



**IX Encontro de Iniciação Científica e Tecnológica**  
**IX EnICT**  
**ISSN: 2526-6772**  
**IFSP – Campus Araraquara**  
**6 de dezembro de 2025**



## **Arquitetura Multiagentes para Tutoria Personalizada aplicada à disciplina de Programação**

FRANCISCO PEREZ NEVES<sup>1</sup>; RICARDO NOGUEIRA DE FIGUEIREDO<sup>2</sup>

<sup>1</sup> Ensino Médio, Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Campus Araraquara, [francisco.neves@aluno.ifsp.edu.br](mailto:francisco.neves@aluno.ifsp.edu.br)

<sup>2</sup> Mestre, Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Campus Araraquara, [ricardonogueira@ifsp.edu.br](mailto:ricardonogueira@ifsp.edu.br)

**Área de conhecimento** (Tabela CNPq): Ciência da Computação – 1.03.00.00-7

**RESUMO:** A evolução da Inteligência Artificial (IA) tem transformado significativamente diversos setores da sociedade, com destaque especial para a educação. No cenário educacional atual, o ensino de programação enfrenta diversos desafios, tornando complicado oferecer atenção individualizada para cada aluno. A integração de tecnologias baseadas em IA, através de sistemas tutores inteligentes, apresenta-se como uma solução atrativa para auxiliar estes desafios. Este projeto propõe um protótipo capaz de oferecer suporte contínuo aos estudantes, adaptando-se às necessidades individuais e tratando diferentes aspectos do processo de aprendizagem simultaneamente.

**PALAVRAS-CHAVE:** educação; inteligência artificial; modelo de linguagem de alta escala; programação; sistemas multiagentes; tutoria personalizada.

### **INTRODUÇÃO**

A evolução da Inteligência Artificial tem transformado significativamente diversos setores da sociedade, com destaque especial para a educação. Com o surgimento dos *Large Language Models* (LLMs) e sistemas multiagentes, caracterizados por agentes autônomos que colaboram por meio de protocolos de comunicação determinados para atingir objetivos predefinidos onde cada agente possui uma área de atividade sobre a qual tem certo controle [2], novas possibilidades surgiram para aprimorar e individualizar o processo de ensino-aprendizagem, especialmente na área de programação de computadores.

Atualmente, o cenário educacional referente ao ensino de programação enfrenta desafios relevantes, frente a turmas numerosas, professores lidam com diferentes níveis de conhecimento e ritmos de aprendizado, tornando difícil oferecer atenção individualizada a cada aluno. Além disso, a natureza abstrata dos conceitos de programação e a necessidade de prática constante demandam um acompanhamento mais próximo e particularizado.

Diante deste contexto, o objetivo deste trabalho consiste em desenvolver um sistema de tutoria inteligente baseado em multiagentes de IA para o ensino de programação de computadores que, posteriormente, servirá como plataforma de testes para avaliar a implicação de tal tutoria na disciplina.

### **FUNDAMENTAÇÃO TEÓRICA**

Os desafios contemporâneos da educação são marcados pela crescente diversidade dos perfis dos estudantes, moedas importantes de atenção individualizada e pela necessidade de adaptação constante aos avanços tecnológicos e metodológicos. Nesse contexto, a personalização do ensino (tutoria) se destaca como

uma estratégia essencial para garantir que cada aluno receba suporte adequado ao seu ritmo e estilo de aprendizagem [3].

A tutoria em programação envolve o desenvolvimento de habilidades práticas, pensamento computacional e capacidade de resolução de problemas [1], estruturados por estudos na área, que enfatizam a importância de elementos como *feedback* imediato, exemplos práticos e suporte contínuo para o sucesso no aprendizado [5]. A análise automática de código, componente fundamental, baseia-se em técnicas de processamento de linguagem natural e análise estática, permitindo uma avaliação abrangente que identifica desde erros sintáticos até problemas de lógica [4]. O monitoramento do progresso dos alunos, por sua vez, requer a consideração de múltiplas métricas e indicadores, possibilitando a identificação de padrões de aprendizado e a personalização do ensino.

Os sistemas multiagentes emergem como uma arquitetura robusta para a implementação de uma tutoria, permitindo uma abordagem distribuída, especializada e adaptativa para diferentes aspectos do processo educacional [2]. No âmbito educacional, esta abordagem permite que funções específicas, como análise de código, suporte conceitual e monitoramento de progresso, sejam tratadas por agentes especializados.

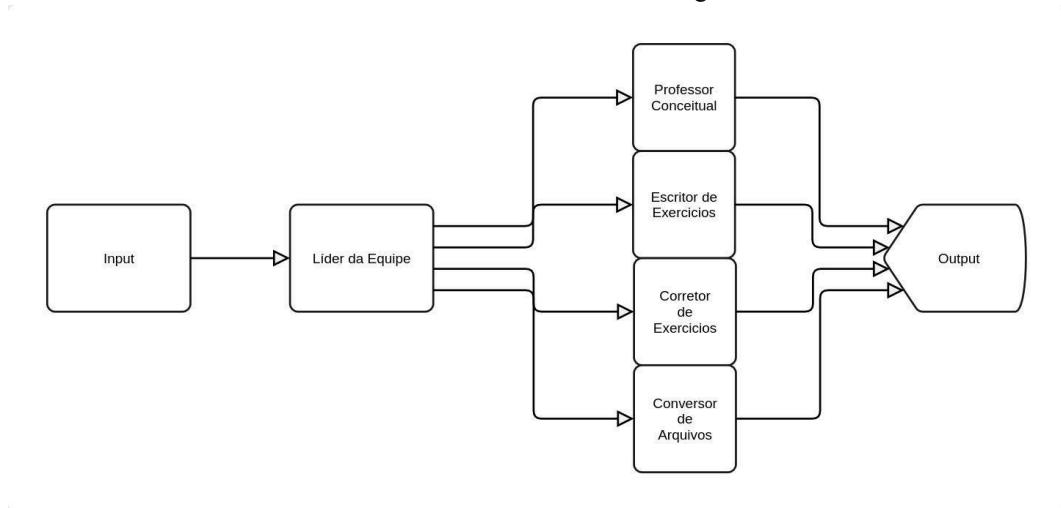
## METODOLOGIA

Este trabalho foi conduzido em fases sequenciais, iniciando com uma revisão bibliográfica abrangente sobre sistemas de tutoria inteligente na literatura científica, o uso de modelos LLMs empregados na educação e arquiteturas multiagentes, estabelecendo assim a base teórica necessária para a construção deste trabalho.

Logo após, foi esboçado a estrutura de agentes necessária para a execução de uma tutoria individualizada, sendo que cada agente possui uma responsabilidade específica. Os agentes foram estruturados considerando as diferentes dimensões do aprendizado construtivista nos quais o projeto se baseia. Para isso, foram criados um Professor Conceitual, um Escritor de Exercícios, um Corretor de Exercícios, um Conversor de Arquivos e o Líder da Equipe. Suas tarefas são ensinar conceitos, preparar material para fixação, corrigir exercícios, converter para arquivos e liderar a equipe, respectivamente.

A plataforma atua de maneira que, cada solicitação do usuário passa pelo Líder da Equipe, sendo posteriormente redirecionado para o agente mais adequado, de acordo com a entrada. O fluxo completo é demonstrado na Figura 1.

FIGURA 1: Fluxo de trabalho dos agentes



Fonte: Elaborada pelo autor

Na fase de implementação, foi implementado o protótipo de *software* utilizando uma biblioteca que permite definir e orquestrar agentes de IA. Concomitantemente, foi selecionado o modelo LLM GPT-4o-mini da empresa OpenAI<sup>1</sup> para servir de apoio aos agentes.

Durante pesquisas iniciais, foi analisada a opção da biblioteca CrewAI<sup>2</sup> para o desenvolvimento deste projeto, entretanto com pesquisas mais aprofundadas foi escolhido o framework Agno<sup>3</sup> devido ao número maior de funcionalidades permitidas. Além disso, no protótipo foi implementada uma interface Web para o software utilizando a biblioteca Flask<sup>4</sup>.

## RESULTADOS E DISCUSSÃO

O principal resultado preliminar do projeto foi o desenvolvimento do sistema exibido abaixo na Figura 2.

FIGURA 2: Exemplo de conversa no protótipo.

The screenshot shows a web-based application titled 'Sistema de Tutoria'. On the left sidebar, there are buttons for 'Sobre o Sistema', '+ Nova Sessão', and 'Últimas Sessões' (with 'Sessão 1' and 'Sessão 2' listed). The main content area has a title 'Claro! Vamos falar sobre estruturas de repetição em programação.' and a section 'O que são Estruturas de Repetição?'. It explains that loops are commands that allow a block of code to be executed multiple times without having to rewrite it. Below this, there's a section 'Como Funciona?' with an example in Python:

```
print(1)
print(2)
print(3)
# ... até 10
```

It notes that this is tedious and introduces a loop structure. A code example is shown:

```
for i in range(1, 11):
    print(i)
```

It explains what each part of the code does: 'for' is the loop keyword, 'i' is the variable, 'range(1, 11)' generates numbers from 1 to 10, and the code inside the loop is executed for each value of 'i'. There's also a section 'Outro Exemplo' with a 'while' loop example and a 'Resumo' section.

At the bottom, there are buttons for 'Anexar arquivos' and 'Digite sua dúvida'.

Fonte: Elaborada pelo autor

Durante testes iniciais feitos pelo autor e por seu orientador, o sistema demonstrou capacidade de:

- Adaptar-se ao estilo de aprendizado do aluno.
- Explicar conceitos de programação de forma clara e adaptada ao nível do aluno;
- Gerar exercícios personalizados com base nas dificuldades identificadas;
- Corrigir automaticamente códigos submetidos, fornecendo feedback imediato e sugestões de melhoria;
- Converter respostas em arquivos para facilitar o armazenamento e o acompanhamento.

<sup>1</sup> <https://platform.openai.com/docs/overview>

<sup>2</sup> <https://docs.crewai.com/>

<sup>3</sup> <https://docs.agno.com/introduction>

<sup>4</sup> <https://flask.palletsprojects.com/en/stable/#>

Do ponto de vista educacional, a arquitetura proposta mostrou-se promissora para lidar com a heterogeneidade de conhecimentos e ritmos de aprendizagem típicos de turmas de programação. A capacidade de oferecer suporte contínuo e personalizado reduz a dependência exclusiva do professor e possibilita que os alunos avancem em seu próprio ritmo, sem perder o engajamento.

## CONCLUSÕES

Este trabalho propôs e desenvolveu um protótipo de sistema de tutoria inteligente baseado em uma arquitetura multiagentes, com o objetivo de oferecer suporte personalizado ao ensino e servir de plataforma de testes para avaliar a implicação de uma tutoria personalizada utilizando IA na disciplina de programação.

Os resultados iniciais indicam que o sistema é capaz de lidar com a diversidade de perfis e ritmos de aprendizagem em turmas de programação, bem como fornecer explicações claras, gerar exercícios adaptados, corrigir códigos com feedback imediato e converter respostas em arquivos, atendendo a diferentes dimensões do processo de aprendizagem. A interface web também contribuiu para a acessibilidade e usabilidade do protótipo.

No entanto, também foram identificados desafios. A dependência de um modelo de linguagem comercial implica custos operacionais e questões de privacidade, que devem ser consideradas em implementações em larga escala. Além disso, vale ressaltar que a avaliação da eficácia pedagógica do sistema ainda requer aplicação em contexto real com alunos, coleta de dados de desempenho e comparação com grupos de controle.

Em comparação com outras ferramentas educacionais baseadas em IA, a abordagem multiagente adotada neste trabalho destaca-se pela modularidade e especialização, o que pode facilitar futuras expansões, como a inclusão de novos agentes para suportar a linguagens de programação específicas ou para o acompanhamento emocional do estudante.

## AGRADECIMENTOS

Manifesto meu profundo reconhecimento à Deputada Thainara Faria, cujo apoio por meio da Emenda Parlamentar 2025.265.66325 e do Termo de Fomento nº 003/2025 foi crucial para a concessão da bolsa de pesquisa que viabilizou este trabalho. Este investimento é um testemunho do seu compromisso com a ciência e a educação pública.

## REFERÊNCIAS

- [1] MALIK, A. R.; PRATIWI, Y.; ANDAJANI, K.; NUMERTAYASA, I. W.; SUHARTI, S.; DARWIS, A. (2023). Exploring artificial intelligence in academic essay: Higher education student's perspective. International Journal of Educational Research Open, 5, 100296. Disponível em: <https://doi.org/10.1016/j.ijedro.2023.100296>.
- [2] NADRLJANSKI, M.; VUKIC, D.; NADRLJANSKI, D. (2018). Multi-agent systems in e-learning. 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, pp. 990-995, doi: 10.23919/MIPRO.2018.8400181. 0990-0995. 10.23919/MIPRO.2018.8400181.
- [3] OLIVEIRA, N. de L.; LEITE, B. S. ANÁLISE DOS CRITÉRIOS PARA UMA EDUCAÇÃO PERSONALIZADA EM ARTIGOS DA ÁREA DE ENSINO PUBLICADOS ENTRE 2010-2020. *Revista Exitus, /S. l./*, v. 11, n. 1, p. e020197, 2021. DOI: 10.24065/2237-9460.2021v11n1ID1594.
- [4] SUN, D.; BOOUUAIA, A.; ZHU, C.; LI, Y. (2024). Would ChatGPT-facilitated programming mode impact college students' programming behaviors, performances, and perceptions? An empirical study. International Journal of Educational Technology in Higher Education. Disponível em: <https://doi.org/10.1186/s41239-024-00446-5>.
- [5] YILMAZ, R.; YILMAZ, F. G. K. Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning. Computers in Human Behavior: Artificial Humans. Disponível em: <https://doi.org/10.1016/j.chbah.2023.100005>.