

UNIVERSIDADE FEDERAL DO PARANÁ

LUCAS MÜLLER

BOT EDUCACIONAL PARA METODOLOGIAS ATIVAS NO ENSINO REMOTO

CURITIBA PR

2025

LUCAS MÜLLER

BOT EDUCACIONAL PARA METODOLOGIAS ATIVAS NO ENSINO REMOTO

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Computação*.

Orientador: Bruno Müller Junior.

CURITIBA PR

2025

Ficha catalográfica

Substituir o arquivo `0-iniciais/catalografica.pdf` pela ficha catalográfica fornecida pela Biblioteca da UFPR (PDF em formato A4).

Instruções para obter a ficha catalográfica e fazer o depósito legal da tese/dissertação (contribuição de André Hochuli, abril 2019. Links atualizados Wellton Costa, Nov 2022):

1. Estas instruções se aplicam a dissertações de mestrado e teses de doutorado. Trabalhos de conclusão de curso de graduação e textos de qualificação não precisam segui-las.
2. Verificar se está usando a versão mais recente do modelo do PPGInf e atualizar, se for necessário (<https://gitlab.c3sl.ufpr.br/maziero/tese>).
3. conferir o *checklist* de formato do Sistema de Bibliotecas da UFPR, em <https://bibliotecas.ufpr.br/servicos/normalizacao/>
4. Enviar e-mail para "referencia.bct@ufpr.br" com o arquivo PDF da dissertação/tese, solicitando a respectiva ficha catalográfica.
5. Ao receber a ficha, inseri-la em seu documento (substituir o arquivo `0-iniciais/catalografica.pdf` do diretório do modelo).
6. Emitir a Certidão Negativa (CND) de débito junto a biblioteca, em <https://bibliotecas.ufpr.br/servicos/certidao-negativa/>
7. Avisar a secretaria do PPGInf que você está pronto para o depósito. Eles irão mudar sua titulação no SIGA, o que irá liberar uma opção no SIGA pra você fazer o depósito legal.
8. Acesse o SIGA (<http://www.prppg.ufpr.br/siga>) e preencha com cuidado os dados solicitados para o depósito da tese.
9. Aguarde a confirmação da Biblioteca.
10. Após a aprovação do pedido, informe a secretaria do PPGInf que a dissertação/tese foi depositada pela biblioteca. Será então liberado no SIGA um link para a confirmação dos dados para a emissão do diploma.

Ficha de aprovação

Substituir o arquivo 0-iniciais/aprovacao.pdf pela ficha de aprovação fornecida pela secretaria do programa, em formato PDF A4.

*"I see now that the circumstances
of one's birth are irrelevant. It is
what you do with the gift of life that
determines who you are."*

— *Mewtwo*

ACKNOWLEDGEMENTS

Inserir os agradecimentos. Os agradecimentos devem ocupar no máximo uma página, devem ser justificados na largura da página e com um afastamento de parágrafo na primeira linha de 1,27 cm. O espaçamento entre linhas deve ser de 1,5 linhas. Não deve haver espaçamento adicional entre parágrafos.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

RESUMO

Esta monografia investiga o uso de *bots* como facilitadores de interações naturais em ambientes virtuais de ensino, visando viabilizar metodologias ativas no contexto remoto. O estudo é embasado da premissa de que a ausência de interações presenciais pode comprometer a eficácia de abordagens pedagógicas centradas no aluno. É explorado a definição e componentes de *bots*, suas aplicações educacionais e destaca três princípios fundamentais para interações mediadas: comunicação multidirecional, engajamento ativo e adaptação contextual. O trabalho apresenta o desenvolvimento de um *bot* educacional integrado ao Discord, plataforma escolhida por suas características que emulam um ambiente interativo, incluindo um *dashboard* exclusivo para professores que permite controle pedagógico não-invasivo e implementa recursos como *feedback* em tempo real, atividades colaborativas e ferramentas para aprendizagem baseada em problemas. A implementação técnica utiliza a biblioteca Concord em C, desenvolvida pelo autor, com arquitetura modular que gerencia publicação de conteúdo, interações, análise de dados e persistência. A prova de conceito demonstra funcionalidades como publicação estruturada, mecanismos de *feedback* rápido e coleta anônima de dúvidas. É proposto uma metodologia de avaliação baseada em questionários, logs automáticos e entrevistas para mensurar engajamento, impacto pedagógico, usabilidade e viabilidade de implementação de metodologias ativas. Se conclui que *bots* educacionais podem efetivamente aproximar o ambiente virtual da espontaneidade das interações presenciais, elemento fundamental para o sucesso das metodologias ativas no ensino remoto.

Palavras-chave: *Bots* educacionais. Metodologias ativas. Ensino remoto. Interações mediadas. *Feedback* em tempo real.

ABSTRACT

This undergraduate thesis investigates the use of bots as facilitators of natural interactions in virtual learning environments, aiming to enable active methodologies in remote teaching contexts. The study is based on the premise that the absence of face-to-face interactions can compromise the effectiveness of student-centered pedagogical approaches. It explores the definition and components of bots, their educational applications, and highlights three fundamental principles for mediated interactions: multidirectional communication, active engagement, and contextual adaptation. The work presents the development of an educational bot integrated with Discord, a platform chosen for its features that emulate an interactive environment, including an exclusive dashboard for teachers that allows non-invasive pedagogical control and implements resources such as real-time feedback, collaborative activities, and tools for problem-based learning. The technical implementation uses the Concord library in C, developed by the author, with a modular architecture that manages content publication, interactions, data analysis, and persistence. The proof of concept demonstrates functionalities such as structured publication, quick feedback mechanisms, and anonymous question collection. It is proposed an evaluation methodology based on questionnaires, automatic logs, and interviews to measure engagement, pedagogical impact, usability, and feasibility of implementing active methodologies. It is concluded that educational bots can effectively bring the virtual environment closer to the spontaneity of face-to-face interactions, a fundamental element for the success of active methodologies in remote teaching.

Keywords: Educational bots. Active methodologies. Remote teaching. Mediated interactions. Real-time feedback.

LIST OF FIGURES

2.1	Arquitetura de um <i>chatbot</i> segundo Huang et al. [1].	15
2.2	Relação entre o <i>bot</i> educacional e o <i>dashboard</i> de controle pedagógico.	19
3.1	Fluxo de interações no ambiente educacional virtual mediado pelo <i>bot</i> . A figura mostra um diagrama com o professor à esquerda, o <i>dashboard</i> do professor como interface de controle, o <i>bot</i> educacional ao centro, e os alunos à direita, ilustrando os fluxos de comunicação: (a) professor controlando a aula via <i>dashboard</i> , (b) <i>dashboard</i> enviando comandos ao <i>bot</i> , (c) <i>bot</i> processando e disponibilizando o material para os alunos, (d) alunos interagindo com o conteúdo, (e) <i>bot</i> coletando <i>feedback</i> dos alunos, e (f) <i>bot</i> fornecendo análises em tempo real ao professor através do <i>dashboard</i>	22

LIST OF TABLES

2.1	Classificação de <i>bots</i> segundo propósito e funcionalidade	16
4.1	Métricas para avaliação da eficácia do <i>bot</i>	33

LIST OF ACRONYMS

UI	<i>User Interface</i>
GUI	<i>Graphical User Interface</i>
NLU	<i>Natural Language Understanding</i>
DM	<i>Dialogue Manager</i>
RG	<i>Response Generation</i>
API	<i>Application Programming Interface</i>
REST	<i>Representational State Transfer</i>
IHC	Interação Humano-Computador
MOOCs	<i>Massive Open Online Courses</i>
CRM	<i>Customer Relationship Management</i>
DINF	Departamento de Informática
UFPR	Universidade Federal do Paraná

CONTENTS

1	INTRODUÇÃO	12
2	REVISÃO BIBLIOGRÁFICA	14
2.1	DEFINIÇÃO E COMPONENTES DE <i>BOTS</i>	14
2.2	CLASSIFICAÇÕES DE <i>BOTS</i>	15
2.3	<i>BOTS</i> NO CONTEXTO EDUCACIONAL	16
2.3.1	Desafios do Ensino Remoto	17
2.3.2	Interação Humano-Computador na Educação	17
2.3.3	Princípios para Interação Mediada por <i>Bots</i> na Educação	17
2.3.4	<i>Dashboards</i> como Ferramenta de Controle Pedagógico	18
2.4	FERRAMENTAS PARA DESENVOLVIMENTO DE <i>BOTS</i> NO DISCORD	19
2.5	TRABALHOS RELACIONADOS	20
2.6	OBJETIVOS DO TRABALHO	20
3	<i>BOT</i> EDUCACIONAL PARA METODOLOGIAS ATIVAS EM AMBIENTES VIRTUAIS	22
3.1	VISÃO CONCEITUAL DA APLICAÇÃO	22
3.2	INTEGRAÇÃO COM O AMBIENTE EDUCACIONAL	22
3.2.1	<i>Dashboard</i> do Professor	23
3.3	RECURSOS PARA PROMOÇÃO DE METODOLOGIAS ATIVAS	23
3.3.1	<i>Feedback</i> em Tempo Real	23
3.3.2	Aprendizagem Baseada em Problemas	24
3.4	EXEMPLO PRÁTICO: AULA DE COMANDOS DE REPETIÇÃO	24
3.4.1	Preparação da Aula	24
3.4.2	Interação Durante a Aula	25
4	PROVA DE CONCEITO: <i>BOT</i> PARA INTERAÇÃO EDUCACIONAL	30
4.1	CONTEXTO DA INTERAÇÃO PROFESSOR-ALUNO EM AMBIENTES REMOTOS	30
4.2	IMPLEMENTAÇÃO TÉCNICA	30
4.2.1	Arquitetura do <i>Bot</i>	30
4.2.2	Implementação do <i>Dashboard</i>	31
4.2.3	Integração Técnica	31
4.3	FUNCIONALIDADES IMPLEMENTADAS	32
4.3.1	Funcionalidades do <i>Bot</i> no Discord	32
4.3.2	Funcionalidades do <i>Dashboard</i> do Professor	32

4.4	METODOLOGIA DE AVALIAÇÃO	33
4.4.1	Ambiente e Participantes	33
4.4.2	Coleta de Dados.	33
4.4.3	Métricas de Avaliação.	33
4.5	RESULTADOS ILUSTRATIVOS	34
4.5.1	Dados Quantitativos.	34
4.5.2	Análise Qualitativa	34
5	CONCLUSÃO	35
5.1	SÍNTESE DOS RESULTADOS	35
5.2	LIMITAÇÕES DO ESTUDO.	35
5.3	CONTRIBUIÇÕES DO TRABALHO	36
5.4	TRABALHOS FUTUROS	36
	REFERENCES	37

1 INTRODUÇÃO

O ensino remoto tem se consolidado como uma alternativa viável para a disseminação do conhecimento, especialmente em cenários que exigem distanciamento social [2]. No entanto, essa modalidade apresenta desafios significativos, como a manutenção do engajamento dos alunos e a efetividade da comunicação entre docentes e discentes. A ausência de interações presenciais pode levar a uma experiência educacional menos dinâmica e participativa, distanciando as práticas pedagógicas de uma comunicação natural e espontânea [2].

As metodologias ativas [3] de aprendizagem representam uma abordagem educacional que coloca o aluno como protagonista do processo de aprendizagem, em contraste com o ensino tradicional onde o estudante assume um papel predominantemente passivo. Estas metodologias envolvem participação direta, reflexão contínua e engajamento prático do aluno na construção do conhecimento. No contexto presencial, técnicas como aprendizagem baseada em problemas [4], sala de aula invertida [5] e aprendizagem colaborativa [6] já demonstraram resultados positivos. No entanto, sua aplicação em ambientes remotos permanece um desafio significativo devido às limitações de interação natural entre os participantes [2].

A integração de tecnologias interativas no ambiente educacional virtual emerge como elemento essencial para superar os obstáculos do ensino remoto e viabilizar metodologias ativas neste contexto.

Dentre essas ferramentas capazes de recriar os elementos de metodologias ativas, estão os *bots*. Os *bots*, programas de computador capazes de simular interações humanas de forma automatizada e personalizada, apresentam-se como ferramentas capazes de recriar elementos de naturalidade na comunicação digital, aproximando o ambiente virtual da espontaneidade característica das interações presenciais [7].

Este trabalho tem como objetivo investigar o uso de *bots* como facilitadores de interações naturais em ambientes virtuais, o que poderia viabilizar a aplicação de metodologias ativas no ensino remoto. A pesquisa parte da observação que os *bots* podem atuar como pontes tecnológicas que diminuem a distância comunicativa entre participantes em ambientes virtuais, promovendo um fluxo mais natural e espontâneo de interações entre professores e alunos, sem que isso represente uma sobrecarga adicional para os docentes.

Para alcançar este objetivo, foi desenvolvido um *bot* integrado ao Discord, uma plataforma de comunicação virtual que, embora não seja tradicionalmente educacional, foi escolhida por oferecer recursos que emulam eficientemente um ambiente de ensino interativo. A plataforma suporta videoconferência, *chat* simultâneo, compartilhamento de conteúdo e criação de enquetes, proporcionando um ecossistema digital onde interações naturais podem ser facilitadas por um *bot* que intermedia as interações entre professor e aluno.

A pesquisa busca analisar como este *bot* pode criar um ambiente virtual onde interações naturais são facilitadas, tornando viável a aplicação de princípios de ensino ativo mesmo à distância. O estudo examina especificamente como esta ferramenta pode transformar a natureza das comunicações digitais educacionais, aproximando-as da fluidez e espontaneidade das interações presenciais, elementos fundamentais para o sucesso das metodologias ativas.

A eficácia da ferramenta é avaliada em três dimensões principais: o aumento do engajamento espontâneo dos alunos durante as aulas remotas, a fluidez e naturalidade da comunicação entre docentes e discentes mediada pelo *bot*, e a receptividade dos usuários quanto à integração dessa tecnologia como elemento natural do processo educacional.

O efeito do uso do *bot* proposto foi avaliado através da coleta de dados via preenchimento de questionários enviados aos participantes, e de informações extraídas pelo *bot* em sua mediação das interações básicas durante a experiência prática.

Este trabalho está organizado da seguinte forma: o Capítulo 2 apresenta a revisão bibliográfica, abordando conceitos fundamentais sobre *bots*, interações naturais e metodologias ativas em ambientes virtuais; o Capítulo 3 descreve a concepção e implementação do *bot* educacional proposto, com foco nas funcionalidades que promovem interações naturais; o Capítulo 4 detalha a prova de conceito e analisa os resultados obtidos no ambiente educacional remoto; e o Capítulo 5 apresenta a conclusão com análise dos resultados, limitações do estudo e sugestões de trabalhos futuros.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta uma revisão da literatura sobre *bots*, seu uso em contextos educacionais e as tecnologias relacionadas ao seu desenvolvimento, estabelecendo as bases conceituais para este trabalho.

O Capítulo está organizado da seguinte forma: a Seção 2.1 apresenta a definição e os componentes de *bots*, a Seção 2.2 discute as classificações de *bots*, a Seção 2.3 aborda os *bots* no contexto educacional, incluindo os desafios do ensino remoto e aspectos de interação humano-computador na educação, a Seção 2.4 analisa bibliotecas e tecnologias para desenvolvimento de *bots*, a Seção 2.5 apresenta trabalhos relacionados, e finalmente, a Seção 2.6 delinea os objetivos específicos deste trabalho com base nos conceitos apresentados.

2.1 DEFINIÇÃO E COMPONENTES DE *BOTS*

Bots são programas automatizados projetados para interagir com usuários ou sistemas, realizando tarefas específicas com diferentes níveis de autonomia. *Bots* podem ser definidos como "aplicações que combinam uma interface conversacional com a capacidade de executar tarefas específicas para o usuário" [8], ou também "uma aplicação que realiza certas tarefas repetitivas de maneira mais rápida que o ser humano" [9].

A arquitetura de um *bot* é descrita de diferentes maneiras na literatura, mas geralmente envolve um conjunto de componentes essenciais que trabalham juntos para processar a entrada do usuário, gerenciar o diálogo e gerar respostas apropriadas. Uma estrutura comum inclui como elementos principais (1) interface do usuário, (2) compreensão de linguagem natural, (3) gerenciador de diálogo, (4) integração com backend, e por fim (5) geração de linguagem natural [1]:

1. **Interface do Usuário (*User Interface - UI*):** Este é o ponto de contato entre o usuário e o *bot*. A *UI* é responsável por receber a entrada do usuário (texto, voz, cliques em elementos gráficos) e apresentar as respostas do *bot* de forma compreensível. Pode variar desde simples janelas de *chat* baseadas em texto até interfaces de voz sofisticadas ou *GUIs* interativas.
2. **Compreensão de Linguagem Natural (*Natural Language Understanding - NLU*):** Este componente é crucial para interpretar a entrada do usuário em linguagem natural. Ele analisa o texto ou a fala para identificar a intenção do usuário (o que o usuário quer fazer) e extrair informações relevantes, conhecidas como entidades (por exemplo, datas, locais, nomes). O *NLU* transforma a entrada não estruturada do usuário em dados estruturados que o *bot* pode processar.
3. **Gerenciador de Diálogo (*Dialogue Manager - DM*):** O *DM* mantém o estado do diálogo (o contexto da conversa), rastreia o histórico de interações e decide qual ação tomar a seguir com base na intenção identificada pelo *NLU* e nas regras de negócio ou na lógica conversacional definida. Isso pode envolver fazer perguntas de esclarecimento, acessar a base de conhecimento, chamar uma *API* externa ou gerar uma resposta. O *DM* pode ser implementado usando abordagens baseadas em regras ou modelos de aprendizado de máquina mais complexos.

4. **Integração com Backend e Base de Conhecimento (*Backend Integration & Knowledge Base*):** Para realizar tarefas úteis e fornecer informações precisas, os *chatbots* frequentemente precisam interagir com sistemas externos e acessar dados. A integração com o *backend* permite que o *bot* se conecte a *APIs*, bancos de dados, sistemas de *CRM*, ou outras fontes de informação e serviços. Essa base de conhecimento pode incluir conhecimento estático (pré-programado), conhecimento dinâmico (acessado em tempo real via *APIs*), conhecimento contextual (histórico do usuário) e até conhecimento colaborativo (gerado pelo usuário). É importante a integração com o *backend* para acessar essas informações e executar ações.
5. **Geração de Resposta (*Response Generation - RG*):** Uma vez que o Gerenciador de Diálogo decide a resposta a ser dada, o componente *RG* a transforma em linguagem natural (texto ou fala) para ser apresentada ao usuário através da Interface do Usuário. A complexidade do *RG* pode variar desde o uso de modelos de resposta pré-definidos até a geração dinâmica de sentenças complexas.

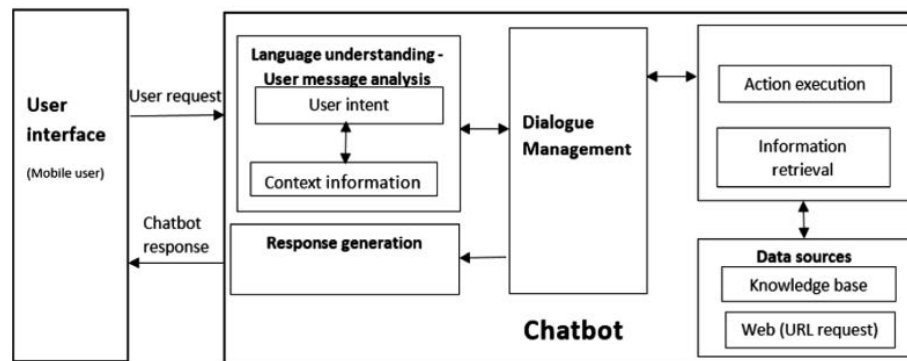


Figure 2.1: Arquitetura de um *chatbot* segundo Huang et al. [1].

2.2 CLASSIFICAÇÕES DE BOTS

Existem diversas formas de classificar *bots*, dependendo de suas características, funcionalidades e aplicações. Os *bots* podem ser classificados de acordo com seu propósito principal [10], ou de acordo com a sua funcionalidade primária [11].

Propósito [10]	Descrição
<i>Bots</i> generalistas	Executam uma ampla variedade de tarefas, como responder perguntas gerais e executar comandos simples.
<i>Bots</i> transacionais	Realizam transações com sistemas externos, como <i>bots</i> bancários ou de compras.
<i>Bots</i> informacionais	Fornecem informações e respondem perguntas específicas dos usuários.
<i>Bots</i> de produtividade	Automatizam tarefas repetitivas, como lembretes ou agendamentos.
<i>Bots</i> de colaboração	Facilitam a interação e colaboração entre usuários, geralmente em ambientes de comunicação.
Funcionalidade [11]	Descrição
Tarefas administrativas	Auxiliam na organização de atividades, como agendar reuniões e gerenciar compromissos.
Entretenimento	Proporcionam atividades lúdicas, como jogos ou interações divertidas.
Funcionalidade e qualidade	Aumentam a eficiência de serviços, como <i>bots</i> de suporte técnico ou de coleta de <i>feedback</i> .
Comunidade	Moderam e gerenciam interações em comunidades <i>online</i> .
Arquivadores	Organizam e recuperam informações, como documentos e registros de mensagens.

Table 2.1: Classificação de *bots* segundo propósito e funcionalidade

Este trabalho concentra-se em *bots* do tipo "produtividade" e "colaboração" como propósito, e "funcionalidade e qualidade" como funcionalidade. O foco é a sua aplicação dentro do contexto educacional, onde eles podem atuar como assistentes virtuais que facilitam a interação entre alunos e professores, a fim de promover um ambiente remoto de aprendizagem mais dinâmico e interativo.

2.3 BOTS NO CONTEXTO EDUCACIONAL

Na educação, os *bots* têm sido utilizados para diversos propósitos, desde fornecer suporte administrativo até oferecer experiências de aprendizado personalizadas. Os *bots* educacionais podem transformar a experiência de aprendizagem ao oferecer suporte contínuo e personalizado que seria impraticável para um professor humano fornecer a todos os alunos simultaneamente [12].

Estudos recentes têm explorado aplicações educacionais específicas de *bots*. Por exemplo, o uso de *chatbots* para melhorar a retenção de conhecimento em estudantes universitários [7], e aumentar o engajamento em cursos *online* abertos e massivos (*MOOCs*) [13].

De uma maneira geral, *bots* educacionais são particularmente eficazes quando (1) fornecem *feedback* imediato aos alunos, (2) oferecem disponibilidade contínua para assistência, (3) personalizam a experiência de aprendizado, (4) reduzem a carga cognitiva dos instrutores e (5) permitem que os instrutores se concentrem em atividades pedagógicas e interativas [14].

A seguir, exploramos quatro dimensões importantes relacionadas ao uso de *bots* educacionais: a Seção 2.3.1 apresenta os desafios específicos do ensino remoto que podem ser mitigados por essas ferramentas, a Seção 2.3.2 discute os princípios de interação humano-

computador relevantes para o *design* de *bots* educacionais eficazes, a Seção 2.3.3 delinea os princípios fundamentais para a interação mediada por *bots* na educação, e finalmente, a Seção 2.3.4 analisa o papel dos *dashboards* como ferramentas de controle pedagógico.

2.3.1 Desafios do Ensino Remoto

O ensino remoto apresenta desafios únicos que podem ser parcialmente mitigados pelo uso de tecnologias interativas como *bots*. Aqui se faz distinção entre "ensino remoto emergencial" e educação *online* planejada, destacando que muitas instituições foram forçadas a adotar o primeiro modelo durante a pandemia de COVID-19, sem tempo adequado para planejamento [15][2].

Entre os principais desafios identificados estão [2]:

- Limitações tecnológicas e acesso desigual
- Competências digitais insuficientes de professores e alunos
- Falta de estrutura para avaliação eficaz
- Dificuldade em manter o engajamento dos alunos
- Ausência de interação social e senso de comunidade

Tecnologias como *bots* podem preencher algumas dessas lacunas ao proporcionar uma interface natural e contínua entre os participantes do processo educacional, oferecendo um canal adicional de comunicação e suporte tanto para alunos quanto para professores [16].

2.3.2 Interação Humano-Computador na Educação

Estudos em IHC destacam a importância de sistemas que se ajustem ao comportamento e às necessidades dos usuários [17]. Na educação, isso implica em promover interfaces que permitam participação ativa, acessibilidade e adaptabilidade aos estilos de aprendizagem dos alunos.

Norman [18] enfatiza o conceito de *design* centrado no usuário, onde a tecnologia deve se adaptar às necessidades humanas e não o contrário. Aplicado ao contexto educacional, este princípio sugere que os *bots* devem ser projetados considerando as necessidades pedagógicas específicas e as limitações cognitivas dos alunos.

Bots educacionais se encaixam nesse contexto por serem acessíveis e flexíveis na forma de interação. Interfaces conversacionais podem reduzir a carga cognitiva associada à navegação em sistemas educacionais complexos, permitindo que os alunos se concentrem no conteúdo do aprendizado em vez de na interface [19].

2.3.3 Princípios para Interação Mediada por *Bots* na Educação

Com base na literatura sobre *bots* educacionais e metodologias ativas, surgem três princípios fundamentais emergem como pilares para o *design* de interações eficazes mediadas por *bots* em ambientes educacionais:

1. **Comunicação multidirecional:** Um *bot* educacional eficaz não deve apenas transmitir informações do professor para os alunos, mas também facilitar o retorno dos alunos para o professor, criando um ciclo contínuo de *feedback*. Esse princípio alinha-se com a concepção de aprendizagem dialógica [20], onde o conhecimento é construído através da interação bidirecional entre educador e educando.

2. **Engajamento ativo:** Através de mecânicas interativas, o *bot* deve estimular constantemente a participação dos alunos, transformando-os de receptores passivos a agentes ativos no processo de aprendizagem. Este princípio está fundamentado nas teorias construtivistas de aprendizagem [21], que enfatizam a importância da experiência prática e da participação na construção do conhecimento.
3. **Adaptação contextual:** O sistema deve se ajustar ao ritmo da aula e às necessidades específicas da disciplina, oferecendo diferentes modos de interação conforme o momento pedagógico. Tecnologias educacionais eficazes devem ser flexíveis o suficiente para se adaptarem a diferentes contextos pedagógicos e estilos de aprendizagem [16].

Estes princípios fornecem uma base teórica para o *design* de *bots* educacionais que efetivamente aprimoram o processo de aprendizagem, especialmente em contextos de metodologias ativas onde a participação e o engajamento dos alunos são essenciais.

2.3.4 *Dashboards* como Ferramenta de Controle Pedagógico

Um elemento crucial no *design* de *bots* educacionais é a interface de controle que permite aos educadores gerenciar o fluxo das interações. Os *dashboards* pedagógicos surgem como uma solução para esta necessidade, oferecendo uma visão consolidada das atividades e permitindo intervenções em tempo real sem interromper o fluxo da aula [22].

Dashboards de aprendizagem são "*displays* únicos que agregam diferentes indicadores sobre aprendiz, atividades de aprendizagem e/ou contexto de aprendizagem em uma ou múltiplas visualizações" [22]. No contexto de *bots* educacionais, estes *dashboards* evoluem para se tornarem não apenas ferramentas de visualização, mas interfaces de comando que permitem aos professores:

1. **Orquestrar atividades:** Iniciar e controlar sequências de aprendizagem sem necessidade de inserir comandos em *chats* públicos
2. **Monitorar em tempo real:** Visualizar métricas de engajamento e compreensão durante a aula
3. **Receber alertas:** Ser notificado sobre padrões que exijam intervenção pedagógica
4. **Personalizar interações:** Adaptar atividades com base nas necessidades observadas
5. **Analisar resultados:** Obter relatórios detalhados após as sessões

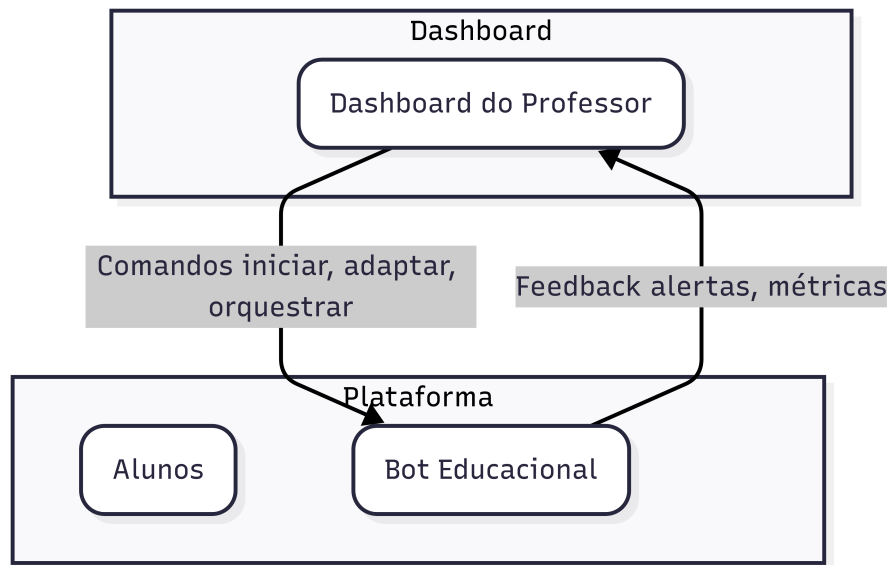


Figure 2.2: Relação entre o *bot* educacional e o *dashboard* de controle pedagógico.

Esta abordagem separa claramente o canal de comando (*dashboard*, visível apenas para o professor) do canal de interação (plataforma de comunicação, visível para todos os participantes), seguindo o princípio de "separação de interesses" como essencial para ambientes de aprendizagem tecnologicamente mediados. A eficácia desse modelo de interação será avaliada em experimentos controlados, onde participantes assumirão papéis de professor e alunos, interagindo em um ambiente de sala de aula simulado.

2.4 FERRAMENTAS PARA DESENVOLVIMENTO DE *BOTS* NO DISCORD

Existem várias plataformas que permitem desenvolver *bots* educacionais, como por exemplo, no Moodle, que é uma plataforma de gestão de aprendizagem amplamente utilizada em ambientes educacionais [23], e também no Slack, que é uma plataforma de comunicação corporativa que também pode ser utilizada para fins educacionais [24]. Dentre destas escolhemos o Discord, uma plataforma de comunicação digital que apesar de ter sido originalmente desenvolvida para comunidades de jogos, apresenta as seguintes características que a tornam adequada para emular um ambiente educacional remoto [25]:

- **Comunicação em tempo real:** Permite interações síncronas e assíncronas
- **Canais temáticos:** Permitem organizar discussões por tópicos específicos
- **Transmissão de voz/vídeo:** Facilita aulas síncronas com interação audiovisual
- **Compartilhamento de tela:** Possibilita demonstrações práticas pelo professor
- **Sistema de reações:** Oferece mecanismo não-verbal para expressão de compreensão ou dúvidas
- **Persistência de mensagens:** Mantém o histórico de interações disponível para consulta posterior

Sendo assim, destacamos algumas bibliotecas específicas para desenvolvimento de *bots* na plataforma do Discord, cada uma com suas particularidades e casos de uso apropriados, (1) JavaScript/Node.js, (2) Python e (3) C.

1. **Discord.js**[26]: Uma biblioteca JavaScript/Node.js que oferece abstração de alto nível para interação com a *API* do Discord. É rica em recursos e possui uma comunidade ativa, sendo adequada para desenvolvedores que preferem desenvolvimento rápido.
2. **Discord.py**[27]: Equivalente ao Discord.js, mas para a linguagem Python. Oferece abstrações semelhantes e é amplamente utilizada para desenvolvimento de *bots* no Discord.
3. **Concord**[28]: Uma biblioteca em C que fornece acesso de baixo nível à *API* do Discord, desenvolvida pelo autor deste trabalho durante o período da pandemia de COVID-19, como um projeto *hobby*. Diferentemente das opções anteriores, a Concord prioriza desempenho e controle direto sobre a *API*, sendo apropriada para aplicações que demandam eficiência computacional e controle granular.

No contexto deste estudo, foi escolhida a biblioteca Concord, por sua implementação na linguagem C, que oferece um equilíbrio entre abstração e controle que a torna adequada para uma ampla gama de aplicações [29]. E também pela familiaridade do autor com a biblioteca, que facilita o desenvolvimento e a manutenção do *bot* educacional proposto.

Para estudos futuros, é importante considerar o uso de linguagens de mais alto nível como Python, JavaScript ou Go, que podem oferecer desenvolvimento mais rápido e maior facilidade de manutenção para equipes maiores, especialmente em projetos onde a curva de aprendizado reduzida seja prioritária em relação ao desempenho bruto. A linguagem de programação em contextos educacionais deve equilibrar considerações pedagógicas, praticidade de implementação e objetivos específicos do projeto [30].

2.5 TRABALHOS RELACIONADOS

Diversos pesquisadores têm explorado o uso de *bots* em contextos educacionais:

Hien et al. [31] desenvolveram um *bot* para suporte a alunos em um curso de programação, que respondia a dúvidas sobre conceitos e sintaxe. Os resultados mostraram uma redução no tempo de resposta para dúvidas comuns e um aumento na satisfação dos alunos com o suporte recebido.

Demetriadis et al. [32] implementaram um agente conversacional para auxiliar alunos em atividades colaborativas de resolução de problemas. O estudo demonstrou que grupos apoiados pelo *bot* apresentaram maior engajamento e melhores resultados de aprendizagem em comparação com grupos sem suporte automatizado.

Yin et al. [33] compararam o fornecimento de *feedback* formativo aos alunos por *chatbot* e professor. A avaliação indicou que alunos que receberam *feedback* regularmente do *chatbot* apresentaram maior interesse de aprendizagem, além de redução da carga cognitiva necessária para aprender conceitos complexos.

Um trabalho particularmente relevante é o de Winkler e Söllner [16], por propor diretrizes para o *design* de *chatbots* educacionais focados em metodologias ativas, destacando a importância de promover interações que estimulem o pensamento crítico e a reflexão.

2.6 OBJETIVOS DO TRABALHO

Com base na revisão bibliográfica apresentada, este trabalho tem como objetivo desenvolver e avaliar um *bot* educacional assistivo e conversacional para plataformas de colaboração,

especificamente o Discord, que facilite a implementação de metodologias ativas em ambientes de ensino remoto.

Os objetivos específicos são:

1. **Desenvolver um *bot* educacional** que incorpore os três princípios fundamentais para interação mediada discutidos na Seção 2.3.3: comunicação multidirecional, engajamento ativo e adaptação contextual.
2. **Implementar funcionalidades específicas** que abordem diretamente os desafios do ensino remoto identificados na Seção 2.3.1, com foco especial em manter o engajamento dos alunos, facilitar o *feedback* imediato e promover interações sociais significativas em ambientes virtuais.
3. **Proporcionar uma integração não-invasiva** da ferramenta ao fluxo de trabalho docente, aplicando os princípios de *design* centrado no usuário e minimizando a carga cognitiva adicional, conforme destacado nos estudos de IHC educacional na Seção 2.3.2.
4. **Criar uma prova de conceito funcional** utilizando tecnologias adequadas ao contexto educacional, considerando aspectos de eficiência, portabilidade e manutenção, conforme discutido na Seção 2.4.
5. **Estabelecer uma metodologia de avaliação experimental** com participantes reais assumindo os papéis de professor e alunos, onde o professor utiliza o *dashboard* de controle enquanto os alunos interagem com o *bot* em um ambiente de sala de aula simulado, permitindo mensurar a eficácia da solução em situações próximas ao uso real, combinando métricas quantitativas e qualitativas, inspirada nos trabalhos apresentados na Seção 2.5.

Tecnicamente, o desenvolvimento será realizado utilizando a biblioteca Concord em C (desenvolvida pelo autor), aproveitando suas vantagens em termos de controle granular sobre a *API* do Discord e sua relevância educacional, conforme discutido na Seção 2.4.

Este trabalho se diferencia dos esforços anteriores apresentados na Seção 2.5 por seu foco específico em facilitar metodologias ativas em ambientes remotos, com ênfase na integração não-invasiva à prática docente e sua metodologia de avaliação que simula condições reais de uso. Enquanto outros trabalhos têm explorado *bots* para responder dúvidas ou fornecer *feedback* automatizado, esta proposta busca transformar a própria dinâmica de interação durante as aulas síncronas, com uma avaliação sistemática em um ambiente controlado.

Os próximos capítulos detalham a concepção e implementação do *bot* (3), bem como sua avaliação através da metodologia experimental com participantes em diferentes papéis (Capítulo 4), demonstrando como os conceitos teóricos discutidos neste capítulo se manifestam na prática educacional.

3 BOT EDUCACIONAL PARA METODOLOGIAS ATIVAS EM AMBIENTES VIRTUAIS

Este capítulo apresenta a concepção do *bot* educacional desenvolvido neste trabalho, sua arquitetura funcional e como ele se integra ao ambiente de ensino remoto para facilitar metodologias ativas. Para ilustrar a aplicação prática, utilizaremos um exemplo concreto baseado em uma aula de programação da disciplina CI1055 (Algoritmos e Estruturas de Dados I) do DINF da UFPR [34].

3.1 VISÃO CONCEITUAL DA APLICAÇÃO

O *bot* educacional proposto foi concebido como um mediador de interações em ambientes virtuais de aprendizagem, especificamente voltado para facilitar a implementação de metodologias ativas durante sessões de ensino remoto. O sistema atua como uma ponte entre professor e alunos, promovendo trocas mais naturais de informações e *feedback*.

A Figura a seguir ilustra o modelo conceitual de interação entre os participantes do processo educacional mediado pelo *bot*:

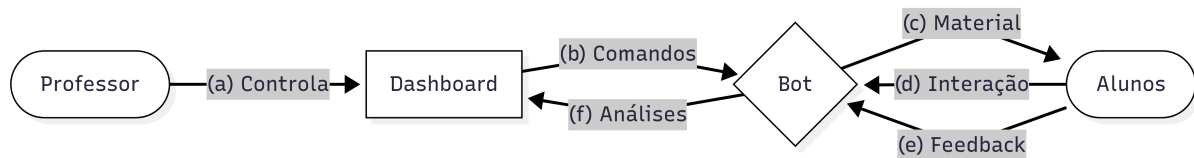


Figure 3.1: Fluxo de interações no ambiente educacional virtual mediado pelo *bot*. A figura mostra um diagrama com o professor à esquerda, o *dashboard* do professor como interface de controle, o *bot* educacional ao centro, e os alunos à direita, ilustrando os fluxos de comunicação: (a) professor controlando a aula via *dashboard*, (b) *dashboard* enviando comandos ao *bot*, (c) *bot* processando e disponibilizando o material para os alunos, (d) alunos interagindo com o conteúdo, (e) *bot* coletando *feedback* dos alunos, e (f) *bot* fornecendo análises em tempo real ao professor através do *dashboard*.

O modelo de interação implementado neste trabalho fundamenta-se nos três princípios para interação mediada por *bots* na educação discutidos na Seção 2.3.3: comunicação multidirecional, engajamento ativo e adaptação contextual. Estes princípios nortearam todo o processo de *design* e desenvolvimento da solução, garantindo que o *bot* efetivamente contribua para a implementação de metodologias ativas no ambiente virtual.

3.2 INTEGRAÇÃO COM O AMBIENTE EDUCACIONAL

O *bot* foi projetado para se integrar ao Discord, pelos motivos discutidos na Seção 2.4. A integração sutil com o ambiente educacional, refere-se à capacidade do *bot* de participar do processo educacional sem causar rupturas no fluxo natural da aula ou exigir mudanças drásticas nas práticas pedagógicas já estabelecidas. Essa sutileza manifesta-se em três dimensões:

1. **Presença não-intrusiva:** O *bot* não interrompe a condução da aula, apenas complementa as atividades quando solicitado ou programado.
2. **Curva de aprendizado reduzida:** Professores e alunos não precisam dominar ferramentas complexas, pois as interações ocorrem através de comandos intuitivos e reações simples.

3. **Flexibilidade metodológica:** O sistema adapta-se a diferentes estilos de ensino, não impondo uma abordagem pedagógica específica.

Para materializar esta integração, o sistema também disponibiliza um *dashboard* específico para uso do professor, como visto em 2.3.4, que permite o controle da aula de forma centralizada e intuitiva, sem a necessidade de comandos complexos ou interrupções no fluxo de comunicação, como será detalhado na seção 3.2.1.

3.2.1 *Dashboard* do Professor

Um elemento chave do sistema é o *dashboard* exclusivo para o professor, que permite controlar o fluxo da aula sem a necessidade de inserir comandos no *chat* principal. Este *dashboard* é apresentado como uma interface web segura, acessível apenas pelo professor, que se comunica com o *bot* em tempo real. Esta abordagem está diretamente alinhada com o objetivo de proporcionar uma integração não-invasiva ao fluxo de trabalho docente, conforme estabelecido na Seção 2.6 do Capítulo 2. Através dele, o professor pode:

- Visualizar estatísticas de engajamento dos alunos em tempo real
- Receber alertas sobre dúvidas e dificuldades dos alunos
- Lançar atividades interativas e acompanhar seu progresso
- Obter relatórios detalhados sobre o desempenho da turma
- Destacar respostas e discussões dos alunos para promover a colaboração

Esta abordagem permite que o professor mantenha o controle pedagógico da aula sem interrupções no fluxo da comunicação, enquanto os alunos interagem diretamente com o *bot* através de comandos e reações no ambiente do Discord.

3.3 RECURSOS PARA PROMOÇÃO DE METODOLOGIAS ATIVAS

O *bot* implementa diversos recursos específicos para viabilizar metodologias ativas em ambiente remoto. Na seção 3.3.1 é apresentado mecanismos de *feedback* em tempo real que permitem ao professor avaliar continuamente a compreensão dos alunos, e 3.3.2 trata das funcionalidades específicas para aprendizagem baseada em problemas que estimulam o raciocínio crítico e a resolução de situações práticas.

3.3.1 *Feedback* em Tempo Real

Um dos principais desafios do ensino remoto é perceber as reações dos alunos. O *bot* permite que os estudantes expressem sua compreensão ou dúvidas durante a explanação, sem interromper o fluxo da aula, através de:

- **Barômetro de compreensão:** Interface visual que agrega as reações dos alunos
- **Alertas de dificuldade:** Notificação ao professor quando um número significativo de alunos indica não compreender um tópico
- **Dúvidas anônimas:** Permite que alunos enviem questões sem exposição pública

3.3.2 Aprendizagem Baseada em Problemas

A aprendizagem baseada em problemas (vide Seção 1) envolve vários elementos que podem ser incluídos em nosso aplicativo. Dentre esses elementos, incluímos os seguintes, por serem os mais simples de se implementar:

- **Desafios temporizados:** Problemas com tempo definido para resolução
- **Pistas progressivas:** Sugestões que são liberadas gradualmente durante a resolução
- **Compilação e execução de código:** Para disciplinas de programação, execução segura de códigos submetidos pelos alunos

3.4 EXEMPLO PRÁTICO: AULA DE COMANDOS DE REPETIÇÃO

Para ilustrar a aplicação concreta do *bot* em um contexto educacional real, apresentamos a seguir um cenário baseado em uma aula da disciplina CI1055 - Algoritmos e Estruturas de Dados I, ministrada no Departamento de Informática da UFPR. O exemplo demonstra como o *bot* auxilia o professor durante uma aula sobre "Comandos de Repetição" em Pascal. Nas próximas seções, detalharemos as etapas de preparação da aula pelo professor e as interações que ocorrem durante a sessão síncrona, evidenciando como os recursos do *bot* facilitam a implementação das metodologias ativas.

3.4.1 Preparação da Aula

Antes da aula, o professor utiliza o *dashboard* para preparar o material didático:

```

1 [Dashboard do Professor]
2 > Criar Nova Aula
3 Título: "Comandos de Repetição em Pascal"
4 Descrição: "Introdução aos comandos de repetição em Pascal com foco no
5 comando while"
6
7 Tópicos: "Loops", "Comando while", "Repetição", "Pascal"
8
9 > Adicionar Conteúdo
10 [Título] "Objetivos da aula"
11 [Conteúdo] "Introduzir conceitos de repetição, apresentar o comando while,
12 resolver exemplos práticos"
13 [Tipo] Slide
14
15 > Adicionar Conteúdo
16 [Título] "Exemplo inicial: imprimir números de 1 a 5"
17 [Conteúdo]
18 '''
19 program imprimir_de_1_a_5;
20 begin
21   writeln(1);
22   writeln(2);
23   writeln(3);
24   writeln(4);
25   writeln(5);
26 end.
27 '''
28 [Tipo] Código Pascal

```

```

28 > Configurar Quiz
29 [Pergunta] "Ao incrementar uma variável dentro de um loop while,
30     qual operação utilizamos em Pascal?"
31 [Opções]
32 - "i := i + 1" (CORRETA)
33 - "i++"
34 - "i += 1"
35 - "increment(i)"
36 [Tempo] 60 segundos

```

O sistema confirma a criação da aula e fornece um código de acesso para os alunos.

3.4.2 Interação Durante a Aula

Durante a aula síncrona, as seguintes interações ocorrem:

```

1 [Dashboard do Professor]
2 > Iniciar Aula "Comandos de Repetição em Pascal"
3 Sistema: Aula iniciada. Os alunos podem acessar o canal de vídeo usando o c
  código
4     de acesso enviado em sua mensagem privada.
5
6 [Discord - Canal da Aula]
7 Bot: @everyone O professor iniciou a aula "Comandos de Repetição em Pascal".
8
9     Entre no canal de videoconferência para confirmar sua presença.
10
11 [Vários alunos entram no canal de vídeo]
12 Professor [no canal de voz]: Vamos começar entendendo por que precisamos de
  comandos
13     de repetição. Observem este exemplo inicial no canal.
14
15 [Dashboard do Professor]
16 > Mostrar Código "Exemplo inicial"
17 Sistema: Código exibido no canal #exemplos-de-codigo.
18
19 [Discord - Canal #exemplos-de-codigo]
20 Bot:
21 ```pascal
22 program imprimir_de_1_a_5;
23 begin
24     writeln(1);
25     writeln(2);
26     writeln(3);
27     writeln(4);
28     writeln(5);
29 end.
30 ```
31 Para testar este código, utilize /executar
32
33 [Dashboard do Professor]
34 > Iniciar Discussão
35 [Pergunta] "Qual o problema desta abordagem se quisermos imprimir de 1 até
  1000?"
36
37 [Discord - Canal da Aula]

```

```

38 Bot: DISCUSSÃO: Qual o problema desta abordagem se quisermos imprimir de 1
    até 1000?
39     Use /responder para participar da discussão.
40
41 Aluno1: /responder Teríamos que escrever mil linhas de código!
42 Aluno2: /responder Código muito repetitivo e difícil de manter.
43
44 [Dashboard do Professor - Painel de Engajamento]
45 Status: 15/23 alunos responderam
46 Participação ativa: 65%
47 Respostas mais comuns: "código repetitivo" (60%), "muitas linhas" (27%)
48
49 [Alguns alunos usam reações no Discord]
50 [5 alunos reagem com "joinha" (entendi)]
51 [2 alunos reagem com "?" (tenho dúvida)]
52
53 [Dashboard do Professor - Alertas]
54 ATENÇÃO 2 alunos indicaram dúvidas sobre o conceito atual.
55 Recomendação: Revisitar o conceito com uma abordagem alternativa.
56
57 Professor [no canal de voz]: Estou vendo que temos algumas dúvidas.
58     Vamos revisitar o conceito de forma diferente.
59
60 [Dashboard do Professor]
61 > Mostrar Exemplo Interativo
62 [Título] "Loop while básico"
63 [Código]
64 ```pascal
65 program exemplo;
66 var i: integer;
67 begin
68     i := 1;
69     while i <= 5 do
70     begin
71         writeln(i);
72         i := i + 1;
73     end;
74 end.
75 ```
76 [Opções] Ativar execução por alunos
77
78 [Discord - Canal #exemplos-de-codigo]
79 Bot: EXEMPLO INTERATIVO: Loop while básico
80 ```pascal
81 program exemplo;
82 var i: integer;
83 begin
84     i := 1;
85     while i <= 5 do
86     begin
87         writeln(i);
88         i := i + 1;
89     end;
90 end.
91 ```
92 Use /executar para ver o resultado deste código.
93
94 Aluno5: /executar

```

```

95 Bot:
96 ```
97 1
98 2
99 3
100 4
101 5
102 ```
103
104 Aluno8: /duvida O que acontece se eu esquecer de incrementar i dentro do
      loop?
105 Bot: @Professor Dúvida enviada anonimamente: "O que acontece se eu esquecer
      de incrementar i dentro do loop?"
106
107 [Dashboard do Professor]
108 > Responder Dúvida
109 [Criar Exemplo] "Loop infinito"
110 ```pascal
111 program loop_infinito;
112 var i: integer;
113 begin
114     i := 1;
115     while i <= 5 do
116     begin
117         writeln(i);
118         // i não éincrementado
119     end;
120 end.
121 ```
122
123 [Discord - Canal #exemplos-de-codigo]
124 Bot: Resposta àdúvida: O que acontece se esquecer de incrementar i
125 ```pascal
126 program loop_infinito;
127 var i: integer;
128 begin
129     i := 1;
130     while i <= 5 do
131     begin
132         writeln(i);
133         // i não éincrementado
134     end;
135 end.
136 ```
137 Aviso: O código acima contém um loop infinito. A execução seria
      interrompida após
138 repetir o mesmo valor várias vezes:
139 ```
140 1
141 1
142 1
143 1
144 ...
145 ```
146
147 [Dashboard do Professor]
148 > Criar Desafio
149 [Título] "Números pares"

```

```

150 [Descrição] "Escreva um programa que leia um número n e imprima todos os nú-
    meros pares de 2 até n"
151 [Tempo] 10 minutos
152 [Testes Automáticos]
153 - Entrada: "10" | Saída esperada: "2 4 6 8 10"
154 - Entrada: "7" | Saída esperada: "2 4 6"
155
156 [Discord - Canal #desafios]
157 Bot: NOVO DESAFIO: Números pares
158 Escreva um programa que leia um número n e imprima todos os números pares
    de 2 até n.
159 Tempo: 10 minutos
160 Use /iniciar-desafio para começar a resolver.
161
162 [Vários alunos utilizam o comando /iniciar-desafio, recebendo acesso a
    canais privados para resolver o problema]
163
164 [Dashboard do Professor - Monitor de Progresso]
165 15/23 alunos iniciaram o desafio
166 8/23 já submeteram soluções
167 6/23 passaram em todos os testes
168
169 [Após o tempo definido]
170 [Dashboard do Professor]
171 > Encerrar Desafio
172 > Compartilhar Solução Anônima [ID: 3]
173
174 [Discord - Canal #desafios]
175 Bot: O tempo para o desafio "Números pares" encerrou!
176 Estatísticas:
177 - 19/23 alunos participaram
178 - 15/19 completaram o desafio com sucesso
179
180 Solução compartilhada (anônima):
181 ```pascal
182 program numeros_pares;
183 var i, n: integer;
184 begin
185     read(n);
186     i := 2;
187     while i <= n do
188     begin
189         writeln(i);
190         i := i + 2;
191     end;
192 end.
193 ```
194
195 [Dashboard do Professor]
196 > Encerrar Aula
197 > Gerar Relatório
198
199 [Discord - Canal da Aula]
200 Bot: Esta aula foi encerrada. Um resumo das atividades foi enviado ao
    professor.
201     Todo o material continua disponível para consulta nos canais respectivos
    .
202

```

```
203 [E-mail do Professor - Relatório Automático]
204 Resumo da aula: "Comandos de Repetição em Pascal"
205 - Participação: 23/25 alunos ativos (92%)
206 - Média de interações por aluno: 7.2
207 - Tópicos com mais dúvidas: "loop infinito" (5 menções), "incremento de
    variáveis" (3 menções)
208 - Desafio "Números pares": 19/23 participaram, 15/19 completaram com
    sucesso
209 - Alunos com participação abaixo do esperado: 2 (lista anexa)
210 - Recomendação: Reforçar o conceito de incremento de variáveis na próxima
    aula
```

4 PROVA DE CONCEITO: *BOT* PARA INTERAÇÃO EDUCACIONAL

Este capítulo apresenta a prova de conceito do *bot* educacional desenvolvido para este trabalho, detalhando sua implementação técnica e a metodologia de avaliação experimental proposta. O *dashboard* contém elementos que são a representação gráfica do *bot*, permitindo ao professor gerenciar as interações conforme estabelecido no Capítulo 2.

4.1 CONTEXTO DA INTERAÇÃO PROFESSOR-ALUNO EM AMBIENTES REMOTOS

Antes de detalhar os aspectos técnicos da implementação, é importante contextualizar como ocorre a interação entre professor e alunos em um ambiente de ensino remoto, particularmente quando se busca aplicar metodologias ativas.

Em aulas remotas tradicionais, observa-se frequentemente um padrão de comunicação unidirecional, onde o professor transmite o conteúdo enquanto os alunos assumem postura predominantemente passiva. As interações tendem a ser limitadas a momentos específicos, como sessões de perguntas ao final da aula, ou através de canais assíncronos como *e-mails* e fóruns. Este modelo apresenta barreiras significativas à implementação de metodologias ativas, que dependem de ciclos rápidos de *feedback* e participação constante dos estudantes.

O *bot* proposto busca transformar este paradigma ao introduzir um mediador digital que facilita:

1. **Trocas síncronas durante a exposição de conteúdo:** Permitindo reações e dúvidas sem interromper o fluxo da aula
2. **Anonimato seletivo para alunos:** Reduzindo a inibição de participação
3. **Coleta sistemática de dados de interação:** Possibilitando ajustes em tempo real na condução da aula
4. **Automação de tarefas repetitivas:** Liberando o professor para focar em aspectos pedagógicos mais relevantes

Entendemos que estas características são fundamentais para aproximar o ambiente virtual das dinâmicas interativas observadas em salas de aula presenciais onde metodologias ativas são aplicadas com sucesso.

4.2 IMPLEMENTAÇÃO TÉCNICA

A implementação técnica do sistema educacional segue uma arquitetura dual composta pelo *bot* Discord e pelo *dashboard* do professor, conforme conceitualmente apresentado na Seção 2.3.4 do Capítulo 2. Esta arquitetura garante a separação entre o canal de comando (exclusivo do professor) e o canal de interação (compartilhado entre todos os participantes).

4.2.1 Arquitetura do *Bot*

O *bot* foi desenvolvido utilizando a biblioteca Concord em C (desenvolvida pelo autor deste trabalho). A implementação seguiu uma arquitetura modular organizada em quatro componentes principais:

- **Módulo de Publicação:** Responsável por processar comandos do professor vindos do *dashboard* e transformá-los em conteúdo formatado nos canais apropriados do Discord. Este módulo implementa recursos de formatação para código, imagens e outros materiais didáticos.
- **Módulo de Interação:** Gerencia as reações e comandos dos alunos, incluindo o processamento de slash commands, reações com emojis e mensagens diretas. Este componente implementa o princípio de comunicação multidirecional discutido na Seção 2.3.3.
- **Módulo de Análise:** Coleta e processa em tempo real as interações para gerar métricas de engajamento, barômetros de compreensão e outros indicadores pedagógicos relevantes. Os resultados são transmitidos ao *dashboard* do professor para visualização.
- **Módulo de Persistência:** Armazena dados estruturados sobre a sessão para análise posterior, possibilitando a geração de relatórios detalhados e o acompanhamento longitudinal do progresso dos alunos ao longo de múltiplas aulas.

4.2.2 Implementação do *Dashboard*

O *dashboard* do professor foi implementado como uma aplicação web utilizando tecnologias modernas de frontend (React.js) e backend (Node.js), comunicando-se com o *bot* através de uma *API REST* segura. Esta separação arquitetural permite que o professor mantenha uma interface de controle independente e privada, sem necessidade de interagir diretamente no *chat* público.

O sistema de comunicação entre *dashboard* e *bot* utiliza um protocolo de mensagens baseado em WebSockets para garantir atualizações em tempo real e baixa latência, aspectos cruciais para o controle efetivo da dinâmica da aula. Esta comunicação bidirecional permite:

1. Envio de comandos do professor para o *bot* (publicação de conteúdo, criação de atividades)
2. Transmissão de métricas e alertas do *bot* para o *dashboard* (nível de engajamento, dúvidas anônimas)
3. Sincronização do estado da aula entre múltiplas sessões de navegador, caso o professor precise alternar entre dispositivos

4.2.3 Integração Técnica

A integração com o Discord foi realizada através das *APIs* fornecidas pela biblioteca Concord.

A arquitetura implementa os cinco componentes essenciais de um *bot* educacional descritos na Seção 2.1: interface do usuário (canais do Discord), compreensão de linguagem natural (processamento de comandos), gerenciador de diálogo (módulo de interação), integração com backend (*dashboard* e sistemas de persistência) e geração de resposta (módulo de publicação).

Esta implementação atende diretamente ao objetivo de criar uma prova de conceito funcional utilizando tecnologias adequadas ao contexto educacional, conforme estabelecido na Seção 2.6 do Capítulo 2.

4.3 FUNCIONALIDADES IMPLEMENTADAS

O sistema desenvolvido consiste em dois componentes principais que trabalham de forma integrada: (1) o *bot* educacional que interage diretamente com os alunos no Discord e (2) o *dashboard* exclusivo para o professor que permite gerenciar essas interações. Esta arquitetura dual implementa o princípio de "separação de interesses" discutido na Seção 2.3.4 do Capítulo 2, onde o canal de comando (*dashboard*) é separado do canal de interação (Discord).

4.3.1 Funcionalidades do *Bot* no Discord

O *bot* no ambiente Discord oferece as seguintes funcionalidades:

1. **Mecanismos de *feedback* rápido:** Permite que alunos utilizem reações para indicar seu nível de compreensão (como "entendi", "tenho dúvida", "confuso"), criando um barômetro de compreensão em tempo real.
2. **Canal de dúvidas anônimas:** Os alunos podem enviar dúvidas de forma privada para o *bot*, que as encaminha ao professor sem identificar o remetente, reduzindo a inibição.
3. **Execução de código:** Para disciplinas de programação, o *bot* permite a execução segura de snippets de código submetidos pelos alunos, mostrando resultados em tempo real.
4. **Atividades interativas:** Disponibiliza *quizzes*, enquetes e desafios temporalizados, coletando e organizando as respostas dos alunos automaticamente.

4.3.2 Funcionalidades do *Dashboard* do Professor

O *dashboard*, como interface de controle pedagógico, implementa as seguintes funcionalidades:

1. **Sistema de alertas:** Notificações automáticas quando determinados padrões são detectados, como uma quantidade significativa de alunos indicando dificuldade.
2. **Gerenciador de atividades:** Ferramentas para criar, lançar e monitorar atividades interativas em tempo real.
3. **Relatórios pós-aula:** Geração de resumos detalhados após a sessão, incluindo métricas de participação, desempenho e tópicos problemáticos.

Esta integração entre *dashboard* e *bot* cria um sistema coeso que permite ao professor manter o controle pedagógico da aula enquanto facilita interações dinâmicas com os alunos. O professor pode, por exemplo, identificar rapidamente conceitos que geraram confusão através do *dashboard* e adaptar sua abordagem ou enviar explicações adicionais através do *bot*, sem interromper o fluxo da aula.

A Figura 2.2 (Capítulo 2) ilustra esta relação integrada, onde o *dashboard* atua como interface de comando exclusiva do professor, enquanto o *bot* serve como ponto de contato e interação para todos os participantes.

4.4 METODOLOGIA DE AVALIAÇÃO

A metodologia de avaliação para o *bot* educacional consiste em uma abordagem experimental com participantes reais assumindo os papéis de professor e alunos em um ambiente de sala de aula simulado, conforme delineado na Seção 2.6 do Capítulo 2. Esta abordagem permite avaliar a eficácia da ferramenta em condições próximas ao uso real, combinando métricas quantitativas e qualitativas.

4.4.1 Ambiente e Participantes

O teste experimental envolve:

- Professores de disciplinas de graduação na área de computação, que utilizarão o *dashboard* de controle para gerenciar as interações
- Alunos de graduação, que interagirão com o *bot* através da interface do Discord
- Sessões de aula remotas simuladas, reproduzindo cenários pedagógicos típicos
- Observadores para registrar aspectos qualitativos da interação

4.4.2 Coleta de Dados

Os dados são coletados através de três mecanismos principais:

1. **Questionários:** Aplicados a professores e alunos para medir percepções sobre o uso do *bot*
2. **Registros automáticos (*logs*):** Dados quantitativos sobre frequência e tipos de interações realizadas
3. **Entrevistas:** Conduzidas com participantes para obter insights qualitativos

4.4.3 Métricas de Avaliação

As seguintes métricas são utilizadas para avaliar a eficácia da solução:

Table 4.1: Métricas para avaliação da eficácia do *bot*

Categoria	Métricas
Engajamento	Número de interações por aluno, distribuição temporal das interações, diversidade de tipos de interação
Impacto pedagógico	Mudanças na condução da aula, percepção de compreensão do conteúdo, tempo dedicado a esclarecimentos
Usabilidade	Facilidade de uso, problemas técnicos, curva de aprendizado
Metodologias ativas	Viabilidade de implementação, comparação com experiências presenciais

4.5 RESULTADOS ILUSTRATIVOS

4.5.1 Dados Quantitativos

4.5.2 Análise Qualitativa

Aspectos a serem analisados incluem:

- Facilidade dos alunos em expressar dúvidas através do *bot*
- Percepção dos professores sobre a identificação de tópicos problemáticos
- Curva de aprendizado da ferramenta
- Natureza não invasiva da ferramenta como fator de adoção

5 CONCLUSÃO

A implementação do *bot* desenvolvido com a biblioteca Concord demonstrou que é possível integrar, de forma natural e não invasiva, tecnologias de interação ativa ao ambiente de ensino remoto. A ferramenta permitiu que alunos interagissem com o conteúdo de aula e expressassem *feedbacks* espontâneos, contribuindo para a adaptação do professor em tempo real.

5.1 SÍNTESE DOS RESULTADOS

5.2 LIMITAÇÕES DO ESTUDO

A metodologia e implementação deste trabalho apresentam algumas limitações que devem ser consideradas na interpretação dos resultados e no planejamento de estudos futuros:

1. **Amostra limitada:** O número de participantes e disciplinas pode não representar adequadamente todos os contextos educacionais, restringindo a generalização dos resultados.
2. **Efeito novidade:** O interesse inicial pela tecnologia pode ter influenciado positivamente os resultados de curto prazo, um fenômeno comum em estudos de novas tecnologias educacionais.
3. **Viés de seleção:** Os participantes voluntários podem não representar adequadamente o perfil completo de alunos e professores, particularmente no que se refere a diferentes níveis de familiaridade tecnológica.
4. **Duração reduzida:** Enquanto um curso real se estende por semanas ou meses, o experimento teve duração limitada, impossibilitando a observação de efeitos de longo prazo como adaptação dos usuários, fadiga ou mudanças no engajamento ao longo do tempo.
5. **Diversidade contextual:** O experimento foi realizado em um contexto específico (disciplina de programação em nível de graduação), o que pode limitar a generalização dos resultados para outros contextos educacionais que possuem necessidades e dinâmicas distintas.
6. **Ambiente simulado:** Embora o experimento tenha buscado recriar condições próximas à realidade, o ambiente de sala de aula simulado pode não ter reproduzido completamente as dinâmicas e desafios de um curso real, como discutido na Seção 2.3.1. Os participantes estavam cientes de que estavam em um experimento, o que pode ter afetado seu comportamento (efeito *Hawthorne*).
7. **Avaliação de IHC:** Conforme discutido na Seção 2.3.2, interfaces centradas no usuário exigem avaliações extensivas. O experimento realizado pode não ter capturado todos os aspectos de usabilidade e experiência do usuário necessários para uma avaliação completa dos princípios de *design* centrado no usuário.
8. **Limitações técnicas:** A implementação em C usando a biblioteca Concord, embora tenha proporcionado controle granular sobre a *API*, pode apresentar desafios de manutenção e extensibilidade em comparação com linguagens de mais alto nível.

Reconhecer estas limitações oferece perspectivas importantes para a interpretação dos resultados e para o direcionamento de pesquisas futuras neste campo.

5.3 CONTRIBUIÇÕES DO TRABALHO

Este trabalho contribui para o campo da educação digital ao:

1. Demonstrar a viabilidade de integração sutil de tecnologias interativas no ensino remoto
2. Fornecer evidências empíricas sobre o impacto positivo de *bots* educacionais no engajamento
3. Propor uma arquitetura flexível e de baixo impacto para interações educacionais
4. Estabelecer métricas para avaliação da eficácia de ferramentas de interação em ambientes virtuais

5.4 TRABALHOS FUTUROS

Para trabalhos futuros, recomenda-se:

1. Expandir a aplicação do *bot* para outros contextos e disciplinas além da programação
2. Integrar suporte a outras plataformas educacionais (como Moodle, Google Meet, BBB (*Big Blue Button*), etc.)
3. Explorar o uso de inteligência artificial para tornar os *feedbacks* ainda mais personalizados e contextuais
4. Realizar estudos longitudinais para avaliar o impacto a longo prazo no desempenho acadêmico
5. Desenvolver recursos adicionais para análise de padrões de interação e geração de insights pedagógicos
6. Investigar o potencial do *bot* como ferramenta de avaliação contínua e formativa
7. Desenvolver *dashboards* com métricas visuais consolidadas de engajamento e participação dos alunos

A promissora interseção entre tecnologia e educação continua a oferecer oportunidades para melhorar a experiência de ensino-aprendizagem, especialmente em contextos remotos, e o presente trabalho busca contribuir com este avanço.

REFERENCES

- [1] X. Huang. Chatbot: Design, architecture, and applications. Master's thesis, University of Pennsylvania, 05 2021. ASCS CIS 498 Senior Capstone Thesis.
- [2] Fabiane Viana, Diego Jorge Dantas, and Ferreira. A COVID-19 e os desafios educacionais. *Revista Científica Multidisciplinar Núcleo do Conhecimento*, 05:1–22, 04 2024. doi: 10.55470/relaec.44455.
- [3] Michael Prince. Does active learning work? a review of the research. *Journal of Engineering Education*, 93:223–231, 07 2004. doi: 10.1002/j.2168-9830.2004.tb00809.x.
- [4] Elaine Yew and Karen Goh. Problem-based learning: An overview of its process and impact on learning. *Health Professions Education*, 2:75–79, 05 2016. doi: 10.1016/j.hpe.2016.01.004.
- [5] David van Alten, C. Phielix, Jeroen Janssen, and Liesbeth Kester. Effects of flipping the classroom on learning outcomes and satisfaction: A meta-analysis. *Educational Research Review*, 28:1–18, 06 2019. doi: 10.1016/j.edurev.2019.05.003.
- [6] Marjan Laal and Seyed Ghodsi. Benefits of collaborative learning. *Procedia - Social and Behavioral Sciences*, 31:486–490, 12 2012. doi: 10.1016/j.sbspro.2011.12.091.
- [7] Chinedu Wilfred Okonkwo and Abejide Ade-Ibijola. Chatbots applications in education: A systematic review. *Computers and Education: Artificial Intelligence*, 2:100033, 2021. doi: 10.1016/j.caeai.2021.100033.
- [8] Robert Dale. The return of the chatbots. *Natural Language Engineering*, 22:811–817, 09 2016. doi: 10.1017/S1351324916000243.
- [9] Meisam Eslahi, Rosli Salleh, and Nor Anuar. Bots and botnets: An overview of characteristics, detection and challenges. In *2012 IEEE International Conference on Control System, Computing and Engineering*, pages 349–354, 11 2012. doi: 10.1109/ICCSCE.2012.6487169.
- [10] Carlene Lebeuf, Alexey Zagalsky, Matthieu Foucault, and Margaret-Anne Storey. Defining and classifying software bots: a faceted taxonomy. In *2019 IEEE/ACM 1st international workshop on bots in software engineering (BotSE)*, pages 1–6. IEEE, 2019. doi: 10.1109/BotSE.2019.00008.
- [11] Joseph Seering, Juan Pablo Flores, Saiph Savage, and Jessica Hammer. The social roles of bots: Evaluating impact of bots on discussions in online communities. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW):1–29, 11 2018. doi: 10.1145/3274426.
- [12] Olaf Zawacki-Richter, Victoria I. Marín, Melissa Bond, and Franziska Gouverneur. Systematic review of research on artificial intelligence applications in higher education—where are the educators? *International Journal of Educational Technology in Higher Education*, 16:39, 10 2019.
- [13] Songhee Han and Min Kyung Lee. Faq chatbot and inclusive learning in massive open online courses. *Computers & Education*, 179:104395, 2022. doi: 10.1016/j.compedu.2021.104395.

- [14] Ligiane Nascimento and Silva. Transformando a educação online: Os impactos da inteligência artificial nos cursos à distância. <http://gestaouniversitaria.com.br/artigos/transformando-a-educacao-online-os-impactos-da-inteligencia-artificial-nos-cursos-a-distancia>, 04 2024.
- [15] Charles Hodges, Stephanie Moore, Barbara Lockee, Torrey Trust, and Mark Bond. The difference between emergency remote teaching and online learning, 03 2020.
- [16] Rainer Winkler and Matthias Söllner. Unleashing the potential of chatbots in education: A state-of-the-art analysis. *Academy of Management Proceedings*, 2018:15903, 04 2018. doi: 10.5465/AMBPP.2018.15903abstract.
- [17] Roy Pea. User centered system design: New perspectives on human-computer interaction. *Journal educational computing research*, 3, 01 1987. available via HAL: fhal-00190545f.
- [18] D. A. Norman. *The design of everyday things: Revised and expanded edition*. MIT Press, 2013.
- [19] John Sweller. *Cognitive Load Theory*. Elsevier, 2023. doi: 10.1016/B978-0-12-818630-5.14020-5.
- [20] Teresa Cristina Calvo and Maté. Estudo sobre a aprendizagem dialógica em ambientes virtuais de aprendizagem. Master's thesis, IFES, 2013. Repositório IFES.
- [21] J. Piaget. *Structuralism*. Basic Books, 1970.
- [22] Katrien Verbert, erik duval, Joris Klerkx, Sten Govaerts, and José Santos. Learning analytics dashboard applications. *American Behavioral Scientist*, 57:1500–1509, 09 2013. doi: 10.1177/0002764213479363.
- [23] Stefan Pietrusky. Development of a chatbot in botpress with own knowledge base and implementation in moodle to support students, 04 2024. URL <https://medium.com/@stefanpietrusky/development-of-a-chatbot-in-botpress-with-own-knowledge-base-and-implementation-in-moodle-to-55d3a46cbaff>.
- [24] Anna Wright, Jillian Rubman, and Joseph Zacher. Enhancing active learning with slack, 08 2024. URL <https://mitsloanedtech.mit.edu/2024/08/28/enhancing-active-learning-with-slack/>.
- [25] Dayana Espinoza, Andre Medina, and Laberiano Andrade-Arenas. Design of the discord application as an e-learning tool at the university of sciences and humanities. 01 2021. doi: 10.18687/LACCEI2021.1.1.9.
- [26] Discord.js Team. Discord.js Documentation. <https://discord.js.org/>, 2025. Accessed: June 1, 2025.
- [27] Rapptz. Discord.py Documentation. <https://discordpy.readthedocs.io/>, 2025. Accessed: June 1, 2025.
- [28] Lucas Müller. Concord: Lightweight discord API c library. <https://github.com/Cogmasters/concord>, 2025. Accessed: June 1, 2025.

- [29] B. W. Kernighan and D. M. Ritchie. *The C Programming Language*. Prentice Hall, 2 edition, 1988.
- [30] Arnold Pears, Stephen Seidman, Lauri Malmi, Linda Mannila, Elizabeth Adams, Jens Bennedsen, Marie Devlin, and James Paterson. A survey of literature on the teaching of introductory programming. *ACM SIGCSE Bulletin*, 39:204–223, 12 2007. doi: 10.1145/1345375.1345441.
- [31] Ho Hien, Cuong Pham-Nguyen, Nam Le Nguyen Hoai, Ho Nhung, and Thang Le Dinh. Intelligent assistants in higher-education environments: The FIT-EBot, a chatbot for administrative and learning support. In *Proceedings of the ninth international symposium on information and communication technology*, pages 69–76, 01 2018. ISBN 978-1-4503-6539-0. doi: 10.1145/3287921.3287937.
- [32] Stavros Demetriadis, Stergios Tegos, Georgios Psathas, Thrasyvoulos Tsiatsos, Armin Weinberger, Santi Caballé, Yannis Dimitriadis, Eduardo Gómez-Sánchez, Pantelis Papadopoulos, and Anastasios Karakostas. Conversational agents as group-teacher interaction mediators in moocs. In *2018 Learning With MOOCS (LWMOOCS)*, pages 43–46, 09 2018. doi: 10.1109/LWMOOCS.2018.8534686.
- [33] Jiaqi Yin, Tiong-Thye Goh, Bing Yang, and Yi Hu. Using a chatbot to provide formative feedback: A longitudinal study of intrinsic motivation, cognitive load, and learning performance. *IEEE Transactions on Learning Technologies*, PP:1–13, 01 2024. doi: 10.1109/TLT.2024.3364015.
- [34] Universidade Federal do Paraná, Departamento de Informática. Ci1055 — algoritmos e estruturas de dados i: Ementa e plano de ensino. Ficha de Disciplina (Unidade Didática) CI1055, Universidade Federal do Paraná, 09 2021. URL <https://www.inf.ufpr.br/bcc/periodos/2021-1/CI1055.pdf>. Semestre 2021-1, Ensino Remoto Emergencial (ERE), carga horária total de 60 h.