada foi utilizado para executar o servidor junto com os modelos de IA. Este hardware foi selecionado devido à sua arquitetura de memória, que permite que a CPU e a GPU compartilhem a mesma RAM de forma eficiente, viabilizando a alocação de mais de 12GB de memória exclusivamente para os modelos de IA, algo crucial para rodar o LLM, o STT e o TTS simultaneamente.

4.2 Protótipo e a Experiência Imersiva

O principal resultado do projeto é um protótipo em RV, que consegue simular um atendimento para o fluxo de levantamento de encomendas. O jogador consegue

passar por todas as etapas do atendimento, que foram baseadas em procedimentos reais que são executados no CTT, em realidade virtual: Desde a recepção inicial do cliente, passando pelo uso de ferramentas como leitor de código de barras e balança, até finalizar o serviço.

É interessante observar como a combinação das tecnologias conseguiram proporcionar uma experiência imersiva e dinâmica para esse tipo de treinamento. Abordagens mais tradicionais dificilmente proporcionam uma distância tão curta entre a prática e a realidade como essa aplicação conseguiu.

4.3 Conversação Dinâmica com IA

As interações com diálogos não pré-definidos são um dos pilares do projeto. A arquitetura cliente-servidor, que conecta a aplicação Unity a uma API em Python para o processamento da IA, provou ser uma abordagem eficiente. O uso do Llama 3.1, recebendo contexto das ações realizadas no jogo, em conjunto com os serviços de STT (*Speech-to-Text*) e TTS (*Text-to-Speech*), permitiu a criação de uma conversação por voz natural e contextualizada. Um ponto fundamental validado em todos os testes foi a coerência da IA em executar as ações programadas[AÇÃO:...]) em resposta direta aos comandos do jogador, demonstrando que o sistema de interpretação de comandos funciona consistentemente.

Para validar a flexibilidade do sistema e sua capacidade de simular diferentes desafios de atendimento, foi testado o fluxo de atendimento, assumindo o papel de jogador, com três diferentes configurações de comportamento para o cliente. Para cada um dos três cenários de cliente definidos (Colaborativo, Inseguro e Distraído), o fluxo de atendimento foi executado integralmente. Durante cada sessão de teste, o histórico da conversa em formato JSON e a tela de feedback com a pontuação gerada pela IA foram salvos para permitir uma análise qualitativa posterior do desempenho tanto do jogador quanto do sistema. A seguir, cada um desses cenários são detalhados.

4.3.1 Cenário: Cliente Colaborativo

O primeiro teste foi pensado para simular uma situação ideal, servindo como referência para o sistema. Nesse cenário, o NPC recebeu o perfil de "Cliente Colaborativo": alguém calmo, cooperativo e que já conhece o procedimento. O

objetivo principal foi garantir que o fluxo padrão de atendimento funciona bem quando tudo corre a favor, avaliando se a comunicação do jogador é clara e se o sistema consegue lidar com uma interação tranquila, sem complicações.

- Configuração do cliente: Temperamento calmo, paciência alta e familiaridade média com o processo.
- Análise da Interação: Este atendimento foi planejado para representar um cenário de baixo desafio, servindo como um fluxo de atendimento padrão. As características atribuídas ao cliente fizeram com que ele colaborasse com o jogador, resultando em uma interação fluida e direta. O principal desafio para o atendente aqui era manter a eficiência e a clareza, sem a pressão de lidar com um cliente problemático.
- Análise do *Feedback*: O LLM deu a nota de 92/100. O *feedback* elogiou a clareza do atendente, mas apontou como melhoria a possibilidade de fornecer mais informações sobre a encomenda antes de solicitar a assinatura.

```
{
  "role": "assistant",
  "content": "Bom dia. Eu vinha levantar uma encomenda, por favor. [AÇÃO:entregar_documento]"
},
{
  "role": "user",
  "content": "Ótimo, já achei você aqui no sistema. Pode recolher o seu documento"
},
{
  "role": "assistant",
  "content": "Perfeito, agradeço imenso! [AÇÃO:recolher_documento_devolvido]"
},
{
  "role": "user",
  "content": "Você teria a carta da encomenda que será levantada?"
},
{
  "role": "assistant",
  "content": "Sim, aqui está minha carta com o código. [AÇÃO:entregar_carta:XPT0123PT]"
},
{
  "role": "user",
  "content": "Ótimo, poderia assinar ela para mim enquanto busco a sua encomenda no armazém?"
},
{
  "role": "assistant",
  "content": "Claro, vou assinar aqui. [AÇÃO:assinar_carta]"
},
{
  "role": "user",
  "content": "Ótimo, obrigado pela assinatura! Pode buscar o seu pacote no ponto de coleta ao lado"
},
{
  "role": "assistant",
  "content": "Ótimo, vou lá buscar então. [AÇÃO:iniciar_coleta_pacote]"
},
{
  "role": "system",
  "content": "Contexto: Você recolheu sua encomenda de código XPT0123PT no ponto de coleta e voltou até o atendente para continuar a conversa"
},
{
  "role": "user",
  "content": "[System Signal: NPC retornou, gerar reação de agradecimento.]"
},
{
  "role": "assistant",
  "content": "Muito obrigado(a)! Já tenho a minha encomenda. [AÇÃO:finalizar_atendimento]"
}
```

Figura 8: Trecho do histórico de conversa com o 'Cliente Colaborativo' em formato JSON. As falas refletem a natureza cooperativa do cliente, resultando em uma interação fluida e direta.

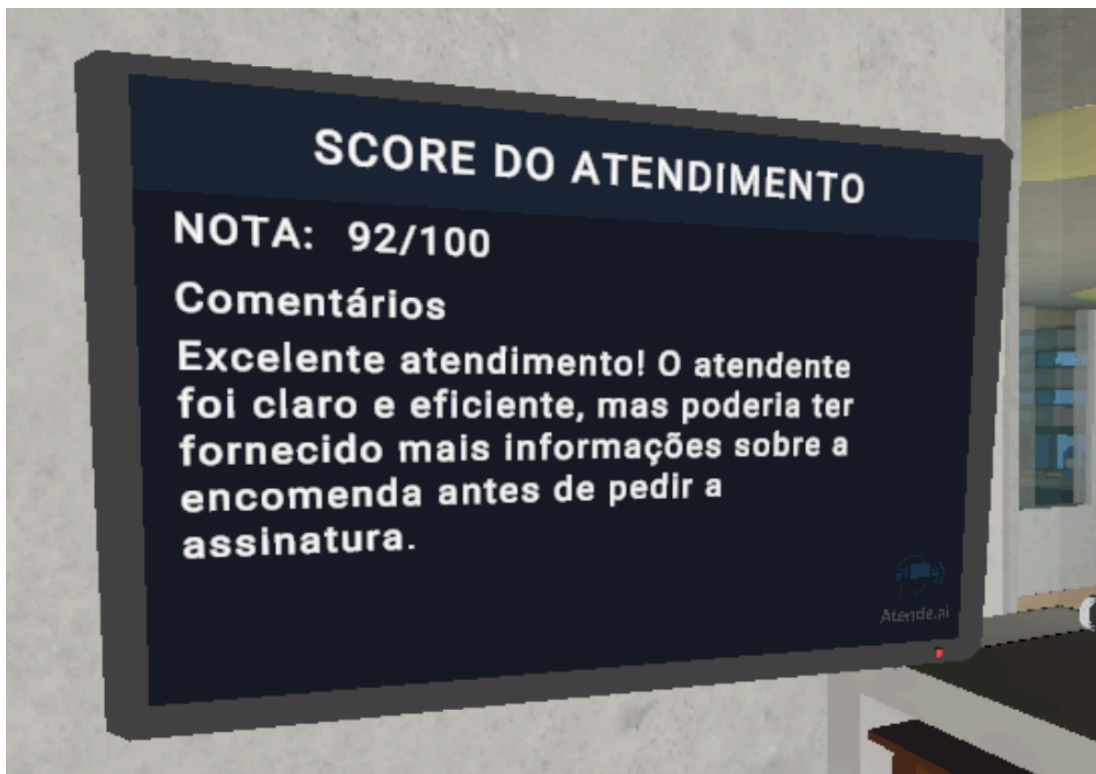


Figura 9: Tela de *feedback* para o cenário 'Colaborativo'. A nota elevada valida o bom desempenho do atendente, enquanto o comentário aponta uma melhoria pontual na comunicação.

4.3.2 Cenário: Cliente Inseguro

Para testar a paciência, empatia e clareza do jogador, o segundo cenário trouxe o "Cliente Inseguro". Apesar de calmo, esse cliente mostrou dúvida e não sabia direito o que fazer, exigindo que o atendente fosse mais explicativo e prestativo. O teste serviu para ver como o jogador se sai ao orientar alguém que precisa de ajuda em cada etapa, além de verificar se o sistema de feedback consegue identificar falhas importantes, como esquecer de pedir documentos.

- Configuração do cliente: Temperamento inseguro, paciência alta e familiaridade baixa com o processo.
- Análise da Interação: Este atendimento foi planejado para representar um cenário focado na capacidade de orientação e paciência do atendente. O cliente ficou hesitante e mostrou que não sabia o que fazer, então o jogador teve que ser mais ativo, explicando cada passo de forma simples e clara para ajudar o cliente. Aqui, a ideia é ver como o treinando se sai quando precisa dar mais suporte para alguém durante o atendimento.
- Análise do *Feedback*: O LLM deu a nota de 55/100. A justificativa critica o atendente por uma falha grave de segurança: não ter solicitado o documento

de identificação do cliente para confirmar sua identidade, etapa crucial no procedimento real dos CTT.

```
{
  "role": "user",
  "content": "Olá, como posso te ajudar hoje?"
},
{
  "role": "assistant",
  "content": "Bom dia... eu preciso levantar uma encomenda, certo?"
},
{
  "role": "user",
  "content": "Tudo bem, posso te ajudar com isso. Você teria a carta com o código de rastreio?"
},
{
  "role": "assistant",
  "content": "Sim, aqui tem a carta... é essa que eu preciso entregar? [AÇÃO:entregar_carta:XPT0123PT]"
},
{
  "role": "user",
  "content": "Ótimo, é esta mesma, Você poderia assinar a carta de rastreio para mim?"
},
{
  "role": "assistant",
  "content": "Claro, vou assinar aqui. [AÇÃO:assinar_carta]"
},
{
  "role": "user",
  "content": "Muito obrigado, espere só um momento enquanto vou buscar sua encomenda na armazém, ok?"
},
{
  "role": "assistant",
  "content": "Perfeito, agradeço imenso!"
},
{
  "role": "user",
  "content": "Muito bem, pode pegar seu pacote no ponto de coleta ao lado"
},
{
  "role": "assistant",
  "content": "Ótimo, vou lá buscar então. [AÇÃO:iniciar_coleta_pacote]"
},
{
  "role": "system",
  "content": "Contexto: Você recolheu sua encomenda de código XPT0123PT no ponto de coleta e voltou até o atendente para continuar a conversa"
},
{
  "role": "user",
  "content": "[System Signal: NPC retornou, gerar reação de agradecimento.]"
},
{
  "role": "assistant",
  "content": "Muito obrigado(a)! Já tenho a minha encomenda."
},
{
  "role": "user",
  "content": "Posso te ajudar em algo mais?"
},
{
  "role": "assistant",
  "content": "Acho que era só isso, muito obrigado! [AÇÃO:finalizar_atendimento]"
}
```

Figura 10: Trecho do histórico de conversa com o 'Cliente Inseguro' em formato JSON. As falas do cliente mostram a incerteza configurada na personalidade..

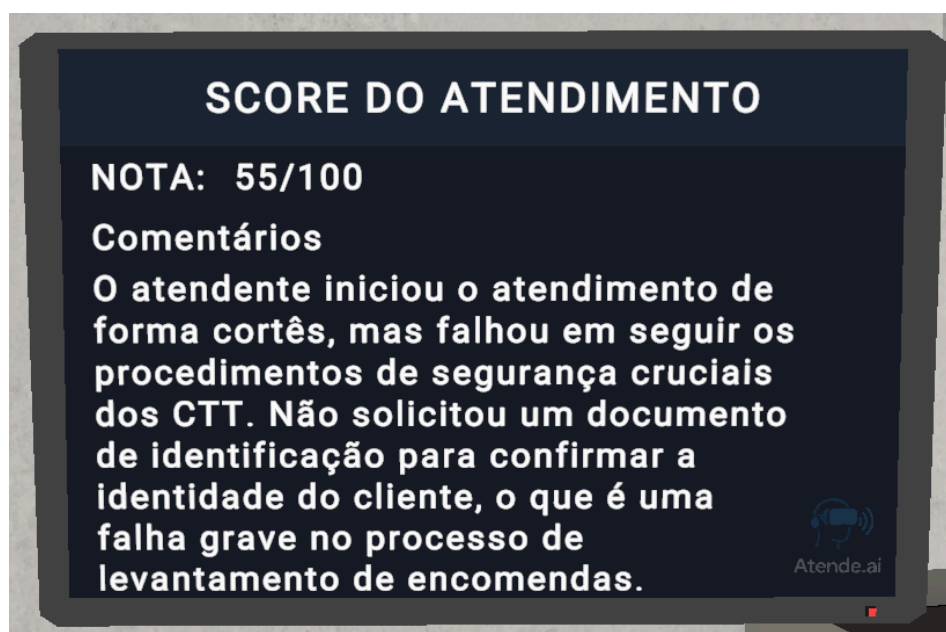


Figura 11: Tela de *feedback* para o cenário 'Inseguro'. A nota reflete uma falha crítica de procedimento (não solicitação de documento).

4.3.3 Cenário: Cliente Distraído

O último cenário foi criado para testar como o jogador lida com um "Cliente Distraído". Esse NPC tinha pouca familiaridade com o serviço e se distraía facilmente, trazendo o desafio de manter o atendimento no rumo certo. O objetivo era ver se o jogador conseguia manter o foco da conversa, sendo direto e educado para garantir que todas as etapas fossem cumpridas, mesmo com o cliente se perdendo no meio do processo.

- Configuração do cliente: Temperamento desatento, paciência alta e baixa familiaridade com o processo.
- Análise da Interação: Este atendimento foi planejado para representar um cenário focado em avaliar a habilidade do jogador em conduzir um atendimento com um cliente com baixo foco. O cliente estava disperso, então era necessário que o jogador assumisse o controle da conversa para garantir que tudo fosse feito do jeito certo. O foco foi testar se o atendente conseguia ser direto, mas educado, para manter o cliente no caminho e garantir que todas as etapas fossem cumpridas, mesmo com a distração.
- Análise do *Feedback*: O LLM deu a nota de 85/100. O comentário elogia o jogador por seguir todo o fluxo obrigatório corretamente. Como sugestão de melhoria, foi apontado que poderia ter sido mais proativo, perguntando se o cliente precisava de ajuda para encontrar a encomenda no ponto de coleta.


```

{
  "role": "user",
  "content": "Olá, como posso te ajudar hoje?"
},
{
  "role": "assistant",
  "content": "Bom dia... eu acho que vim levantar uma encomenda. Eu não lembro exatamente..."
},
{
  "role": "user",
  "content": "Te ajudo com isso, poderia me fornecer seu documento para verificar se há alguma coisa no sistema?"
},
{
  "role": "assistant",
  "content": "Claro, aqui está o meu documento! [AÇÃO:entregar_documento]"
},
{
  "role": "user",
  "content": "Acho que é isso mesmo que você veio fazer aqui, pois olhei seus dados no sistema e tem uma encomenda pendente para levantamento."
},
{
  "role": "assistant",
  "content": "Sim, é isso mesmo! Eu vinha levantar uma encomenda. E agora que você disse, eu lembro... é para mim! [AÇÃO:entregar_carta:XPT0123PT]"
},
{
  "role": "user",
  "content": "Ótimo, muito obrigado senhor Matheus, vou buscar a sua encomenda no galpão, mas enquanto isso o senhor poderia assinar essa carta que me deu?"
},
{
  "role": "assistant",
  "content": "Claro, vou assinar aqui. [AÇÃO:assinar_carta]"
},
{
  "role": "user",
  "content": "Encontrei seu pacote, enquanto faço a liberação dele para o senhor, pode coletar de volta o seu documento"
},
{
  "role": "assistant",
  "content": "Perfeito, agradeço imenso! [AÇÃO:recolher_documento_devolvido]"
},
{
  "role": "user",
  "content": "Pronto, pode buscar a sua encomenda no ponto de coleta ao lado, por favor"
},
{
  "role": "assistant",
  "content": "Ótimo, vou lá buscar então. [AÇÃO:iniciar_coleta_pacote]"
},
{
  "role": "system",
  "content": "Contexto: Você recolheu sua encomenda de código XPT0123PT no ponto de coleta e voltou até o atendente para continuar a conversa"
},
{
  "role": "user",
  "content": "[System Signal: NPC retornou, gerou reação de agradecimento.]"
},
{
  "role": "assistant",
  "content": "Muito obrigado! Já tenho a minha encomenda. [AÇÃO:finalizar_atendimento]"
}

```

Figura 12: Trecho do histórico de conversa com o 'Cliente Distraído' em formato JSON. A interação mostra o cliente lembrando do seu objetivo após a intervenção do atendente.

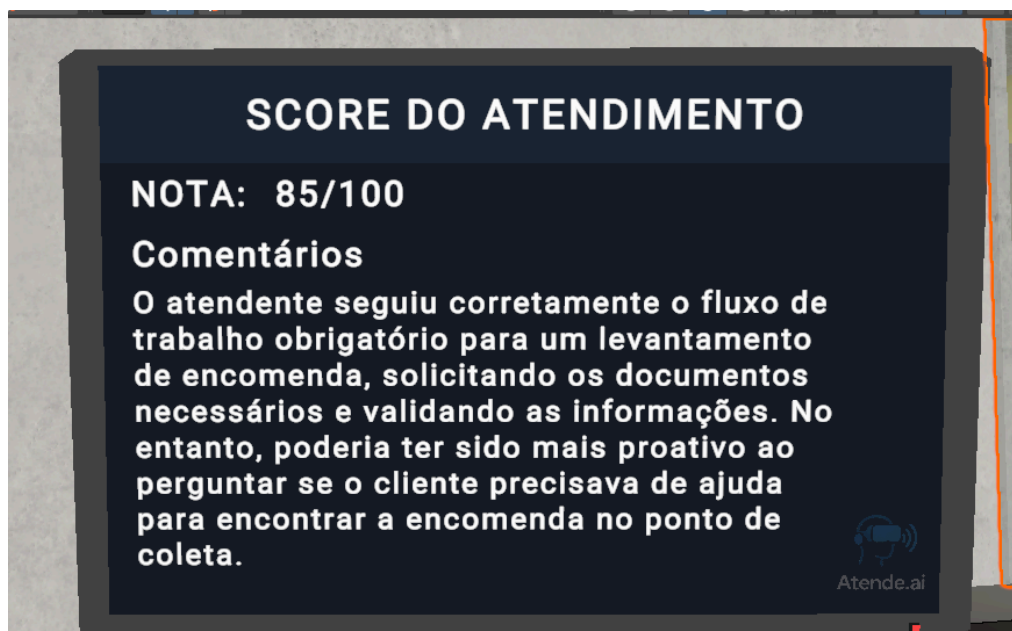


Figura 13: Avaliação de desempenho para o cenário 'Distraído'. O sistema elogia o cumprimento do fluxo, mas sugere melhorias na proatividade.

4.4 Potencial do Sistema de *Feedback*

O *feedback* é um ponto essencial no processo de treinamento. Como observado nos cenários de teste, foi desenvolvido um sistema que, com base no histórico da conversa e em critérios de avaliação definidos, utiliza o LLM para gerar uma pontuação acompanhada de uma justificativa.

Esse resultado mostra que a IA não só consegue dar uma nota, mas também fornecer uma análise qualitativa e construtiva. Isso valida o potencial da solução como ferramenta de desenvolvimento de competências, capaz de identificar nuances na comunicação e falhas processuais que seriam difíceis de avaliar automaticamente. No entanto, como detalhado na seção 3.5.3, essa funcionalidade ainda precisa de ajustes nos *prompts* e de uma calibração mais extensa, ficando como trabalho futuro.

4.5 Otimização da Latência

Um dos principais desafios técnicos enfrentados durante o desenvolvimento foi a latência de aproximadamente 13 segundos na resposta da IA. Uma contribuição importante do projeto foi a otimização desse tempo. Ao reestruturar a API para um modelo modular que possibilita uma abordagem mista entre local e nuvem, foi possível alterar os modelos de STT e TTS pelos serviços do Google Cloud e diminuir a latência total para uma média de 7 segundos. Essa redução de quase 50% foi significativa, tornando a interação mais fluida, embora ainda haja espaço para melhorar esse tempo em versões futuras.

No geral, os resultados mostram que os objetivos do projeto foram, em grande parte, alcançados, resultando num protótipo sólido que serve como prova de conceito para o uso de RV e IA no treinamento profissional. A capacidade de simular diferentes perfis de clientes e de fornecer *feedbacks* detalhados e contextualizados valida a aplicação como uma ferramenta de capacitação eficaz e inovadora.

5. Conclusão

O “Atende.AI” atingiu o objetivo principal ao criar um *Serious Game* funcional em Realidade Virtual, integrando Inteligência Artificial para treinar o atendimento ao público. Ao juntar essas tecnologias, o protótipo mostrou ser mais eficaz que os métodos tradicionais ao oferecer uma experiência de capacitação dinâmica e imersiva.

Entre as principais conquistas do trabalho estão a criação de uma arquitetura cliente-servidor robusta e a redução da latência das respostas da IA para cerca de 7 segundos, o que deixou o diálogo mais natural.

Este projeto tem impacto direto no meio profissional, ao apresentar um protótipo que serve como uma prova de conceito robusta. Ele demonstra o potencial da combinação entre a Realidade Virtual e a Inteligência Artificial para a criação de ferramentas de treinamento dinâmicas, imersivas e eficazes. Embora não seja uma solução final, o trabalho desenvolvido valida a viabilidade técnica e representa uma contribuição relevante para a área de treinamento profissional.

Como trabalhos futuros, sugere-se a contínua otimização da latência e a expansão do simulador para abranger novos cenários de treinamento, aproveitando a modularidade da arquitetura desenvolvida.

6. Referências Bibliográficas

Almeida, L. H. N. de. (2019). *Imersão em Realidade Virtual através de um jogo RPG*.

Universidade de Brasília (UnB), Faculdade UnB Gama (FGA).

Antognoli, D., & Fisher, J. (2023, junho 20). *A Proposed Taxonomy for the Design Qualities of Video Game Loading Interfaces and Processes*. Conference Proceedings of DiGRA 2023 Conference: Limits and Margins of Games Settings.

<https://doi.org/10.26503/dl.v2023i1.1928>

Barbato, D. S. (2016). Serious Games baseados em Simulações: Frameworks, definições e requisitos. *Proceedings of SBGames 2016*.

Barilli, E. C. V. C., & Cunha, G. G. (2003). *Aplicação Realidade Virtual para Formação Profissional Permanente de Recursos Humanos a distância, cuja Competência Exija o Desenvolvimento de Habilidades Motoras: Uma proposta de aplicação no campo da saúde*. Fundação Oswaldo Cruz (FIOCRUZ) e Universidade Federal do Rio de Janeiro (UFRJ/COPPE).

Buoro, D. M., Da Rocha, R. V., Belarmino, G. D., & Goya, D. H. (2021). Desenvolvimento de Jogos Orientado a Modelo para Jogos Sérios: Uma Revisão Sistemática. *Anais Estendidos do XX Simpósio Brasileiro de Games e Entretenimento Digital (SBGames Estendido 2021)*, 58–67.

https://doi.org/10.5753/sbgames_estendido.2021.19625

Dutra, A. Q. N., & Eboli, M. P. (2022). EDUCAÇÃO CORPORATIVA: UMA REVISÃO SISTEMÁTICA E BIBLIOMÉTRICA. *Anais do XLVI Encontro da ANPAD (EnANPAD)*.

Esteves, M. (2009). Construção e desenvolvimento das competências profissionais dos professores. *Sísifo. Revista de Ciências da Educação*, 08, 37–48.

Ferreira, A. C. S., Souza, F. R. V. de, Torres, I. da S., & Monteiro, M. A. (2024).

TREINAMENTO E DESENVOLVIMENTO COMO FERRAMENTA EDUCACIONAL

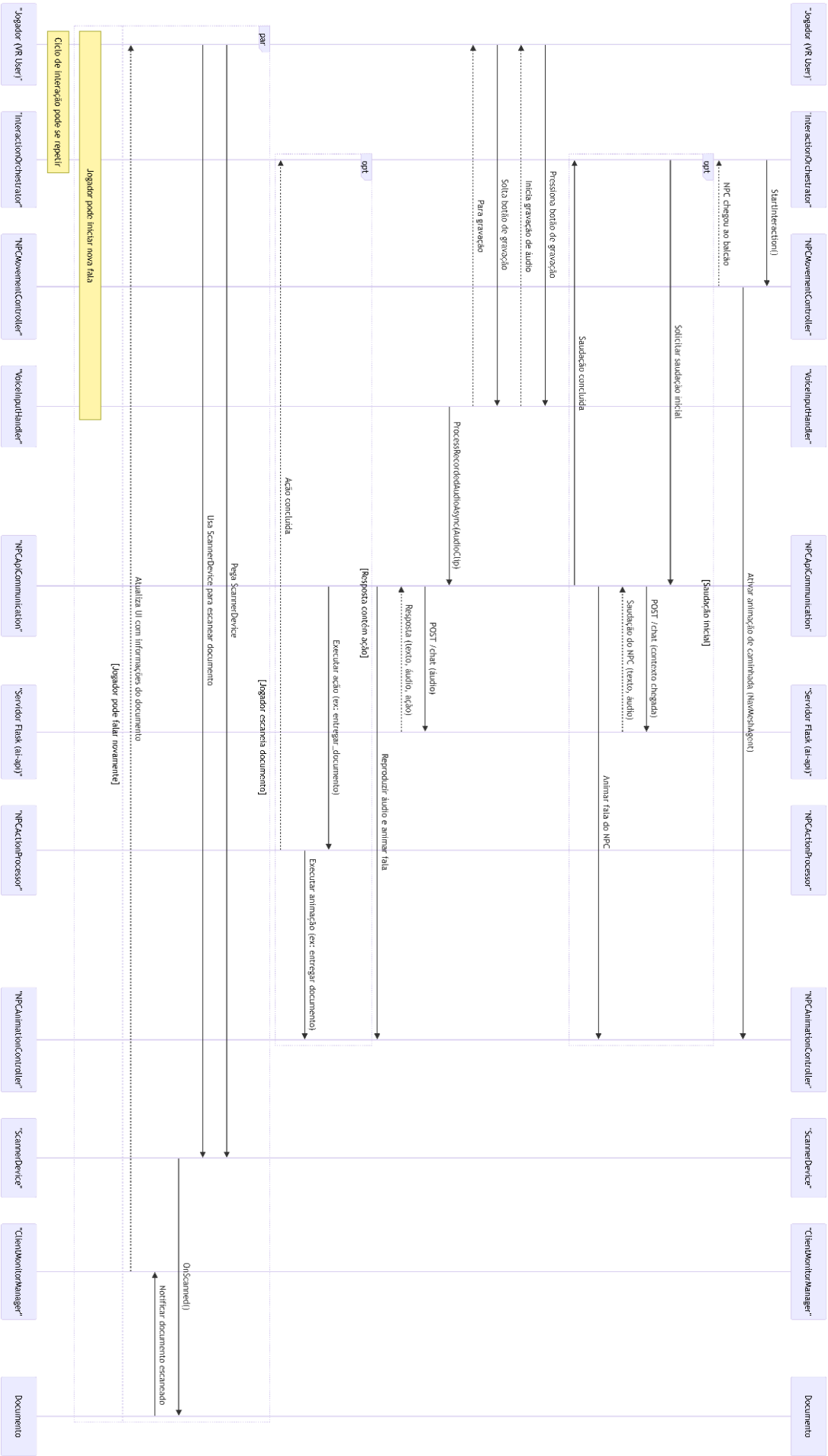
- PARA OS SERVIÇOS DE ATENDIMENTO AO CLIENTE. *Revista Educação & Ensino - ISSN 2594-4444*, 8(1). <https://doi.org/10.71136/ree.v8i1.729>
- Lacerda, M. I. S. D., & Casagrande, D. J. (2022). EXCELÊNCIA NO ATENDIMENTO AO CLIENTE: A importância da qualidade em serviços. *Revista Interface Tecnológica*, 19(2), 431–443. <https://doi.org/10.31510/infa.v19i2.1455>
- Lima, M. L. V. (2020). *GAMIFICAÇÃO E DESENVOLVIMENTO DE COMPETÊNCIAS PROFISSIONAIS DOS OPERÁRIOS DA CONSTRUÇÃO CIVIL*. Universidade Federal da Bahia (UFBA), Escola de Administração.
- Mendanha, T. S., Santos, S. A. D. F., & De Jesus, E. M. (2025). O USO DA INTELIGÊNCIA ARTIFICIAL COMO MECANISMO DE TREINAMENTO NA FORMAÇÃO MÉDICA EM ÁREAS REMOTAS DE GOIÁS: UMA PERSPECTIVA DE DESENVOLVIMENTO SOCIAL. *ARACÊ*, 7(2), 8223–8246. <https://doi.org/10.56238/arev7n2-214>
- Mourão, L., Abbad, G. D. S., & Zerbini, T. (2014). Avaliação da efetividade e dos preditores de um treinamento a distância em uma instituição bancária de grande porte. *Revista de Administração*, 49(3), 534–548. <https://doi.org/10.5700/rausp1166>
- Rocha, R. S. (2015). *A Importância do Treinamento no Atendimento aos Clientes Segundo a Perspectiva de Empresários do Município de Sabará*. Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Sabará.
- Rodrigues, J. de O. (2019). *O Impacto da Gamificação na Educação Corporativa em Empresas Brasileiras*. ISCTE Business School, Instituto Universitário de Lisboa.
- Rolim, F. M. (2023). *LLMs e IA Generativa em Jogos*. Universidade Federal do Rio de Janeiro (UFRJ), Instituto de Computação.
- Sá, B. M. S. de, Silva Júnior, R. J. da, & Nascimento, K. T. de M. (2022). *TREINAMENTO PROFISSIONAL: FERRAMENTA EMPRESARIAL PARA RESULTADOS EFICAZES*. Centro Universitário Brasileiro (UNIBRA).

- Silva, A. P. D., Lopes, A. F., Reis, F. V. W. D., & Catapan, M. F. (2024). Metodologia para aplicação de realidade virtual em treinamentos de integração de segurança industrial. *OBSERVATÓRIO DE LA ECONOMÍA LATINOAMERICANA*, 22(12), e8142. <https://doi.org/10.55905/oelv22n12-081>
- Silva, F. R. D. da. (2022). *ISA - INTERFACE DE SUPORTE À ACESSIBILIDADE: UM PROTÓTIPO DE CLOSED CAPTION ACESSÍVEL*. Universidade Estadual Paulista “Júlio de Mesquita Filho” (UNESP), Faculdade de Arquitetura, Artes, Comunicação e Design.
- Silva, L., & Natale, L. P. (2024). *VocationalLab: Una Experiencia Inmersiva con Realidad Virtual para Apoyo a la Orientación Vocacional*. 5(2).
- Silva, P. H. R. (2019). *INTELIGÊNCIA ARTIFICIAL APLICADA A JOGOS DIGITAIS*. Centro Universitário Atenas.
- Souza Figueiredo Junior, J., Florian, F., & Mirella Farina, R. (2024). INTELIGÊNCIA ARTIFICIAL (IA) NA REPRODUÇÃO DE VOZES HUMANAS: EXPLORANDO VANTAGENS E DESAFIOS. *Revista Científica Semana Acadêmica*, 12(251), 1–19. <https://doi.org/10.35265/2236-6717-251-13060>
- Souza, G. L. F. D., Souza, G. A. D., França, M. L. D. N., Grana, J. R., & Vieira, M. M. M. (2024). A REVOLUÇÃO TECNOLÓGICA NO ATENDIMENTO AO CLIENTE: O IMPACTO DA INTELIGÊNCIA ARTIFICIAL NA SATISFAÇÃO E EXPERIÊNCIA DO CONSUMIDOR. *Revista ft*, 29(140), 18–19. <https://doi.org/10.69849/revistaft/cl10202411201118>
- Stoque, D. C., Santos, E. R. P. dos, Stoque, K., & Dias, V. G. (2002). MÉTODOS DE TREINAMENTO. *REVISTA ELETRÔNICA DE ADMINISTRAÇÃO*, 1(3).
- Yezzi, R. M., & Barros Neto, J. P. D. (2025). A percepção da qualidade na visão do cliente como insumo para elaboração de levantamento de necessidades de treinamento

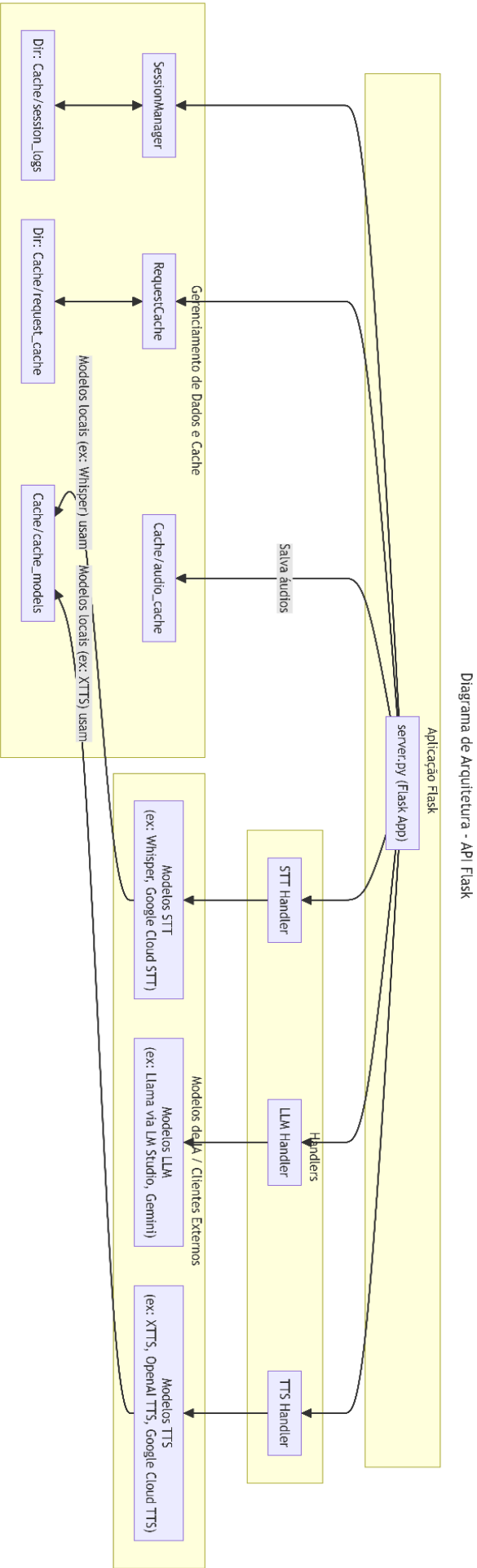
dos colaboradores pelo empreendedor de franquias. *Brazilian Journal of Business*, 7(1), e77217. <https://doi.org/10.34140/bjbv7n1-025>

Anexos

ANEXO A



ANEXO B



ANEXO C

