

Uma Abordagem para o uso de Chats Inteligentes e Design Instrucional como Apoio à Aprendizagem de Programação

Cláudio Roberto Ribeiro¹, Rodrigo Bonacin^{1,2}, Julio Cesar dos Reis³

¹ Centro Universitário Campo Limpo Paulista (UNIFACCAMP)
Campo Limpo Paulista – SP – Brazil.

²Centro de Tecnologia da Informação Renato Arhcer (CTI)
Campinas – SP – Brazil.

³Instituto de Computação, Universidade Estadual de Campinas (UNICAMP)
Campinas – SP – Brazil.

crrprof@gmail.com, rbonacin@cti.gov.br, jreis@ic.unicamp.br

Abstract. *Teaching programming is a recurring challenge due to low performance, low motivation, and high student dropout rates. This study investigates how intelligent chat systems can help in teaching programming, the acceptance of this technology, and how it can influence student performance. We propose integrating intelligent chats with Virtual Learning Environments and adopting instructional design models. The approach was implemented in a system and evaluated with programming students in technical education, who highlighted good acceptance of the technology and associated benefits and challenges.*

Resumo. *O ensino de programação é um desafio recorrente devido ao baixo desempenho, a baixa motivação e aos altos índices de evasão dos alunos. Este estudo investiga como os sistemas de chats inteligentes podem auxiliar no ensino de programação, a aceitação desta tecnologia e como exerce influência no desempenho dos estudantes. É proposta uma abordagem que inclui a integração de chats inteligentes com Ambientes Virtuais de Aprendizagem e a adoção do modelo de design instrucional. A abordagem foi concretizada em um sistema e avaliada com estudantes de programação no ensino técnico, que destacaram uma boa aceitação da tecnologia, bem como benefícios e desafios associados.*

1. Introdução

Aprender programação de computadores pode ser uma tarefa desafiadora para muitas pessoas de diferentes idades, uma vez que envolve o domínio de conceitos teóricos, como lógica de programação e algoritmos, bem como habilidades práticas de escrita de código [Carreira et al. 2022]. A aprendizagem de programação apresenta diferentes desafios, especialmente para aqueles que estão iniciando na área, tais como os relacionados à lógica, sintaxe, complexidade, falta de prática, e falta de recursos, como ambiente de aprendizagem limitado e ausência de orientação e suporte adequado [Lin 2022].

Existem diversas metodologias que podem ser utilizadas para ensinar e aprender programação de computadores (e.g., a aprendizagem baseada em projetos e problemas). A escolha da mais adequada dependerá do perfil e da necessidade do estudante. É importante ressaltar que metodologias, diretrizes e recursos tecnológicos podem ser combi-

dados e adaptados conforme a necessidade de cada estudante, visando proporcionar uma abordagem de aprendizado mais personalizada e eficaz [Banić et al. 2023].

Para ensinar programação utilizando computadores [Bispo Jr. et al. 2020], faz-se necessário buscar abordagens inovadoras que possam motivar e apoiar os estudantes, tornando a aprendizagem de programação mais efetiva e acessível. Nesse sentido, a utilização de chats inteligentes pode ser uma solução promissora, por permitir que os alunos tenham acesso a um ambiente mais interativo e personalizado, uma vez que os chats podem auxiliá-los na compreensão dos conceitos, tirar dúvidas e na resolução de problemas de programação a qualquer momento, sem limitações de tempo ou disponibilidade [Ji and Yuan 2022].

Embora o uso de chats inteligente na aprendizagem seja uma área de pesquisa em crescimento, ainda há lacunas na literatura sobre o assunto [Sophia and Jacob 2021]. Por exemplo, uma lacuna é a avaliação da eficácia da abordagem, na prática, uma vez que embora haja muitos estudos que mostram que a abordagem de aprendizagem baseada em chats inteligentes é promissora, ainda há poucos estudos que efetivamente avaliam sua efetividade em sala de aula. Também existe a necessidade por pesquisas que articulem essa tecnologia com metodologias de ensino adequadas, bem como a criação de itinerários, métodos e ferramentas que explorem chats inteligentes em atividades em conjunto com outros recursos e Ambientes Virtuais de Aprendizagem (AVA).

Este artigo investiga e propõe uma abordagem para o uso de chats inteligentes, bem como implementa um sistema e avalia a aceitação e desempenho dessa tecnologia com estudantes do ensino técnico. Essa abordagem consiste em integrar chats inteligentes a AVA por meio de modelos e métodos de design instrucional. A abordagem é concretizada no design de um sistema que integra o ChatGPT 3.5 a um AVA baseado no WordPress especialmente desenvolvido para o ensino de programação. Ao concretizá-la foi considerado o modelo de design instrucional ADDIE (Análise, Design, Desenvolvimento, Implementação e Avaliação) [Martenstyaro and Rosmansyah 2015]. O ADDIE foi utilizado em dois momentos, primeiramente no desenvolvimento do AVA e integração com o chat inteligente e, posteriormente, na elaboração do conteúdo durante a avaliação com estudantes.

Entre as principais contribuições desta pesquisa se destaca a melhor compreensão dos benefícios (e desafios) no uso de chats inteligentes no ensino de programação. Em longo prazo, espera-se contribuir para melhorar o desempenho no processo de aprendizagem, bem como promover melhoria na qualidade dos desenvolvedores formados para trabalhar com habilidades necessárias ao uso de ferramentas inteligentes para aprimorar sua produtividade e desempenho em atividades de programação. Ainda assim, é importante ressaltar que o ChatGPT, neste trabalho, é visto como uma ferramenta complementar ao ensino atual de programação, e não como uma substituição [Wang et al. 2024].

O restante deste artigo está organizado da seguinte maneira: a seção 2 apresenta o referencial teórico e metodológico e os trabalhos relacionados; a seção 3 descreve a metodologia empregada e o desenvolvimento do sistema CodeChatPro; a seção 4 detalha a avaliação incluindo o método, os resultados obtidos e a discussão. Por fim, a seção 5 conclui o artigo e apresenta os trabalhos futuros.

2. Referencial Teórico e Metodológico e Trabalhos Relacionados

A subseção 2.1 apresenta conceitos do ensino de programação, enquanto a subseção 2.2 descreve a tecnologia de chats inteligentes; a subseção 2.3 relata conceitos e abordagens de design instrucional. A subseção 2.4 discute os trabalhos relacionados.

2.1. Ensino de Programação

A disciplina de programação é atualmente essencial não só para cursos de programação e tecnologia, mas também para formação em diversas áreas de atuação. Ela apresenta diversos desafios e a complexidade da programação pode parecer intimidante para iniciantes. É fundamental incentivar a persistência, dar feedback construtivo e demonstrar aos alunos como seus esforços levam a resultados tangíveis [Henno et al. 2023].

Essa complexidade está muitas vezes ligada ao desenvolvimento do pensamento lógico e à resolução de problemas. A complexidade está igualmente relacionada ao desenvolvimento da capacidade de abstração de conceitos. Outro aspecto a ser considerado é o estilo de aprendizagem, uma vez que os alunos têm diferentes estilos e é crucial adaptar abordagens e recursos para atender às necessidades individuais. Isso leva a refletir que não existe um único método, sendo importante adotar uma abordagem flexível [Wang et al. 2024].

Do ponto de vista tecnológico, a demanda por aprender diferentes linguagens no tempo tem se tornado um desafio. É importante ressaltar que o ensino de programação, envolve uma pluralidade de abordagens, métodos e tecnologias para transmitir e construir conhecimento sobre programação em conjunto com os alunos. Bem como, é importante enfatizar que, apesar dos desafios, os benefícios de aprender programação são inegáveis por desenvolver habilidades críticas e abrir portas para oportunidades em diversas áreas [Ji and Yuan 2022].

2.2. Chats Inteligentes

A história dos chats inteligentes remonta ao final dos anos 1960 com o desenvolvimento do ELIZA, um programa que imitava um psicoterapeuta [Lakshmi 2020]. Durante as décadas seguintes, os chats inteligentes continuaram a evoluir, sendo desenvolvidos programas que atendiam a uma variedade de funções. Nos últimos anos, os avanços na tecnologia e na inteligência artificial permitiram que os chats inteligentes se tornassem mais sofisticados e personalizados.

Chats inteligentes têm se beneficiado muito dos avanços na área de inteligência artificial, em particular no campo do Processamento de Linguagem Natural (PLN) [Ganesan et al. 2020], tais como os modelos de linguagem pré-treinados em grandes quantidades de dados e o aprendizado por reforço [Ashfaq et al. 2020]. Esses modelos, como o GPT-3/4 (*Generative Pre-trained Transformer*), podem ser usados para melhorar a qualidade das respostas de um chatbot. Já o aprendizado por reforço permite que um chatbot aprenda a partir da interação com os usuários [Finnie-Ansley et al. 2022].

Essas tecnologias têm um grande potencial de impacto em diversas áreas, desde assistentes virtuais, chatbots, análises de dados e apoio em processos de tomada de decisão. Nota-se que elas ainda apresentam limitações e desafios, como a possibilidade de gerar respostas enviesadas ou inadequadas em algumas situações, além de questões de privacidade e segurança.

O ChatGPT é uma implementação específica do GPT desenvolvida pela OpenAI para fins de conversação. Os GPT-3, GPT-3.5 e GPT-4.0 são modelos de linguagem de grande escala treinados com bilhões de parâmetros. O ChatGPT, por sua vez, é um chatbot que utiliza as capacidades de linguagem natural do modelo para interagir com os usuários de forma fluida e coerente. Ele pode responder a perguntas, dar sugestões, fornecer informações e realizar outras tarefas similares [OpenAI 2024].

2.3. Design Instrucional

Existem várias abordagens e modelos de design instrucional que orientam o processo de criação de experiências de aprendizado. São apresentados modelos e abordagens amplamente utilizados que foram considerados neste trabalho [Martenstyaro and Rosmansyah 2015]:

- *ADDIE*: é um processo que envolve as fases de análise, design, desenvolvimento, implementação e avaliação de materiais e programas de ensino. Cada fase do ADDIE tem um propósito específico no ciclo de design instrucional. Começa com a análise das necessidades de aprendizado e termina com a avaliação da eficácia do programa, permitindo ajustes contínuos.
- *SAM (Successive Approximation Model)*: é um modelo iterativo que enfatiza o desenvolvimento rápido de protótipos e feedbacks. Organiza o design instrucional em três fases principais: análise, design e desenvolvimento iterativo.
- *Gagne's Nine Events of Instruction*: se baseia em nove eventos de ensino sequenciais, desde a captação da atenção até a avaliação do desempenho. Cada evento tem um propósito específico, como a apresentação do conteúdo, a orientação da aprendizagem e a prática ativa.
- *Modelo Dick e Carey*: é um modelo que enfatiza a análise detalhada das necessidades e a criação de objetivo de instrução. O modelo inclui a análise de metas, design de instrução, desenvolvimento de materiais, implementação e avaliação.
- *Modelo 4C/ID (Four-Component Instructional Design)*: destaca quatro componentes essenciais, o aprendizado de tarefas autênticas, aprendizado *just-in-time*, aprendizado baseado em casos e aprendizado orientado pelo aluno.

A escolha do melhor modelo de design instrucional depende das necessidades específicas do público-alvo, dos objetivos de aprendizagem e do contexto em que a aprendizagem ocorrerá [Trilaksono and Santoso 2017].

2.4. Trabalhos Relacionados

Foi realizada uma revisão sobre o uso de chats inteligentes na programação de computadores. Essa revisão foi baseada no guia apresentado em [Kitchenham 2004] e descrita em [Ribeiro 2024]. A partir dessa, foi conduzida a seleção de trabalhos com objetivos, problemas e contribuições mais próximas às pretendidas neste estudo.

Finnie-Ansley *et al.* [Finnie-Ansley et al. 2022] exploraram as implicações do CODEX da OpenAI na programação inicial, comparando seu desempenho com o de estudantes. O trabalho destacou que a capacidade do CODEX superar a maioria dos estudantes e discute questões de integridade acadêmica. A contribuição do estudo está em fornecer insights sobre os desafios e oportunidades apresentados por tecnologias como o CODEX na educação.

Wang *et al.* [Wang et al. 2024] apresentaram um modelo de educação paralela, destacando o uso do ChatGPT para complementar o aprendizado em sala de aula. Portanto, contribuindo na exploração de um novo paradigma educacional e na identificação de oportunidades e desafios associados à implementação da educação paralela com o ChatGPT.

Lin & Tsai [Lin and Tsai 2019] apresentaram um chatbot baseado no *IBM Watson Assistant* e no *Facebook Messenger*, para auxiliar estudantes iniciantes de programação. A contribuição está na aplicação da tecnologia para melhorar o engajamento e interesse dos alunos. Já [Lin 2022] desenvolveu uma visão abrangente sobre a criação de chatbots para o ensino de programação, destacando a importância de considerar o perfil do usuário, a interação do chatbot e o contexto de suporte.

Luo & Gonda [Luo and Gonda 2019] abordaram a integração de chatbot no ensino superior, na criação de experiências de aprendizagem online. Os autores propuseram o uso de plataformas comerciais para simplificar o desenvolvimento de chatbot, permitindo que os professores se concentrem no diálogo em vez dos aspectos técnicos de IA.

Murad *et al.* [Murad et al. 2019] relataram um sistema que integra o chatbot com o aplicativo de mensagens LINE e um sistema de gerenciamento de conteúdo para estudantes e professores. A pesquisa apontou que o uso do chatbot em smartphones pode maximizar a aprendizagem do aluno, aumentando o interesse e o desempenho dele.

Ai *et al.* [Ai et al. 2020] descreveram a implementação de um chatbot educacional chamado *Ed.dy*, projetado para apoiar o ensino virtual e híbrido. O chatbot ofereceu recursos como lembretes, cronômetros e enquetes visando melhorar a experiência de aprendizado.

Carreira *et al.* [Carreira et al. 2022] apresentaram a avaliação do chatbot Pyo, desenvolvido para auxiliar estudantes iniciantes em programação. A pesquisa exploratória revelou que o Pyo foi útil, e apontou a necessidade de explicações mais diretas sobre suas funcionalidades.

Junior *et al.* [Junior et al. 2023] discutiram como o ChatGPT pode auxiliar no processo de aprendizagem de programação. Um estudo de caso quali-quantitativo foi realizado com a participação de alunos do ensino médio integrado ao técnico em informática. Os resultados demonstraram que o ChatGPT é uma tecnologia promissora ao fornecer fornecendo explicação de conceitos, exemplos de código, suporte e feedback personalizado.

Embora os artigos supracitados foquem elementos importantes para o avanço do ensino de programação com o uso de chats inteligentes, tais como estudos empíricos e integração com outras tecnologias, nenhum dos estudos abordou a integração de um AVA e chats inteligentes por meio de um modelo de design instrucional (no desenvolvimento do sistema e conteúdo) para o ensino de programação. Assim, a revisão realizada aponta que o presente estudo é original em endereçar essa linha de investigação.

3. Metodologia

Esta seção apresenta primeiramente a metodologia de pesquisa utilizada e a escolha do chat inteligente, modelo de design instrucional e AVA (subseção 3.1). Na sequência é apresentado o sistema CodeChatPro, incluindo o design, desenvolvimento e funcionalidades (subseção 3.2).

3.1. Metodologia da Pesquisa

A metodologia de pesquisa empregada foi estruturada em cinco etapas, que visam, a partir da literatura, propor e avaliar a abordagem proposta:

1. Revisão exploratória sobre o ensino de programação, definição de lacunas e identificação de dificuldades.
2. Revisão sistemática sobre uso de chats inteligentes no ensino de computação.
3. Análise comparativa para seleção de chat inteligente, tecnologia e modelo de design instrucional.
4. Design e desenvolvimento de um sistema que materializa a abordagem, bem como permite avaliação com estudantes.
5. Avaliação com estudantes do ensino técnico.

Os resultados das etapas 1 e 2 foram brevemente apresentados na última seção; os resultados da etapa 3 são apresentados a seguir; da etapa 4 na próxima subseção e da etapa 5 na seção 4.

Foi realizada uma comparação entre os seguintes chats inteligentes: ChatGPT, Google Bard, Dialogflow, IBM Watson e Amazon Lex. Foram listadas as vantagens percebidas e possíveis desafios ou limitações de cada chat com o propósito de oferecer uma referência ágil para orientar a seleção. Com base nisso, foi escolhido o ChatGPT devido à sua avançada compreensão da linguagem natural, capacidade de manter conversas naturais e fluídas em testes exploratórios, apresentar bons índices como gerador de código, bem como facilidade de uso e integração com AVA.

Quanto aos modelos de Design Instrucional foram considerados: ADDIE, SAM, Gagné's Nine, Dick e Carey, e 4C/ID. A análise objetivou oferecer uma visão equilibrada para orientar a escolha do modelo mais adequado às necessidades específicas do projeto, considerando diferentes abordagens e características. O modelo ADDIE foi escolhido devido a sua sistemática e estrutura, a disponibilidade de material e o detalhamento necessário para orientar o desenvolvimento do sistema, bem como a experiência prévia dos pesquisadores.

Quanto aos AVA foram considerados o Moodle, Canvas, Blackboard, Google Classroom, Edmodo e WordPress, por serem amplamente utilizados. O uso de extensões do WordPress para elaboração do AVA foi escolhida devido à flexibilidade, existência de vários plugins e ao fato de ser uma plataforma gratuita, possibilitando uma melhor customização para o desenvolvimento deste trabalho. Os parâmetros apresentados em [Maquiné 2020] foram considerados na escolha.

3.2. O sistema CodeChatPro

Primeiro, é apresentado o design e desenvolvimento do sistema CodeChatPro (subseção 3.2.1), na sequência, as principais funcionalidades do sistema são apresentadas (subseção 3.2.2)

3.2.1. Design e Desenvolvimento do Sistema CodeChatPro

Definiram-se os principais cenários de uso para alunos (acessar curso, realizar atividades, suporte, chat inteligente e visualizar conteúdo) e professores (criar curso, cadastrar atividade e cadastrar login). A partir da descrição das principais histórias de usuário e fluxos

de cenários atreladas às essas, o design e desenvolvimento do CodeChatPro foi guiado pelo ADDIE conforme destacado a seguir:

- *Análise*: Foi realizada análise das necessidades dos alunos (no contexto do estudo de caso desta investigação) nas lições de lógica de programação que revelou lacunas no entendimento dos conceitos fundamentais. Esse procedimento orientou a integração do ChatGPT, visando fornecer suporte e esclarecimento de dúvidas durante o curso. Identificou-se a necessidade de uma plataforma de fácil navegação. Essa análise serviu como base para guiar as próximas etapas do ADDIE.
- *Design*: Com base nas conclusões da análise, foi idealizada uma proposta de design da interação, enfatizando a colaboração do ChatGPT nas lições para otimizar a experiência educacional. A estrutura da página inicial da proposta de design foi planejada para integrar as interfaces de Cursos, ChatGPT e Dashboard.
- *Desenvolvimento*: Na fase de desenvolvimento, especialmente no componente de Cursos, buscou-se incorporar recursos do ChatGPT para fomentar uma colaboração ativa. O foco foi esclarecer dúvidas e orientar os alunos de maneira eficaz. Essa abordagem visou proporcionar suporte contínuo durante as atividades, conforme delineado no design.
- *Implementação*: Na etapa de implementação, buscou-se garantir a disponibilidade do AVA, assegurando uma colaboração contínua do ChatGPT nas lições de lógica de programação. O ChatGPT foi incorporado visando proporcionar suporte aos alunos de maneira integrada e rápida.
- *Avaliação*: Os alunos podem monitorar e avaliar seu próprio desempenho por meio de um dashboard com estatísticas do curso, oferecendo autonomia na gestão do aprendizado. Foi avaliada a integração do ChatGPT para fornecer feedback de dúvidas durante a realização dos exercícios, além disso, o ChatGPT foi considerado um canal de diálogo adicional ao professor para esclarecimento de dúvidas.

3.2.2. Funcionalidades do sistema CodeChatPro

O CodeChatPro está estruturado em cursos disponibilizados para exploração prévia e inscrição dos alunos. A Figura 1 apresenta a parte superior da página de um curso. Essa solução de design foi projetada para ser visualmente atraente, exibindo informações essenciais sobre cada curso, facilitando a escolha informada por parte do aluno. O assistente está sempre disponível ao aluno, por meio do ícone na parte inferior esquerda da página. Ao rolar a página, estão disponíveis as abas “informações sobre o curso”, “comentários” e “anúncios”. A aba “informações sobre o curso”, por exemplo, apresenta o conteúdo do curso, bem como as lições associadas a ele.

A Figura 2 apresenta a lição intitulada “Introdução à Lógica de Programação”; do lado esquerdo estão recursos atrelados à lição como acesso aos vídeos, documentos, exercícios e avaliações. No meio são apresentados conteúdos como a visão geral da lição e comentários (fórum) de discussão entre alunos e professores. Do lado direito, o chat inteligente provê feedback imediato segundo as perguntas formuladas pelos alunos e tema da lição apresentada.

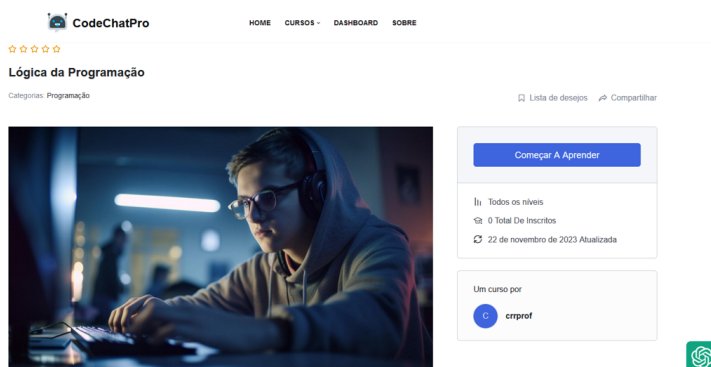


Figura 1. Exemplo de página de curso no sistema CodeChatPro

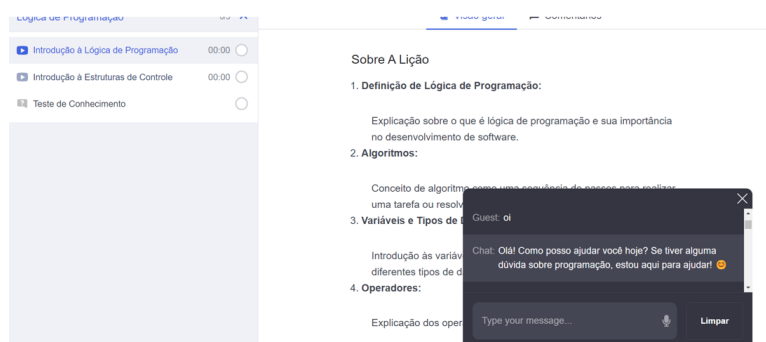


Figura 2. Exemplo do uso do ChatGPT na realização de exercícios

4. Avaliação

Esta seção apresenta a avaliação com alunos do ensino técnico incluindo detalhes sobre o local, participantes e metodologia de avaliação (subseção 4.1); os resultados (subseção 4.2); e a análise e discussão com base nos resultados obtidos (subseção 4.3).

4.1. Contexto, Participantes e Procedimentos da Avaliação

Esta pesquisa objetivou investigar e analisar a efetividade da abordagem desenvolvida e materializada no sistema CodeChatPro. No âmbito de curso técnico em informática, o foco é avaliar como essa tecnologia é aceita pelos estudantes, o apoio fornecido e as possíveis melhorias no desempenho em testes.

A pesquisa foi realizada no Senac Mogi Guaçu, com termos de consentimento assinados pelos participantes (todos maiores) e aprovados pela instituição. Participaram do estudo 20 alunos do curso Técnico em Informática, matriculados na unidade curricular de desenvolvimento de algoritmos, com idade de 18 a 50 anos. Os alunos foram organizados aleatoriamente em dois grupos, sendo 10 participantes do Grupo de Controle (GC) e 10 participantes do Grupo Experimental (GE). O seguinte procedimento foi adotado:

- *Pré-Teste:* foi conduzida avaliação preliminar com 10 perguntas básicas para determinar a habilidade prévia de todos os participantes (e novo sorteio caso necessário).
- *Instrumentos e Materiais:* incluem conteúdo planejado conforme a metodologia ADDIE, sistema CodeChatPro e acesso à internet. O conteúdo (que inclui exercícios) se refere à lógica de programação básica, tais como estruturas sequenciais, decisão e repetição.

- *Distribuição dos Exercícios:* Os grupos foram criados. O GC recebeu o conteúdo via email, enquanto o GE recebeu acesso ao CodeChatPro com o conteúdo já pré-cadastrado.
- *Execução:* Os participantes realizaram as atividades em dois dias com 4 aulas cada (total de 08 horas). Após o pré-teste, o conteúdo foi distribuído. O GC podia pedir apoio aos colegas, utilizar material didático e consultar a Web ou outros meios para tirar dúvidas. O GE poderia pedir apoio ao ChatGPT integrado ao sistema em caso de dúvidas. Após realizar a atividade, foi executado o pós-teste.
- *Pós-Teste:* conduzida uma nova avaliação com 10 perguntas mais avançadas para determinar o desempenho dos dois grupos no tópico abordado.
- *Métodos de Coleta de Dados:* Os métodos de coleta de dados incluíram formulários de avaliação quantitativa e qualitativa.
- *Local:* As atividades foram realizadas de maneira presencial em um laboratório com 24 computadores pessoais, em que 10 alunos de GC utilizaram o lado A e os alunos de GE utilizaram o lado B do laboratório.

Para o desenvolvimento do formulário foram utilizados conceitos do TAM (*Technology Acceptance Model*) [Silva et al. 2012]: utilidade percebida, facilidade de uso percebida, intenção comportamental e uso real do sistema. Mais detalhes sobre a execução da avaliação, conteúdo e testes podem ser obtidos em [Ribeiro 2024], bem como material suplementar incluindo, por exemplo, código fonte, exercícios e dados brutos podem ser obtidos com o primeiro autor.

4.2. Resultados

A Figura 3 apresenta que 60% dos alunos do GC tinha conhecimento prévio limitado e 40% tinha nível moderado de conhecimento. Enquanto para o GE 50% indicou conhecimento prévio limitado, 40% indicaram um nível moderado e 10% indicou conhecimento avançado.

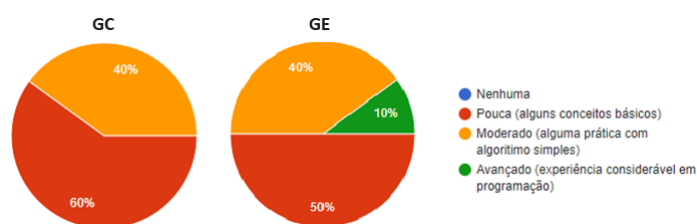


Figura 3. Nível de Conhecimento Prévio em Lógica de Programação.

A Figura 4 apresenta as respostas para questões 2 a 4 endereçadas ao GC. A questão 2 diz respeito onde eles buscaram ajuda em caso de dúvidas, sendo que 60% buscaram informações na web; 20% com colegas e 20% em livros. A questão 3 diz respeito à efetividade da ajuda obtida, sendo que 40% disseram que as dúvidas foram esclarecidas parcialmente, 30% disseram que não foram muito bem esclarecidas e apenas 30% tiveram as dúvidas completamente esclarecidas. Já a questão 4 considerou a expectativa do GC (que não usou o sistema) perante o uso do ChatGPT para auxílio às atividades, sendo que 70% achavam que ele poderia ajudar, enquanto 30% acham que existe a possibilidade de ajuda (talvez).

A Figura 5 apresenta as respostas para as questões 2 a 5 endereçada ao GE, baseadas no TAM. A questão 2 diz respeito à efetividade da ajuda obtida por meio do sistema,

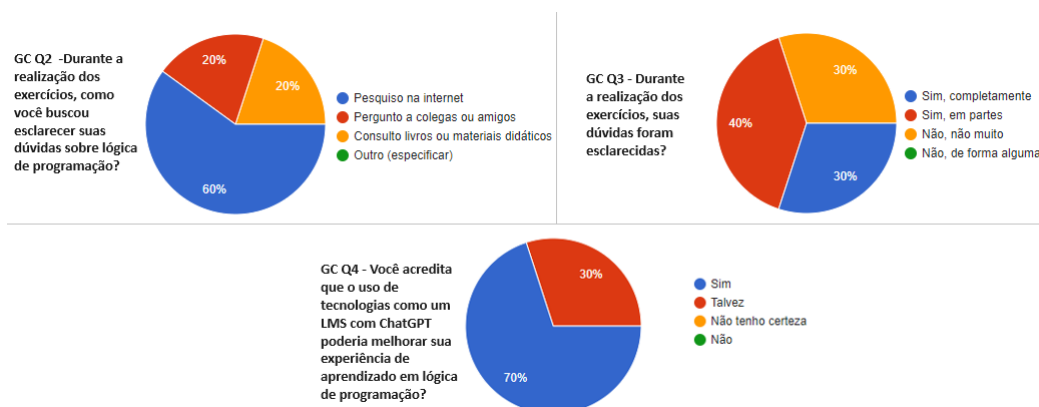


Figura 4. Respostas do grupo de controle às questões 2 a 4.

sendo que 60% foram completamente resolvidas, 30% em partes e apenas 10% disseram que não foram muito bem esclarecidas.

A questão 3 diz respeito à percepção dos participantes quanto à capacidade do sistema em melhorar o desempenho em atividades de lógica de programação, sendo que 60% disseram que houve melhoras e 30% disseram que essas melhoras foram parciais. A questão 4 considerou o quão intuitivo foi o uso do ChatGPT no sistema, sendo que (dos 9 que responderam) 77,8% considerou intuitivo e 22,2% considerou intuitivo “em partes”. A questão 5 diz respeito ao quão propenso estão os participantes a usar o ChatGPT durante as aulas de programação (após a experimentação), dos quais 90% responderam que usariam e 10% responderam que não.

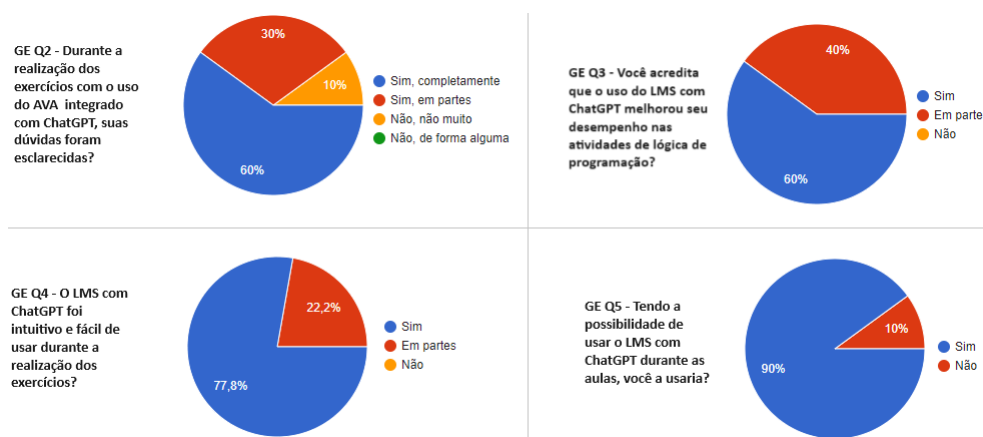


Figura 5. Respostas do grupo de experimentação às questões 2 a 5.

Os dois grupos responderam uma questão dissertativa sobre suas experiências nas atividades. As respostas são apresentadas na Figura 6.

Quanto aos testes com alunos, dadas as características e tamanho das amostras, optou-se por verificar a significância com o teste de postos sinalizados de Wilcoxon com nível de significância de 0,05 (ou seja, 95% de confiança) para as seguintes hipóteses: *Pré-Test*: H_{0pre} não há diferença de pontuação entre os participantes dos GC e GE no pré-teste; H_{apre} : há diferença de pontuação entre os participantes dos GC e GE no pré-teste. *Pós-Test*: H_{0pos} não há diferença de pontuação entre os participantes dos GC e GE

GC	GE
Você poderia compartilhar sua experiência com as ferramentas que usou.	Você poderia compartilhar sua experiência sobre o uso do LMS com ChatGPT?
5 respostas	8 respostas
A conexão instável ou lenta com a internet era uma fonte constante de frustração.	A experiência com o LMS com ChatGPT foi muito positiva para mim. Eu me senti mais apoiado e confiante.
A internet me proporcionou acesso a uma variedade de recursos de aprendizado	Às vezes eu sentia que suas respostas eram genéricas demais e não abordavam minhas dúvidas específicas.
Embora a internet ofereça acesso a uma vasta gama de tutoriais e recursos de aprendizado, muitas vezes eu me sentia perdido	Sempre que eu estava presa em um exercício, bastava digitar minha pergunta e obter uma resposta útil em segundos
A dependência da internet como minha principal fonte de aprendizado me deixava vulnerável a distrações online.	A experiência com o LMS com ChatGPT não foi tão boa quanto eu esperava
me senti sobrecarregado pela quantidade de conteúdo disponível,	A ferramenta tornou a aprendizagem mais interativa e envolvente
	Pude receber respostas instantâneas para minhas perguntas, o que acelerou meu aprendizado.
	tornou as atividades de lógica de programação mais dinâmicas
	Minha experiência com o LMS com ChatGPT foi incrível! A ferramenta tornou a aprendizagem mais interativa

Figura 6. Respostas do grupos GC e GE para questões dissertativas sobre a experiência com o uso de ferramentas de apoio.

no pós-teste; $H_{a_{pos}}$: há diferença de pontuação entre os participantes dos GC e GE no pós-teste.

Ao considerar os resultados do teste de Wilcoxon para o pré-teste, obteve-se $p - valor = 0.1573$, ou seja, $p > 0,05$. Portanto, aceita-se a hipótese nula $H_{0_{pre}}$ e conclui-se que não há diferença de pontuação entre os GC e GE para o pré-teste. Já ao considerar o pós-teste, obteve-se $p - valor = 0.009654$, ou seja, $p < 0,05$. Portanto, rejeita-se a hipótese nula $H_{0_{pos}}$ e conclui-se que há diferença de pontuação entre os GC e GE para o pós-teste. A Figura 7 apresenta diagramas Box Plot para pontuação do GC e GE no pré-teste, à esquerda, e para pontuação do GC e GE no pós-teste, à direita.

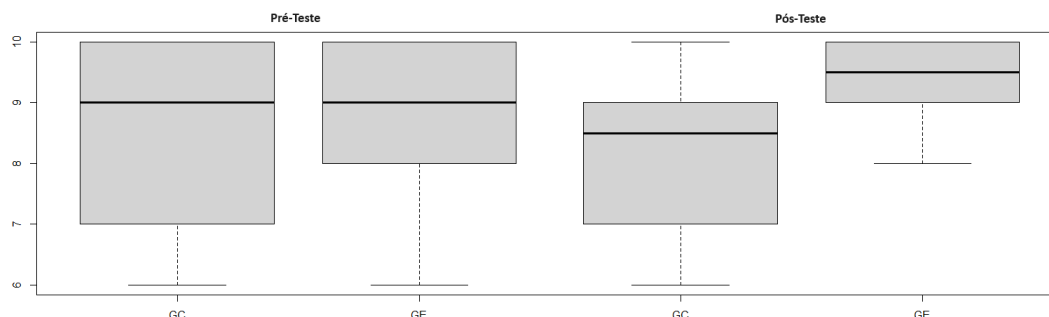


Figura 7. Boxplots com pontuação do GC e GE no pré-teste e pós-teste.

4.3. Discussão

Primeiramente, destaca-se que ao analisar a experiência no desenvolvimento do protótipo do sistema CodeChatPro que a tecnologia de chats inteligentes já está suficientemente madura e passível de integração com AVA, possibilitando apoio instantâneo aos alunos. Foi possível integrá-lo via ícone permanente no canto inferior esquerdo que aciona uma janela flutuante com resposta suficientemente rápida ao prompt inicial personalizado sobre o conteúdo apresentado ao aluno. É possível explorar outras maneiras de utilizar chats inteligentes no apoio tanto ao aluno e ao professor, tais como geração automática de exercícios, correção de códigos e geração de questões [Chico et al. 2024].

Ao analisar os resultados, houve um maior percentual de esclarecimento de dúvidas do grupo GE em relação ao GC, que utilizaram internet, amigos e materiais didático. Os participantes do GE destacaram que via ChatGPT foi possível melhorar o

desempenho e foi intuitivo utilizá-lo no sistema proposto. Destaca-se que o GC apresentou uma alta expectativa quanto ao uso do ChatGPT em conjunto com AVA no auxílio ao aprendizado de lógica de programação, assim como a propensão ao uso do ChatGPT com AVA durante a aula no GE também é grande.

Nas respostas às questões dissertativas (lado esquerdo da Figura 6), os participantes do GC destacaram a Web como fonte de acesso a uma vasta gama de recursos, mas também que essa quantidade pode ser uma fonte de distração; bem como o sentimento de estar sobrecarregado e a frustração com problemas de conexão. Já do lado direito da Figura 6, os participantes do GE destacaram positivamente as respostas instantâneas, deixando a aprendizagem mais dinâmica e interativa, enquanto outros alunos expressaram frustração perante as expectativas e a dificuldade em obter respostas para as dúvidas.

Os resultados da avaliação final apontaram que o GE acertou significativamente mais questões do que o GC. Vale ressaltar que este estudo é limitado em termos de quantidade de participantes (10), tempo de estudo (8H) e quantidades de questões (10), não resultando em conclusões definitivas e com validade estatística para toda população. Ressalta-se igualmente o perfil específico dos participantes e tema abordado, sendo necessários estudos adicionais para avaliar a proposta em outros temas, exercícios e perfis. Contudo, esse fato não invalida as lições aprendidas no estudo. Salienta-se que os resultados apresentaram indícios muito positivos da contribuição da abordagem, destacando que a tecnologia de Chats Inteligentes para o ensino de programação deve ser explorada e debatida por educadores e pesquisadores em computação.

5. Conclusão

O ensino de programação é uma tarefa desafiadora para professores e alunos, assim como a tecnologia está em constante evolução impondo novos desafios e oportunidades. A tecnologia de Chats Inteligentes evoluiu rapidamente nos últimos anos, sendo capaz de resolver problemas de programação complexos e tem potencial para auxiliar alunos e professores no ensino de programação.

Este artigo investigou o uso de Chat Inteligentes em uma abordagem que integra seu uso em AVA por meio do design instrucional. O sistema CodeChatPro explorou o uso do ChatGPT 3.5, integrado a um AVA construído por meio de extensão do WordPress em um processo de design baseado no ADDIE, para inclusão de cursos baseados em ADDIE. A solução foi avaliada experimentalmente com 20 alunos do curso Técnico do Senac Mogi Guaçu organizados em um grupo de controle e outro de experimentação. Resultados apontaram para uma boa aceitação e uso de Chats Inteligentes pelo GE quando comparados aos participantes do GC, que utilizaram outras alternativas para buscar respostas às dúvidas (ex: Web, colegas e material didático). Além disso os participantes do GE obtiveram melhores notas no pós-teste do que os participantes do GC.

Os próximos passos desta pesquisa incluem o aprimoramento da integração do Chat Inteligente com o sistema CodeChatPro de modo a perceber as dificuldades do aluno na execução dos exercícios e com isso prover auxílio mais direcionado quando solicitado, e personalizado se possível. Planeja-se investigar o uso de outros chats inteligentes (e novas versões do GPT) e a pesquisa por outros métodos de design instrucional. Destaca-se também a necessidade de estudos em larga escala e em longo prazo para investigar com maior profundidade a abordagem proposta.

Agradecimentos

Este trabalho foi desenvolvido no contexto do projeto ‘PIND/FAEPEX – Programa de Incentivo a Novos Docentes da Unicamp’ (#2560/23).

Referências

- Ai, R., Kohli, D., Maina, L., Manelski, E., Dhar, N. T., and Brzycki, C. (2020). Programming an educational chatbot to support virtual learning. In *2020 IEEE MIT Undergraduate Research Technology Conference (URTC)*, pages 1–5.
- Ashfaq, M. W., Tharewal, S., Iqbal, S., and Kayte, C. N. (2020). A review on techniques, characteristics and approaches of an intelligent tutoring chatbot system. In *2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC)*, pages 258–262.
- Banić, B., Konecki, M., and Konecki, M. (2023). Pair programming education aided by chatgpt. In *2023 46th MIPRO ICT and Electronics Convention (MIPRO)*, pages 911–915.
- Bispo Jr., E. L., Raabe, A., Matos, E., Maschio, E., Barbosa, E. F., Carvalho, L. G., Bittencourt, R. A., Duran, R. S., and Falcão, T. P. (2020). Tecnologias na educação em computação: Primeiros referenciais. *Revista Brasileira de Informática na Educação*, 28:509–527.
- Carreira, G., Silva, L., Mendes, A. J., and Oliveira, H. G. (2022). Pyo, a chatbot assistant for introductory programming students. In *2022 International Symposium on Computers in Education (SIIE)*, pages 1–6.
- Chico, V. J. S., Tessler, J. F., Bonacin, R., and Dos Reis, J. C. (2024). Bequizzer: Ai-based quiz automatic generation in the portuguese language. In *29th International Conference on Applications of Natural Language to Information Systems (NLDB 2024)*, page In Press.
- Finnie-Ansley, J., Denny, P., Becker, B. A., Luxton-Reilly, A., and Prather, J. (2022). The robots are coming: Exploring the implications of openai codex on introductory programming. In *Proceedings of the 24th Australasian Computing Education Conference, ACE '22*, page 10–19, New York, NY, USA. Association for Computing Machinery.
- Ganesan, M., C., D., B., H., A.S., K., and B., L. (2020). A survey on chatbots using artificial intelligence. In *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, pages 1–5.
- Henno, J., Jaakkola, H., and Mäkelä, J. (2023). What do we know about learning – conversations with chatgpt. In *2023 46th MIPRO ICT and Electronics Convention (MIPRO)*, pages 572–577.
- Ji, S. and Yuan, T. (2022). Conversational intelligent tutoring systems for online learning: What do students and tutors say? In *2022 IEEE Global Engineering Education Conference (EDUCON)*, pages 292–298.
- Junior, S. S., Freitas, R., Morais, M., and Costa, D. (2023). Chatgpt no auxílio da aprendizagem de programação: Um estudo de caso. In *Anais do XXXIV Simpósio Brasileiro de Informática na Educação*, pages 1375–1384, Porto Alegre, RS, Brasil. SBC.

- Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26.
- Lakshmi, S. (2020). A study on machine learning based conversational agents and designing techniques. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pages 965–968.
- Lin, Y.-H. (2022). Chatbot script design for programming language learning. In *2022 IEEE 5th Eurasian Conference on Educational Innovation (ECEI)*, pages 123–125.
- Lin, Y.-H. and Tsai, T. (2019). A conversational assistant on mobile devices for primitive learners of computer programming. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, pages 1–4.
- Luo, C. J. and Gonda, D. E. (2019). Code free bot: An easy way to jumpstart your chatbot! In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, pages 1–3.
- Maquiné, G. (2020). Recursos para avaliação da aprendizagem: estudo comparativo entre ambientes virtuais de aprendizagem. In *Anais do XXVI Workshop de Informática na Escola*, pages 299–308, Porto Alegre, RS, Brasil. SBC.
- Martenstyaro, R. and Rosmansyah, Y. (2015). A framework for designing survey training based on 3d virtual learning environment using sloodle. In *2015 International Conference on Information Technology Systems and Innovation (ICITSI)*, pages 1–6.
- Murad, D. F., Irsan, M., Akhirianto, P. M., Fernando, E., Murad, S. A., and Hendra Wijaya, M. (2019). Learning support system using chatbot in "kejar c package" homeschooling program. In *2019 International Conference on Information and Communications Technology (ICOIACT)*, pages 32–37.
- OpenAI (2024). Gpt-4 technical report. <https://arxiv.org/abs/2303.08774>.
- Ribeiro, C. R. (2024). Uma abordagem para a aprendizagem de programação baseada em chats inteligentes. Mestrado em ciência da computação, Centro Universitário Campo Limpo Paulista, Campo Limpo Paulista.
- Silva, P., Pimentel, V., and Soares, J. (2012). A utilização do computador na educação: aplicando o technology acceptance model (tam). *Biblionline, João Pessoa*, 8(esp):263–272.
- Sophia, J. and Jacob, T. (2021). Edubot-a chatbot for education in covid-19 pandemic and vqabot comparison. In *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 1707–1714.
- Trilaksono, K. and Santoso, H. B. (2017). Moodle based learning management system development for kinesthetic learning style. In *2017 7th World Engineering Education Forum (WEEF)*, pages 602–606.
- Wang, J., Tang, Y., Hare, R., and Wang, F. (2024). Parallel intelligent education with chatgpt. *Frontiers of Information Technology and Electronic Engineering*, 25(1):12–18.