

AVALIAÇÃO DA EFICÁCIA DE GRANDES MODELOS DE LINGUAGEM NA GERAÇÃO DE PLANOS DE ESTUDO PERSONALIZADOS BASEADOS NOS ESTILOS DE APRENDIZAGEM HONEY-ALONSO

EVALUATING THE EFFICACY OF LARGE LANGUAGE MODELS IN GENERATING PERSONALIZED STUDY PLANS BASED ON THE HONEY-ALONSO LEARNING STYLES

Davi Wesley Monteiro Gomes da Silva

dwmgs@discente.ifpe.edu.br

Flávio Rosendo da Silva Oliveira

flavio.oliveira@paulista.ifpe.edu.br

RESUMO

A personalização da aprendizagem tem se tornado um tema cada vez mais relevante diante da crescente aplicação das tecnologias digitais no contexto educacional. Nesse cenário, os Large Language Models (LLMs) apresentam potencial para auxiliar na elaboração de planos de estudo adaptados às características individuais dos discentes. Este trabalho investiga a utilização de LLMs na construção de planos de estudo personalizados com base nos estilos de aprendizagem definidos pela teoria Honey-Alonso (CHAEA). Para isso, foi realizada uma pesquisa de caráter exploratório, fundamentada em revisão bibliográfica acerca dos estilos de aprendizagem, da inteligência artificial (IA) e dos Large Language Models. A metodologia proposta contempla a definição de cenários educacionais, a elaboração de um modelo padronizado para geração de planos de estudo e a construção de critérios destinados à avaliação dos materiais produzidos pelos LLMs analisados. A pesquisa busca contribuir para a compreensão das possibilidades de aplicação da IA generativa na educação, especialmente no contexto da personalização do ensino e do apoio ao processo de aprendizagem, fornecendo subsídios para futuras investigações sobre o uso dessas tecnologias como ferramentas de suporte educacional.

Palavras-chave: Inteligência Artificial; Large Language Models; Personalização da Aprendizagem; Estilos de Aprendizagem; CHAEA; Educação.

ABSTRACT

Personalized learning has become an increasingly relevant topic due to the growing application of digital technologies in educational contexts. In this scenario, Large Language Models (LLMs) have shown potential to support the development of study plans tailored to students' individual characteristics. This study investigates the use of LLMs in the creation of personalized study plans based on the learning styles defined by the Honey-Alonso Learning Styles Questionnaire (CHAEA). To achieve this objective, an exploratory research approach was conducted, supported by a literature review on learning styles, artificial intelligence, and Large Language Models. The proposed methodology includes the definition of educational scenarios, the development of a standardized model for study plan generation, and the establishment of criteria for evaluating the materials produced by the analyzed LLMs. This research seeks to contribute to the understanding of the potential applications of generative artificial intelligence in education, particularly in the context of personalized learning and support for the learning process, providing a foundation for future investigations into the use of these technologies as educational support tools.

Keywords: Artificial Intelligence; Large Language Models; Personalized Learning; Learning Styles; CHAEA; Education.

1 INTRODUÇÃO

Segundo a UNESCO (2023, p. 7), "a tecnologia digital aumentou de forma dramática o acesso a recursos de ensino e aprendizagem". Esse avanço tecnológico tem transformado o cenário educacional, em que a migração dos meios tradicionais para o meio digital vem se tornando cada vez mais presente no cotidiano educacional. Um ponto crucial neste processo é o acompanhamento individual e personalizado, pois, como destacam Bacich, Tanzi Neto e Trevisani (2015), a tecnologia permite romper com o modelo de educação padronizada. Desse modo, a adaptação do ensino torna-se viável, considerando que cada indivíduo tem seus próprios gostos e preferências. Para que a adaptação seja efetiva, é necessário compreender os possíveis perfis dos estudantes. Nesse sentido:

Os estilos de aprendizagem são os traços cognitivos, afetivos e fisiológicos que servem como indicadores relativamente estáveis da forma como os estudantes percebem, interagem e respondem aos seus ambientes de aprendizagem (Alonso; Gallego; Honey, 2007, p. 48, tradução do autor).

Diante deste cenário, a utilização de Inteligência Artificial (IA) – área da computação focada no desenvolvimento de sistemas capazes de perceber ambientes, raciocinar e agir para alcançar objetivos (Russell; Norvig, 2022) –, mais especificamente dos *Large Language Models* (LLMs) surge como uma proposta interessante. Os LLMs, definidos como modelos computacionais avançados treinados em escalas massivas de dados textuais para compreender e gerar linguagem natural (Zhao *et al.*, 2025), apresentam grande potencial para construir planos de estudo específicos com base nas particularidades de cada discente. Este trabalho propõe uma investigação sobre o uso de diferentes LLMs e suas aplicações para a aprendizagem de estudantes. A partir disso, busca-se analisar como os modelos podem contribuir para a construção de planos de estudo personalizados.

O restante deste trabalho possui a seguinte estrutura: a Seção 2 aborda o Referencial Teórico onde serão discutidos conceitos que fundamentam esta pesquisa e pontos semelhantes vistos em trabalhos anteriores que utilizam IA focada na educação; a Seção 3 apresenta a Metodologia, detalhando os métodos, técnicas e processos utilizados para avaliação dos planos de ensino; na Seção 4 serão abordados os Resultados e Discussões referentes aos LLMs analisados; e por fim na Seção 5 a Conclusão onde será abordado como novas pesquisas podem abordar o tema e as principais contribuições geradas por este trabalho.

2 REFERENCIAL TEÓRICO

Esta seção pretende realizar a fundamentação teórica responsável por contextualizar conceitos relacionados à temática deste trabalho como os estilos de aprendizagem, modelos de LLMs e uma revisão de artigos anteriores e relacionados.

2.1 Estilos de aprendizagem

Para analisar o perfil dos estudantes, é essencial entender o conceito que molda suas ações. Nesse sentido, destaca-se que “os estilos são algo como conclusões a que chegamos sobre a forma como as pessoas agem. São úteis para classificar e analisar comportamentos” (Alonso; Gallego; Honey, 2007, p. 43, tradução do autor). No campo pedagógico, os estilos são geralmente utilizados para o agrupamento de padrões de comportamento, tornando-se ferramentas bastante úteis para classificação e análise.

A teoria dos estilos de aprendizagem apresenta que cada indivíduo possui uma maneira singular e particular de processar e assimilar informações conforme definido por Alonso, Gallego e Honey (2007), tratando os traços cognitivos, afetivos e psicológicos como indicadores que auxiliam e classificam um indivíduo em um ou mais estilos de aprendizagem.

Com base na proposta definida pelos autores, os estilos de aprendizagem se subdividem em:

- **Ativos:** As pessoas que têm predominância no Estilo Ativo envolvem-se plenamente e sem preconceitos em novas experiências. São de mente aberta, nada céticas e enfrentam com entusiasmo as novas tarefas;
- **Reflexivos:** Os reflexivos gostam de considerar as experiências e observá-las a partir de diferentes perspectivas. Reúnem dados, analisando-os cuidadosamente antes de chegar a alguma conclusão;
- **Teóricos:** Os teóricos adaptam e integram as observações em teorias lógicas e complexas. Eles abordam os problemas de forma vertical e escalonada, por etapas lógicas;
- **Pragmáticos:** O ponto forte das pessoas com predominância no Estilo Pragmático é a aplicação prática das ideias. Elas descobrem o lado

positivo das novas ideias e aproveitam a primeira oportunidade para experimentá-las.

A identificação de cada estilo se dá por meio do Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA), desenvolvido por Alonso, Gallego e Honey (2007). O instrumento tem como base oitenta questões, em que cada uma descreve uma ação cotidiana. Ao respondê-lo, o participante deve assinalar as opções que mais se adequam à sua vivência. Cabe ressaltar que todo indivíduo possui características referentes aos quatro estilos de aprendizagem; contudo, o questionário visa identificar qual ou quais se manifesta de forma predominante, indicando a maneira mais eficaz pela qual o discente assimila os conteúdos. A estrutura completa do instrumento, bem como o seu método de pontuação e avaliação, encontra-se detalhada no Anexo A.

2.2 Large Language Models

Os LLMs referem-se a modelos de linguagem *Transformer* (Zhao *et al.*, 2025). Esses modelos de IA foram desenvolvidos para compreender e gerar textos em linguagem natural, permitindo que o modelo consiga aprender e contextualizar associando as palavras de forma eficaz. Estes modelos são caracterizados por uma quantidade elevada de parâmetros que frequentemente ultrapassam a casa dos bilhões e são treinados com grandes volumes de dados textuais. Os LLMs exibem fortes capacidades para compreender a linguagem natural e resolver tarefas complexas (por meio da geração de texto) (Zhao *et al.*, 2025).

Além da geração textual, os LLMs são capazes de realizar tarefas como tradução, resposta a perguntas, resolução de problemas e até mesmo a elaboração de planos de estudo (Chun *et al.*, 2025). Quando aplicados corretamente no ambiente educacional, os LLMs demonstram grande potencial para atuar como assistentes pessoais, auxiliando os estudantes ao longo do processo de aprendizagem, pois são capazes de fornecer suporte, esclarecer dúvidas em tempo real e adaptar o ritmo de ensino às necessidades do discente.

Modelos como GPT-5 e Gemini 3.5 Flash vêm se destacando por suas capacidades avançadas e geração de respostas rápidas. A consolidação global dessas ferramentas é evidenciada pelo *Artificial Intelligence Index Report* da Universidade de Stanford (Maslej *et al.*, 2025), que aponta que o uso de IA generativa saltou de 33% para 71% em apenas um ano. Impulsionadas por esse crescimento expressivo em larga escala, essas tecnologias ganham cada vez mais espaço no ecossistema educacional, sendo aplicadas em cenários reais como assistentes virtuais, tutores personalizados e geradores de conteúdos.

2.3 Trabalhos relacionados

Kanchon et al. (2024) propõem um sistema gerenciador de aprendizagem para aprimorar a aprendizagem personalizada, focando na detecção automática de estilos de aprendizagem e na modificação do conteúdo educacional. Utilizando um dataset próprio com 506 amostras, os autores aplicaram diversas técnicas de Machine Learning, como o modelo blending ensemble, para classificar os estudantes de perfis visual, auditivo ou cinestésico. Cabe destacar que essa categorização pertence a uma vertente teórica distinta da abordada neste estudo, fundamentando-se no modelo VARK, desenvolvido por Fleming e Mills (1992). Baseados nessa classificação, os autores alcançaram uma acurácia de 97.56%. Uma vez identificado o estilo, o sistema emprega modelos de processamento de linguagem natural, como GPT-3, para adaptar o material de estudo, gerando recursos como textos, mapas mentais e resumos em áudio. A abordagem de Kanchon et al. se distingue deste trabalho por focar na classificação baseada em dados de sistemas de gestão de aprendizagem e na modificação de conteúdo existente. Este trabalho, por sua vez, investiga o uso de LLMs de forma geral, para conduzir a geração de planos de estudo personalizados.

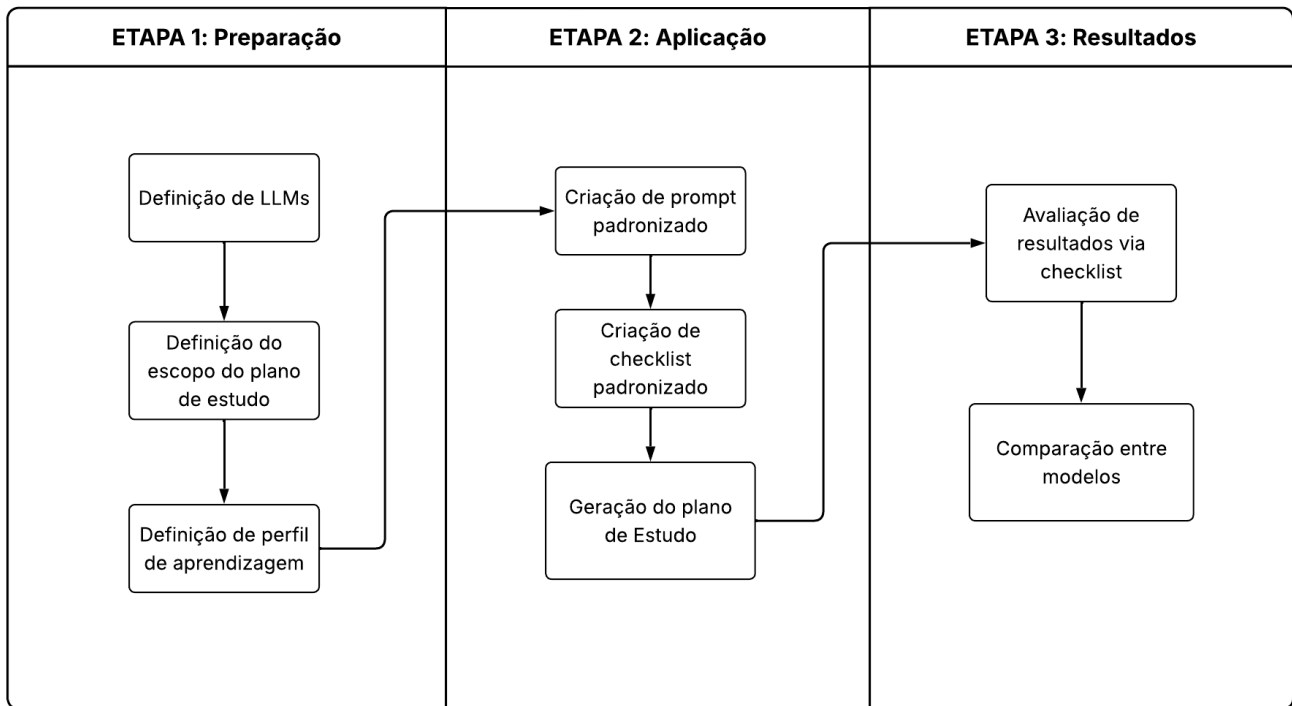
Wang, Lee e Mutlu (2025) exploram de forma semelhante como os LLMs podem potencializar a educação *online* através do sistema LearnMate. A contribuição central do trabalho é a definição de personalização para LLMs, estruturadas em quatro dimensões: metas, tempo, ritmo e caminho. O sistema permite que os usuários configurem suas preferências nessas dimensões para gerar um plano de estudo detalhado e apresentado em um calendário interativo. O LearnMate oferece suporte em tempo real ao analisar o conteúdo de materiais de estudo, como vídeos, para fornecer respostas contextuais às dúvidas dos discentes. A proposta do LearnMate se alinha a este trabalho no que tange à geração de planos de estudo. Contudo, esta pesquisa se diferencia por investigar a capacidade dos LLMs de identificar e se adaptar a modelos pedagógicos específicos de estilos de aprendizagem, como a teoria de Honey-Alonso, e por avaliar comparativamente a eficácia de diferentes LLMs.

3 METODOLOGIA

No contexto deste trabalho, foi definido o seguinte problema de pesquisa: Qual a eficácia de diferentes LLMs na geração de planos de estudo personalizados de acordo com o perfil de aprendizagem CHAEA do estudante? A hipótese central é que os LLMs selecionados possuem capacidade suficiente para gerar estes planos de estudo personalizados de maneira adequada.

Para responder a esse problema e testar a hipótese levantada, o método avaliativo foi estruturado conforme detalhado na Figura 1. O processo subdivide-se em três etapas principais: (i) a Preparação na qual ocorrem as definições e padronizações iniciais, (ii) a Aplicação focada na criação de *prompt*, *checklist* e geração dos planos e (iii) a fase de Resultados destinada à avaliação e comparação dos modelos.

Figura 1 - Fluxograma da metodologia adotada neste trabalho.



Fonte: Elaboração própria (2026).

3.1 Etapa de preparação

A primeira fase do método consistiu na seleção dos LLMs que seriam submetidos à avaliação. A seleção dos modelos baseou-se nos critérios de popularidade e acessibilidade, buscando simular o cenário real de um discente em busca de suporte autônomo para sua aprendizagem. Por esse motivo, optou-se pelas versões gratuitas de duas das principais plataformas do mercado atual: o Gemini na sua versão 3.5 Flash e o GPT-5. Essa escolha justifica-se pela ampla disseminação e gratuidade dessas ferramentas, o que permite avaliar a eficácia de modelos que estão diretamente disponíveis ao público estudantil, sem barreiras financeiras.

Em seguida, definiu-se quais componentes curriculares seriam utilizados no experimento. Foram selecionados dois componentes curriculares do curso de

Análise e Desenvolvimento de Sistemas com naturezas e níveis de complexidade distintos. O primeiro componente, Introdução à Programação, comumente voltada aos fundamentos de lógica, variáveis e funções, representa um componente curricular base, considerado menos complexo para a estruturação de um plano de estudo. O segundo componente, Teste de Software, foi escolhido com o propósito de elevar a dificuldade de geração dos planos de estudo, por se tratar de uma área técnica específica que exige maior aprofundamento.

Embora a prática pedagógica ideal recomende que um mesmo componente curricular aborde múltiplas estratégias para atender à diversidade de estilos dos discentes, para fins de viabilidade e delimitação deste experimento computacional, optou-se por isolar perfis específicos em cada cenário. Assim, a definição dos estilos de aprendizagem foi estruturada com a intenção deliberada de criar um ambiente de teste controlado e desafiador para os modelos avaliados, forçando-os a adaptar o conteúdo ao invés de apenas replicar ementas padronizadas. Para o componente curricular de Introdução à Programação, atribuiu-se como variável de teste o estilo Teórico. Como essa área possui uma natureza inerentemente prática de tentativa e erro, exigir que a IA adapte o ensino para um perfil que prioriza a estruturação lógica e modelos conceituais visa avaliar o potencial dos LLMs em adequar o roteiro de forma estrita ao perfil solicitado.

Por outro lado, para o componente curricular de Teste de Software, selecionou-se como parâmetro o estilo Reflexivo. O objetivo é forçar a ferramenta a ir além da mera execução prática de testes, desenvolvendo roteiros que incentivem o estudante a analisar cenários, ponderar hipóteses e praticar a autoavaliação. Esse delineamento culminará na exigência de que os LLMs estruturem os conteúdos em cronogramas de curto prazo, tipicamente variando entre duas a quatro semanas.

Portanto, a abordagem consistiu em aplicar os LLMs aos dois componentes curriculares de acordo com os perfis definidos. O processo resultou na geração de quatro planos de estudo sob as mesmas diretrizes, permitindo uma comparação direta entre os sistemas.

3.2 Etapa de aplicação

A etapa de aplicação tem como objetivo operacionalizar os parâmetros definidos na fase de preparação. Para isso, o processo foi estruturado em três frentes: a elaboração de um *prompt* (i.e. comando de interação) padronizado, a criação de um *checklist* de validação quantitativa, derivado deste *prompt* e, por fim, a execução prática da geração dos roteiros pelos LLMs selecionados.

3.2.1 Criação de *prompt* padronizado

Para garantir que os modelos recebessem diretrizes precisas e comparáveis, desenvolveu-se um *prompt* padronizado de autoria própria. Este comando foi subdividido em seis parâmetros centrais: Visão Geral, Objetivo, Escopo, Tempo, Perfil de Aprendizagem e Fontes Externas. Vale destacar que os termos delimitados por colchetes [] funcionam como *placeholders*, sendo alterados sistematicamente de acordo com a disciplina e o estilo de aprendizagem avaliados em cada cenário. A estrutura final aplicada aos modelos seguiu o seguinte padrão:

- Visão Geral:
 - Seu papel é desenvolver um plano de estudo que consiga atender aos seguintes requisitos com eficiência: o plano deve possuir objetivos claros, descrever bem as atividades a serem realizadas, incentivar a autonomia, autoavaliação e reflexão do discente, e propor formas avaliativas e revisões. O foco deve ser voltado ao estilo de aprendizagem baseado no Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA), onde o seu diferencial é a aplicação desse perfil na construção do plano, desenvolvendo dinâmicas e tarefas de execução específicas que engajem os traços do perfil solicitado.
- Objetivo:
 - Desenvolver um plano de estudo para [componente curricular ou Conteúdo].
- Escopo:
 - Inserir o que deve estar no conteúdo contido no plano.
- Tempo:
 - O plano deve conter entre 2 a 4 semanas.
- Perfil de Aprendizagem:

- [Aplicar o estilo de aprendizagem baseado na teoria de Honey-Alonso: Ativo, Reflexivo, Teórico ou Pragmático].
- Fontes Externas:
 - Com base no perfil de aprendizagem, o plano deve conter uma quantidade razoável de fontes onde possa basear o estudo como Livros, Vídeos, Músicas e Conteúdos diversos.

3.2.2 Criação de *checklist* padronizado

Com o *prompt* definido, fez-se necessária a criação de um método para validar se as instruções fornecidas foram efetivamente executadas pelos modelos, mitigando o risco de alucinações ou desvios de escopo. A seleção das dez métricas não ocorreu de maneira arbitrária. A construção deste instrumento consistiu em derivar critérios avaliativos diretamente dos parâmetros estabelecidos no *prompt*: Visão Geral, Objetivo, Escopo, Tempo, Perfil de Aprendizagem e Fontes Externas. Desse modo, o questionário foi elaborado para investigar três tópicos fundamentais dos planos gerados:

- Adaptação ao Perfil: atesta se o modelo compreendeu e aplicou corretamente o estilo de aprendizagem exigido, traduzindo-o em instruções e dinâmicas específicas, conforme medido pelos indicadores 1, 2 e 8;
- Rigor técnico e estrutural: verifica se o roteiro respeitou estritamente o escopo do conteúdo solicitado e o horizonte temporal projetado, sem extrapolações, como explícito nos indicadores 3 e 4;
- Elementos de estudo: avalia a presença de componentes essenciais para o acompanhamento do discente, englobando a clareza das metas, a curadoria de fontes externas, o estímulo à autonomia e a estruturação de métodos avaliativos e de revisão, atributos verificados pelos indicadores 5, 6, 7, 9 e 10.

A avaliação adota uma pontuação binária para cada indicador: atribui-se o valor de 1 caso o requisito esteja presente e satisfatório no plano gerado, e 0 caso não tenha sido atendido. Desta forma, o desempenho de cada roteiro gerado resulta em uma nota final que pode atingir a pontuação máxima de 10 pontos. O detalhamento preciso do que se buscou avaliar em cada um dos dez indicadores encontra-se estruturado na Figura 2.

Figura 2 – Checklist de avaliação de eficácia dos LLMs.

Indicadores	Descrição
1. As atividades são adaptadas ao estilo de aprendizagem?	Foco em avaliar se o plano cita e aplica conceitualmente o estilo de aprendizagem em sua construção.
2. Existem instruções específicas relacionadas ao estilo de aprendizagem?	Foco em verificar se há orientações diretas sobre <i>como</i> o estudante deve executar as atividades de acordo com o seu perfil.
3. Não há inclusão de temas que fogem do escopo?	Foco em atestar se o roteiro se mantém estritamente dentro do conteúdo técnico solicitado no <i>prompt</i> , sem desvios.
4. O objetivo do plano foi distribuído adequadamente dentro do tempo estimado?	Foco em verificar se a carga de estudos foi bem fracionada no horizonte temporal proposto (2 a 4 semanas).
5. As metas são claras e bem estruturadas?	Foco em avaliar se as metas propostas são tangíveis, mensuráveis e de fácil compreensão e aplicação.
6. O plano estabelece métodos claros de avaliação de aprendizagem?	Foco em identificar a presença de instrumentos práticos para testar a retenção do que foi estudado.
7. O plano inclui formas de revisar o aprendizado?	Foco em verificar a existência de rotinas ou momentos programados para a fixação do conteúdo abordado.
8. As tarefas propostas apresentam dinâmicas de execução específicas para os traços do perfil solicitado?	Foco em avaliar se o plano consegue estimular o engajamento por meio de dinâmicas práticas voltadas exclusivamente ao perfil do estudante.
9. Há indicação de fontes ou materiais de apoio (links, livros, sites)?	Foco em constatar a inclusão de uma seção dedicada à curadoria de materiais externos complementares.
10. O plano incentiva a autonomia, autoavaliação e reflexão do estudante?	Foco em avaliar se o plano estimula o discente a gerenciar o próprio aprendizado e a aplicar métodos de autoanálise.

Fonte: Elaboração própria (2026).

3.2.3 Geração do plano de estudo

A etapa de aplicação foi concluída com a submissão do *prompt* padronizado aos modelos Gemini 3.5 Flash e GPT-5. Cada modelo produziu dois planos de estudo, um para Introdução à Programação e outro para Teste de Software, ambos com seus respectivos perfis designados anteriormente com base nos estilos de aprendizagem. A consolidação destes quatro documentos possibilitou a aplicação do *checklist* e a subsequente análise comparativa dos resultados.

3.3 Etapa de Resultados

A terceira e última fase da metodologia destina-se a estabelecer como os dados obtidos serão tratados e comparados. Como definido na etapa de aplicação, a avaliação dos roteiros gerados pelos LLMs segue uma abordagem estritamente quantitativa, fundamentada na aplicação do *checklist* estruturado na Figura 2.

Com base no material retornado pelo Gemini 3.5 Flash e pelo GPT-5, o processo de avaliação consistiu na verificação de cada um dos dez indicadores. A atribuição da nota (i.e. 0 para não atendimento e 1 para atendimento do requisito) resultará em uma pontuação final de eficácia para cada plano de estudo, com um limite máximo de 10 pontos. A principal vantagem dessa quantificação é a mitigação da subjetividade analítica, garantindo que ambos os LLMs sejam avaliados pelas mesmas métricas.

Uma vez que os quatro planos de estudo, sendo dois de cada LLM, estejam devidamente pontuados, a fase de comparação ocorrerá por meio do cruzamento direto dessas notas finais. A análise comparativa será estruturada respeitando as combinações elaboradas na etapa de preparação, contrastando o desempenho dos modelos no cenário de Introdução à Programação com perfil cognitivo teórico e no cenário de Teste de Software com perfil cognitivo reflexivo.

Ao contrastar as pontuações finais, será possível atestar de forma objetiva, com base nos dados obtidos na experimentação, qual dos LLMs apresenta maior eficácia na geração de planos de estudo personalizados. Assim, encerra-se o ciclo metodológico, cujos resultados e pontuações obtidas serão integralmente expostos e discutidos na seção subsequente.

4 RESULTADOS

Esta seção apresenta os dados obtidos a partir da aplicação da metodologia e da avaliação dos planos de estudo gerados pelo Gemini 3.5 Flash e pelo GPT-5. O objetivo é demonstrar, de forma quantificável, como cada modelo se comportou perante o que foi solicitado no *prompt*.

Para facilitar a compreensão dos experimentos realizados, esta seção foi dividida em duas. A primeira parte consiste na análise individualizada de cada plano gerado, nas subseções 4.1 a 4.4. Nesta primeira parte, é exposta uma revisão sobre o comportamento de cada modelo, acompanhada das justificativas para a pontuação atribuída a cada indicador dentro do *checklist*. A segunda parte, contida na subseção 4.5, compreende a avaliação geral dos experimentos. Nesta parte é apresentado um quadro comparativo condensado para viabilizar o cruzamento de dados e aferir o desempenho global dos LLMs avaliados.

4.1 Cenário 1: Introdução à Programação – Perfil Teórico utilizando o Gemini 3.5 Flash

Neste primeiro experimento, o *prompt* padronizado foi aplicado ao modelo Gemini 3.5 Flash. O material completo — incluindo o comando de entrada, o plano gerado e o *checklist* avaliativo — encontra-se no Apêndice A.

Na avaliação quantitativa, a ferramenta alcançou 9 pontos. De modo geral, o modelo conseguiu atender bem às exigências, estruturando um roteiro de quatro semanas muito adequado à abstração do perfil teórico. A perda de 1 ponto ocorreu

apenas no Indicador 7, pois o modelo não definiu momentos claros para a revisão do aprendizado. Embora o texto sugira que o aluno conecte conceitos novos aos anteriores, faltou propor uma rotina específica para fixar o conteúdo. Apesar desse pequeno desvio, sob a ótica didática e educacional, o plano do Gemini mostrou-se extremamente contextualizado e detalhado. O modelo se destacou por ir além de tópicos genéricos, indicando livros, vídeos e plataformas de forma nominal, citando os respectivos autores. Essa completude pedagógica é um diferencial muito positivo para o aprendizado autônomo.

4.2 Cenário 2: Introdução à Programação – Perfil Teórico utilizando o GPT-5

No segundo experimento, o mesmo *prompt* foi submetido ao modelo GPT-5. O conjunto dos dados correspondente a este cenário encontra-se detalhado no Apêndice B.

O plano gerado destacou-se pelo nível de detalhe na divisão dos tópicos ao longo das semanas. Comparando com o modelo anterior, nota-se que o GPT-5 fragmentou os assuntos em vários subtópicos práticos, oferecendo uma quantidade maior de sugestões de estudo. Contudo, como consequência dessa divisão extensa, as descrições acabaram ficando curtas e objetivas demais. Do ponto de vista educacional, essa abordagem gerou um roteiro generalizado e superficial, carecendo do aprofundamento didático necessário. Apesar dessa forte limitação qualitativa na construção do conhecimento, a estrutura atendeu aos parâmetros literais e binários do *checklist*, atingindo nota máxima na avaliação quantitativa.

4.3 Cenário 3: Teste de Software – Perfil Reflexivo utilizando o Gemini 3.5 Flash

Avançando para o segundo componente curricular, focado em Teste de Software aliado ao perfil Reflexivo, o *prompt* adaptado foi submetido ao Gemini 3.5 Flash. O relatório completo da interação está disponível no Apêndice C.

A avaliação por meio do *checklist* demonstrou um excelente desempenho da ferramenta para este contexto, alcançando os 10 pontos previstos. O modelo compreendeu bem o cenário e, diferente do que ocorreu no primeiro plano com perfil teórico, cumpriu rigorosamente a etapa de revisão. O roteiro sugeriu a criação de um Rito de Revisão Semanal ao final de cada ciclo de estudos para ajudar o discente a consolidar o conhecimento. Essa prática atendeu perfeitamente à necessidade de análise e ponderação que caracteriza o estudante reflexivo. Neste cenário, o Gemini provou a sua consistência educacional ao manter o alto nível de detalhamento didático observado no teste anterior. A ferramenta continuou oferecendo uma curadoria rica de fontes externas e contextualizou profundamente as práticas exigidas, resultando em um plano de estudo completo e pedagogicamente maduro.

4.4 Cenário 4: Teste de Software – Perfil Reflexivo utilizando o GPT-5

Fechando a etapa de avaliação individual, o *prompt* referente ao componente curricular Teste de Software foi processado pelo modelo GPT-5, cujos dados estão detalhados no Apêndice D.

O modelo manteve o seu bom desempenho e conquistou novamente a nota máxima de 10 pontos. Avaliando as respostas da ferramenta nos dois testes, foi possível identificar um padrão: em ambos, o GPT-5 iniciou o texto explicando as características do próprio estilo de aprendizagem antes de apresentar o cronograma. Além disso, novamente subdividiu bastante os tópicos semanais. No entanto, assim como observado no primeiro cenário, essa estratégia de criar várias subtarefas resultou em um plano de estudos muito curto e direto. Sob a ótica educacional, faltou um aprofundamento real nas orientações. O modelo confirmou sua competência em cumprir rigorosamente as regras estruturais do *prompt*, mas produziu um material mais generalista e com menor riqueza pedagógica quando comparado ao seu concorrente.

4.5 Avaliação Geral

Avaliando o cenário de forma global, percebe-se que as ferramentas conseguiram atender com clareza às exigências do *prompt*. Os resultados de todas as métricas foram reunidos na Tabela 1 e na Tabela 2, o que torna mais simples visualizar e comparar como cada modelo se saiu nos testes.

A análise conjunta dos dados evidencia que os dois modelos executaram a tarefa com grande eficácia. Observa-se que eles não apenas seguiram o escopo dos conteúdos e a duração dos planos conforme exigido, mas também conseguiram aplicar de forma coerente os perfis de aprendizagem do CHAEA. Essa capacidade de adaptação demonstra que essas ferramentas podem ter um valor prático significativo para o ambiente educacional, podendo atuar no auxílio à criação de trilhas de estudo sob medida para os alunos.

Tabela 1 – Desempenho geral dos LLMs por plano de estudo.

Modelo	Disciplina	Perfil de Aprendizagem	Nota Final
Gemini	Introdução à Programação	Teórico	9
ChatGPT	Introdução à Programação	Teórico	10
Gemini	Testes de Software	Reflexivo	10
ChatGPT	Testes de Software	Reflexivo	10

Fonte: Elaboração própria (2026).

Tabela 2 – Pontuação detalhada dos modelos por indicador do *checklist*.

Indicador	Nota Gemini IP	Nota ChatGPT IP	Nota Gemini TS	Nota ChatGPT TS
1. As atividades são adaptadas ao estilo de aprendizagem?	1	1	1	1
2. Existem instruções específicas relacionadas ao estilo de aprendizagem?	1	1	1	1
3. Não há inclusão de temas que fogem do escopo?	1	1	1	1
4. O objetivo do plano foi distribuído adequadamente dentro do tempo estimado?	1	1	1	1
5. As metas são claras e bem estruturadas?	1	1	1	1
6.O plano estabelece métodos claros de avaliação de aprendizagem?	1	1	1	1
7. O plano inclui formas de revisar o aprendido?	0	1	1	1
8. As tarefas propostas apresentam dinâmicas de execução específicas para os traços do perfil solicitado?	1	1	1	1

9. Há indicação de fontes ou materiais de apoio (links, livros, sites)?	1	1	1	1
10. O plano incentiva a autonomia, autoavaliação e reflexão do estudante?	1	1	1	1

Fonte: Elaboração própria (2026).

Além da verificação quantitativa, é fundamental ressaltar o contraste qualitativo e teórico das estratégias elaboradas pelas IAs. As abordagens geradas para o estilo Teórico diferenciam-se por focar estritamente na lógica estrutural do conceito geral para o código. Nesses cenários, os modelos exigiram a criação de fluxogramas, pseudocódigos e o entendimento da arquitetura do software antes da execução prática, alinhando-se à necessidade de racionalidade e ordenação desse perfil. Em contrapartida, as estratégias delineadas para o estilo Reflexivo seguiram uma vertente fundamentada na observação, coleta de dados e ponderação. Em vez de construir modelos matemáticos rígidos, as atividades exigiam que o estudante analisasse falhas reais de sistemas já existentes, fizesse revisões de especificações e registrasse suas impressões em 'Diários de Aprendizagem Reflexivos' antes de tomar qualquer decisão. Essa distinção evidencia que as ferramentas não apenas geraram tópicos de estudo, mas conseguiram emular, com sucesso, o núcleo metodológico de cada perfil cognitivo delineado por Alonso, Gallego e Honey (2007).

5 CONCLUSÃO

Este trabalho teve como objetivo investigar a eficácia dos LLMs, especificamente o Gemini 3.5 Flash e o GPT-5, na geração de planos de estudo personalizados. A pesquisa fundamentou-se na teoria de estilos de aprendizagem CHAEA, com foco nas abordagens Teórica e Reflexiva. Os resultados revelaram que ambas as ferramentas apresentaram consistência e alta proficiência técnica, alcançando notas mínimas de 9, em um total de 10, na avaliação baseada no *checklist* estruturado. Tais métricas atestam a viabilidade e a capacidade destes LLMs na criação de roteiros instrucionais ricos em conteúdo e adequados às particularidades cognitivas dos estudantes.

Apesar da capacidade técnica observada na estruturação dos planos avaliados neste trabalho, ressalta-se que os modelos atuam essencialmente como ferramentas de suporte. Os roteiros gerados oferecem uma base para direcionar a trilha de aprendizagem; contudo, a utilização de IA generativa não substitui a necessidade de mediação pedagógica. A supervisão e a orientação de um docente continuam sendo indispensáveis para validar o conteúdo, garantir o alinhamento com os objetivos curriculares e conduzir o acompanhamento e o planejamento final dos componentes curriculares.

Entre as limitações deste estudo, destaca-se o recorte metodológico aplicado aos estilos de aprendizagem. A pesquisa avaliou apenas as categorias Teórica e Reflexiva. Essa delimitação ocorreu de forma intencional para viabilizar a execução quantitativa deste estudo exploratório, gerando um contraste analítico rigoroso na avaliação do desempenho das ferramentas, sem expandir de forma impraticável o escopo de testes e cruzamentos de dados. Dessa forma, as variáveis de perfis Ativo e Pragmático foram deixadas de fora do roteiro de validação, abrindo espaço para investigações futuras.

Mesmo com as limitações apontadas, este estudo oferece uma base exploratória inicial que pode inspirar novas investigações. Um dos principais caminhos a seguir é utilizar esses resultados para fundamentar a criação de um modelo de LLM especializado, focado exclusivamente em produzir planos de estudo guiados pelos estilos de aprendizagem. Adicionalmente, sugere-se a introdução de

um terceiro LLM para atuar como agente mediador, responsável por automatizar a aplicação do checklist e avaliar a qualidade dos roteiros gerados de forma imparcial e escalável. Além da simples estruturação de roteiros, será fundamental que trabalhos futuros testem a viabilidade prática dessas ferramentas como tutores virtuais. Dessa forma, seria possível avaliar a capacidade da IA de oferecer um acompanhamento contínuo e personalizado, adaptando-se a diferentes componentes curriculares e às necessidades diárias de cada discente.

REFERÊNCIAS

ALONSO, Catalina M.; GALLEGGO, Domingo J.; HONEY, Peter. **Los estilos de aprendizaje**: procedimientos de diagnóstico y mejora. 7. ed. Bilbao: Mensajero, 2007.

BACICH, Lilian; TANZI NETO, Adolfo; TREVISANI, Fernando de Mello (Org.). **Ensino híbrido**: personalização e tecnologia na educação. Porto Alegre: Penso, 2015.

CHUN, Jiwon et al. PlanGlow: Personalized Study Planning with an Explainable and Controllable LLM-Driven System. In: ACM CONFERENCE ON LEARNING @ SCALE, 12., 2025, Palermo, Itália. **Proceedings of the Twelfth ACM Conference on Learning @ Scale**. New York: ACM, 2025. p. 1–12. DOI: 10.1145/3698205.3729541. Acesso em: 17 jul. 2026.

FLEMING, Neil D.; MILLS, Colleen. Not another inventory, rather a catalyst for reflection. **To improve the academy**, v. 11, n. 1, p. 137-155, 1992.

KANCHON, Md. Kabin Hasan et al. Enhancing personalized learning: AI-driven identification of learning styles and content modification strategies. **International Journal of Cognitive Computing in Engineering**, [S. l.], v. 5, p. 269–278, 2024. DOI: 10.1016/j.ijcce.2024.06.002. Acesso em: 17 jul. 2026.

MASLEJ, Nestor et al. **The AI Index 2025 Annual Report**. Stanford: Stanford University, Institute for Human-Centered Artificial Intelligence (HAI), 2025. Disponível em: <https://hai.stanford.edu/ai-index/2025-ai-index-report>. Acesso em: 17 jul. 2026.

MIRANDA, Luísa; MORAIS, Carlos. Estilos de aprendizagem: O questionário CHAEA adaptado para a língua portuguesa. **Revista de Estilos de Aprendizaje**, v. 1, n. 1, p. 66-87, abr. 2008.

RUSSELL, Stuart J.; NORVIG, Peter. **Inteligência Artificial**: uma abordagem moderna. 4. ed. Rio de Janeiro: GEN LTC, 2022.

UNESCO. **Relatório Global de Monitoramento da Educação 2023**: A tecnologia na educação: Uma ferramenta a serviço de quem?. Paris: UNESCO, 2023.

WANG, Xinyu Jessica; LEE, Christine P.; MUTLU, Bilge. LearnMate: Enhancing Online Education with LLM-Powered Personalized Learning Plans and Support. In: CHI CONFERENCE ON HUMAN FACTORS IN COMPUTING SYSTEMS, 2025, Yokohama, Japão. **Extended Abstracts of the CHI Conference on Human Factors in Computing Systems**. New York: ACM, 2025. p. 1–10. DOI: 10.1145/3706599.3719857. Acesso em: 17 jul. 2026.

ZHAO, Wayne Xin; ZHOU, Kun; LI, Junyi; TANG, Tianyi; WANG, Xiaolei; HOU, Yupeng; MIN, Yingqian; ZHANG, Beichen; ZHANG, Junjie; DONG, Zican; et al. **A Survey of Large Language Models**. arXiv, 31 mar. 2023 (v. 1); última versão v. 16, 11 mar. 2025. arXiv:2303.18223 [cs.CL; cs.AI]. Disponível em: <https://arxiv.org/abs/2303.18223>. Acesso em: 17 jul. 2026.

Apêndice A - Introdução à Programação, Modelo Gemini, Perfil Teórico

Documento de Avaliação

LLM: Gemini 3.5 Flash.

componente curricular: Introdução à Programação

Perfil de Aprendizagem: Teórico.

O documento conta com a seguinte estrutura:

- *Prompt*;
- *Output*;
- Avaliação.

Prompt

Visão Geral: Seu papel é desenvolver um plano de estudo que consiga atender aos seguintes requisitos com eficiência: o plano deve possuir objetivos claros, descrever bem as atividades a serem realizadas, incentivar a autonomia, autoavaliação e reflexão do discente, e propor formas avaliativas e revisões. O foco deve ser voltado ao estilo de aprendizagem baseado no Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA), onde o seu diferencial é a aplicação desse perfil na construção do plano, desenvolvendo dinâmicas e tarefas de execução específicas que engajem os traços do perfil solicitado.

Objetivo: Desenvolver um plano de estudo para Introdução à Programação.

Escopo: O plano deve abordar o conteúdo inicial de lógica de programação até funções.

Tempo: O plano deve conter entre 2 a 4 semanas.

Perfil de Aprendizagem: Aplicar o estilo de aprendizagem baseado na teoria de Honey-Alonso: Teórico.

Fontes Externas: Com base no perfil de aprendizagem, o plano deve conter uma quantidade razoável de fontes onde possa basear o estudo como Livros, Vídeos, Músicas e Conteúdos diversos.

Output

Este plano de estudo foi desenvolvido sob medida para o perfil **teórico** do questionário CHAEA. Estudantes com essa predominância intelectual tendem a ser analíticos, sintéticos, profundos e focados na lógica e na racionalidade. Eles aprendem melhor quando os conteúdos são apresentados a partir de teorias sólidas, modelos, estruturas e sistemas lógicos, e quando podem questionar o "porquê" das coisas.

Para engajar esse perfil, o plano evita o aprendizado puramente por tentativa e erro (estilo Ativo) e foca na **compreensão dos fundamentos matemáticos e estruturais** antes da escrita do código.

Plano de Estudo: Fundamentos de Programação (Perfil Teórico)

Duração: 4 semanas

Objetivo Geral: Compreender a lógica estrutural da programação, desde o pensamento computacional até a modularização com funções, construindo uma base conceitual sólida.



Metodologia e Dinâmicas de Estudo para o Teórico

- **Abordagem Top-Down:** Primeiro entende-se o sistema e a regra geral; depois se aplica ao código.
- **Fichamento e Modelagem:** Em vez de apenas digitar código, você será incentivado a criar diagramas de blocos, pseudocódigos e mapas conceituais.
- **Autoavaliação Crítica:** Testes baseados em lógica e consistência para validar se o conhecimento faz sentido dentro de um sistema maior.



Cronograma Semanal

Semana 1: Pensamento Computacional e Lógica de Programação

- **Objetivo:** Compreender os axiomas da lógica e como os computadores processam instruções sequenciais.
- **Atividades Práticas:**
 1. Leitura analítica sobre a história da lógica binária e arquitetura de Von Neumann.
 2. Criação de fluxogramas detalhados para a resolução de problemas cotidianos (ex: o algoritmo para trocar um pneu ou calcular a média de notas).
- **Dinâmica Teórica:** Escrever o pseudocódigo de um problema antes de conhecer qualquer linguagem de programação. O foco é a sintaxe mental.

Semana 2: Variáveis, Tipos de Dados e Estruturas Condicionais

- **Objetivo:** Entender como a memória armazena informações e como o fluxo de um programa se ramifica logicamente.
- **Atividades Práticas:**
 1. Mapeamento de tipos de dados: criar uma tabela comparativa sobre o espaço de memória e o comportamento de inteiros, floats, strings e booleanos.
 2. Resolução de tabelas-verdade aplicadas a estruturas **IF/ELSE** (Se/Senão).
- **Dinâmica Teórica:** "Teste de Mesa". Pegar um código pronto com condicionais e rastrear o valor de cada variável a cada linha, usando papel e caneta, sem executar no computador.

Semana 3: Estruturas de Repetição (Loops)

- **Objetivo:** Compreender a teoria da iteração e a otimização de processos repetitivos.
- **Atividades Práticas:**
 1. Análise comparativa: Quando utilizar **WHILE** (enquanto) versus **FOR** (para)? Desenhar um diagrama de decisão para a escolha do loop correto.
 2. Identificação de loops infinitos: Analisar códigos quebrados para deduzir teoricamente por que o critério de parada falhou.
- **Dinâmica Teórica:** Dedução matemática. Calcular manualmente quantas vezes um loop irá rodar com base em sua expressão de controle antes de rodar o script.

Semana 4: Funções e Modularização

- **Objetivo:** Dominar o princípio da abstração, escopo de variáveis e reaproveitamento de código.
- **Atividades Práticas:**
 1. Decomposição de problemas: Pegar um problema complexo e dividi-lo em subfunções teóricas (conceito de "dividir para conquistar").
 2. Estudo de escopo: Criar um mapa mental que explique visualmente a diferença entre variáveis locais e globais.
- **Dinâmica Teórica:** Definição de contratos de função. Escrever a documentação (Inputs, Outputs e Efeitos Colaterais) de uma função antes mesmo de codificá-la.



Fontes Externas Recomendadas

Para alimentar a necessidade de profundidade e contextualização do perfil teórico, utilize o cardápio de recursos abaixo:



Livros (Base Conceitual)

- *Algoritmos: Teoria e Prática* (Thomas H. Cormen) – O "guia sagrado" dos algoritmos. Foque nos capítulos iniciais de introdução ao pensamento matemático.
- *Introdução à Programação com Python* (Nilo Ney Coutinho Menezes) – Excelente didática linear, explicando o *porquê* de cada recurso da linguagem.



Vídeos e Cursos (Estrutura Completa)

- *Curso em Vídeo: Algoritmos* (Gustavo Guanabara) – Uma base teórica impecável com uso de pseudocódigo (Portugal) que ajuda a separar a lógica da sintaxe de uma linguagem complexa.
- *Harvard CS50 (Introdução à Ciência da Computação)* – Disponível gratuitamente no YouTube/edX. Ideal para teóricos, pois começa explicando a nível de hardware (transistores, binários) e sobe para o software.

🎵 Músicas (Foco Cognitivo)

O teórico precisa afastar distrações externas para focar em conexões lógicas complexas.

- *Música Clássica (Bach e Vivaldi)*: A estrutura matemática das composições barrocas ajuda na organização do raciocínio lógico.
- *Synthwave / Chillstep Instrumental*: Ritmos repetitivos e sem letras, excelentes para longas sessões de leitura de documentação técnica.

🌐 Conteúdos Diversos

- *Visualgo.net*: Uma ferramenta interativa que permite visualizar o comportamento teórico de algoritmos passo a passo.
- *Documentação Oficial da Linguagem* (ex: *Python.org* ou *MDN Web Docs*): Teóricos adoram ir direto à fonte da verdade técnica em vez de apenas tutoriais mastigados.



Autonomia, Reflexão e Avaliação

Para garantir que você está retendo o conhecimento de forma crítica e autônoma, adote os seguintes mecanismos:

1. Diário de Bordo Cognitivo (Reflexão)

Ao final de cada semana, responda a estas três perguntas em um caderno ou bloco de notas:

- *O conceito estudado nesta semana se conecta de que forma com o que aprendi na semana anterior?*
- *Qual é a regra geral por trás do erro de sintaxe que eu mais cometi nesses dias?*
- *Se eu tivesse que explicar essa estrutura para um computador usando lógica pura, onde estariam os pontos fracos da minha explicação?*

2. Forma Avaliativa: O Projeto Arquitetural (Semana 4)

Em vez de uma prova tradicional de marcar x, sua avaliação final será a **defesa de um projeto**.

- **Tarefa**: Desenhe a arquitetura de um pequeno sistema (ex: um sistema de gerenciamento de biblioteca ou uma calculadora financeira).
- **O que entregar**: Você não entregará apenas o código funcionando. Deve entregar um documento contendo:
 1. O fluxograma do sistema.

2. A justificativa do porquê escolheu determinada estrutura de repetição ou condicional.
3. O mapa de funções mostrando como os dados entram e saem de cada módulo.

Se o seu modelo teórico estiver coerente e sem falhas lógicas, o seu código refletirá essa solidez com facilidade.

Avaliação

Indicadores	Descrição	Nota
1. As atividades são adaptadas ao estilo de aprendizagem?	Foco em avaliar se o plano cita e aplica conceitualmente o estilo de aprendizagem em sua construção.	1
2. Existem instruções específicas relacionadas ao estilo de aprendizagem?	Foco em verificar se há orientações diretas sobre <i>como</i> o estudante deve executar as atividades de acordo com o seu perfil.	1
3. Não há inclusão de temas que fogem do escopo?	Foco em atestar se o roteiro se mantém estritamente dentro do conteúdo técnico solicitado no <i>prompt</i> , sem desvios.	1
4. O objetivo do plano foi distribuído adequadamente dentro do tempo estimado?	Foco em verificar se a carga de estudos foi bem fracionada no horizonte temporal proposto (2 a 4 semanas).	1
5. As metas são claras e bem estruturadas?	Foco em avaliar se as metas propostas são tangíveis, mensuráveis e de fácil compreensão e aplicação.	1
6. O plano estabelece métodos claros de avaliação de aprendizagem?	Foco em identificar a presença de instrumentos práticos para testar a retenção do que foi estudado.	1

7. O plano inclui formas de revisar o aprendizado?	Foco em verificar a existência de rotinas ou momentos programados para a fixação do conteúdo abordado.	0
8. As tarefas propostas apresentam dinâmicas de execução específicas para os traços do perfil solicitado?	Foco em avaliar se o plano consegue estimular o engajamento por meio de dinâmicas práticas voltadas exclusivamente ao perfil do estudante.	1
9. Há indicação de fontes ou materiais de apoio (links, livros, sites)?	Foco em constatar a inclusão de uma seção dedicada à curadoria de materiais externos complementares.	1
10. O plano incentiva a autonomia, autoavaliação e reflexão do estudante?	Foco em avaliar se o plano estimula o discente a gerenciar o próprio aprendizado e a aplicar métodos de autoanálise.	1

Nota Final: 9

Apêndice B - Introdução à Programação, Modelo GPT, Perfil Teórico

Documento de Avaliação

LLM: Chat GPT 5.5.

componente curricular: Introdução à Programação

Perfil de Aprendizagem: Teórico.

O documento conta com a seguinte estrutura:

- *Prompt*;
- *Output*;
- *Avaliação*.

Prompt

Visão Geral: Seu papel é desenvolver um plano de estudo que consiga atender aos seguintes requisitos com eficiência: o plano deve possuir objetivos claros, descrever bem as atividades a serem realizadas, incentivar a autonomia, autoavaliação e reflexão do discente, e propor formas avaliativas e revisões. O foco deve ser voltado ao estilo de aprendizagem baseado no Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA), onde o seu diferencial é a aplicação desse perfil na construção do plano, desenvolvendo dinâmicas e tarefas de execução específicas que engajem os traços do perfil solicitado.

Objetivo: Desenvolver um plano de estudo para Introdução à Programação.

Escopo: O plano deve abordar o conteúdo inicial de lógica de programação até funções.

Tempo: O plano deve conter entre 2 a 4 semanas.

Perfil de Aprendizagem: Aplicar o estilo de aprendizagem baseado na teoria de Honey-Alonso: Teórico.

Fontes Externas: Com base no perfil de aprendizagem, o plano deve conter uma quantidade razoável de fontes onde possa basear o estudo como Livros, Vídeos, Músicas e Conteúdos diversos.

Output

Plano de Estudo — Introdução à Programação

Perfil de Aprendizagem CHAEA: Teórico

Duração: **4 semanas**

Foco: **Lógica de Programação até Funções**

1. Características do Perfil Teórico (Honey-Alonso)

O perfil **Teórico** aprende melhor quando:

- Existe organização e sequência lógica;
- O conteúdo possui fundamentação conceitual;
- Há modelos, regras e estruturas bem definidas;
- O estudante consegue analisar, comparar e sistematizar informações;
- As atividades estimulam raciocínio lógico e reflexão crítica.

Por isso, este plano prioriza:

- estudo estruturado;
 - mapas conceituais;
 - resolução analítica de problemas;
 - explicações detalhadas;
 - revisões sistemáticas;
 - autoavaliações reflexivas;
 - construção gradual do conhecimento.
-

2. Objetivo Geral

Desenvolver a compreensão dos fundamentos da programação, capacitando o estudante a:

- compreender lógica computacional;
- elaborar algoritmos;
- utilizar estruturas condicionais e de repetição;
- manipular variáveis e estruturas básicas;
- construir funções simples;
- analisar problemas de maneira lógica e estruturada.

3. Organização do Plano

Se mana	Conteúdo Principal	Objetivo
1	Lógica de programação e algoritmos	Desenvolver raciocínio lógico
2	Variáveis, operadores e condicionais	Criar programas com tomada de decisão
3	Estruturas de repetição	Automatizar processos repetitivos
4	Funções e revisão geral	Modularizar e organizar código

SEMANA 1 — Lógica de Programação e Algoritmos

Objetivos da Semana

- Entender o conceito de algoritmo;
 - Desenvolver pensamento lógico;
 - Interpretar problemas computacionais;
 - Criar fluxos de solução.
-

Conteúdos

- O que é programação;
 - Algoritmos;
 - Fluxogramas;
 - Pseudocódigo;
 - Sequência lógica;
 - Entrada, processamento e saída.
-

Atividades

1. Estudo Teórico Estruturado

Ler sobre:

- conceito de algoritmo;
- representação lógica;
- etapas de resolução de problemas.

2. Construção de Mapas Conceituais

Criar mapas contendo:

- algoritmo;
- lógica;
- processamento;
- entrada e saída de dados.

3. Exercícios Analíticos

Resolver problemas como:

- cálculo de média;
- conversão de temperatura;
- cálculo de idade;
- cálculo de salário.

4. Reflexão Escrita

Responder:

- “Qual foi a lógica utilizada para resolver o problema?”
 - “Existe uma solução mais eficiente?”
-

Estratégias Compatíveis com Perfil Teórico

- Organização em esquemas;
 - Comparação entre diferentes algoritmos;
 - Justificativa lógica das respostas;
 - Uso de tabelas e fluxos.
-

Autoavaliação

Responder ao final da semana:

Pergunta	Sim	Parcial	Não
Consigo explicar o que é um algoritmo?			
Consigo criar pseudocódigo simples?			
Entendi a lógica de entrada e saída?			

Revisão

- Refazer os exercícios sem consultar material;
 - Criar um resumo conceitual de 1 página.
-

SEMANA 2 — Variáveis, Operadores e Condicionais

Objetivos da Semana

- Manipular dados;
 - Entender tipos de variáveis;
 - Utilizar operadores;
 - Desenvolver lógica condicional.
-

Conteúdos

- Variáveis;
 - Tipos de dados;
 - Operadores matemáticos;
 - Operadores lógicos;
 - Estruturas condicionais (**if**, **else**).
-

Atividades

1. Quadro Comparativo

Construir tabelas contendo:

- tipos de variáveis;
- operadores;
- exemplos de uso;
- diferenças entre operadores lógicos.

2. Exercícios Progressivos

Resolver:

- calculadora simples;
- sistema de aprovação escolar;
- verificação de número par/ímpar;
- classificação de idade.

3. Análise de Código

Observar pequenos códigos e responder:

- qual o objetivo do algoritmo;
- quais variáveis foram usadas;
- quais condições estão sendo verificadas.

4. Diário de Aprendizagem

Escrever:

- principais dificuldades;
 - conceitos mais importantes;
 - padrões percebidos nos exercícios.
-

Estratégias Compatíveis com Perfil Teórico

- Classificação sistemática;
 - Relação entre teoria e aplicação;
 - Comparação entre estruturas condicionais;
 - Explicação detalhada das decisões lógicas.
-

Autoavaliação

Pergunta	Sim	Parcial	Não
Consigo usar variáveis corretamente?			
Entendi operadores lógicos?			
Consigo criar condições simples?			

Revisão

- Criar um resumo dos operadores;
 - Resolver novamente os exercícios com tempo reduzido.
-

SEMANA 3 — Estruturas de Repetição

Objetivos da Semana

- Automatizar tarefas;
 - Compreender repetição lógica;
 - Desenvolver algoritmos iterativos.
-

Conteúdos

- Laços de repetição;
 - `while`;
 - `for`;
 - contadores;
 - acumuladores.
-

Atividades

1. Estudo Comparativo

Comparar:

- `for` vs `while`;
- vantagens;
- aplicações ideais.

2. Exercícios

Criar programas para:

- tabuada;
- soma de números;
- contagem regressiva;
- cálculo fatorial;
- média de várias notas.

3. Análise Estrutural

Responder:

- quantas vezes o laço executa;
- condição de parada;
- possíveis erros lógicos.

4. Desafio Reflexivo

Explicar:

- “Por que estruturas de repetição reduzem trabalho?”
 - “Qual estrutura é mais adequada em cada situação?”
-

Estratégias Compatíveis com Perfil Teórico

- Análise de padrões;
 - Estruturas sistemáticas;
 - Identificação de regras;
 - Comparação entre métodos.
-

Autoavaliação

Pergunta

imarcial P
ão

Sei quando usar `for` ou `while`?

Consigo criar repetições corretamente?

Entendo condição de parada?

Revisão

- Criar um mapa mental dos laços;
 - Resolver exercícios sem consultar exemplos.
-

SEMANA 4 — Funções e Revisão Geral

Objetivos da Semana

- Modularizar programas;
 - Reutilizar código;
 - Organizar soluções lógicas.
-

Conteúdos

- Conceito de função;
 - Parâmetros;
 - Retorno;
 - Escopo básico;
 - Organização de código.
-

Atividades

1. Estudo Conceitual

Ler sobre:

- modularização;

- reutilização;
- vantagens das funções.

2. Exercícios

Criar funções para:

- soma;
- média;
- verificação de número primo;
- cálculo de IMC;
- conversão de moedas.

3. Refatoração

Transformar exercícios anteriores em funções reutilizáveis.

4. Projeto Final

Criar um pequeno sistema contendo:

- entrada de dados;
- condicionais;
- repetição;
- funções.

Exemplos:

- sistema escolar;
 - calculadora;
 - cadastro simples;
 - menu interativo.
-

Estratégias Compatíveis com Perfil Teórico

- Organização modular;
 - Estruturação lógica;
 - Análise de eficiência;
 - Relação entre partes do sistema.
-

Autoavaliação Final

Critério	Ex celente	om	Precisa Melhorar
Compreensão lógica			
Uso de condicionais			
Uso de repetições			
Criação de funções			
Organização do código			

4. Métodos Avaliativos

Avaliação Diagnóstica

- Exercícios iniciais de lógica.

Avaliação Formativa

- Exercícios semanais;
- mapas conceituais;
- reflexões escritas;
- análises de código.

Avaliação Somativa

- Projeto final integrador.
-

5. Estratégias de Autonomia e Reflexão

O plano incentiva autonomia através de:

- diário de aprendizagem;
- revisões independentes;

- autoavaliações semanais;
 - comparação entre soluções;
 - pesquisa complementar;
 - análise crítica de algoritmos.
-

6. Fontes de Estudo Recomendadas

Livros

Algoritmos: Lógica para Desenvolvimento de Programação

Excelente para base lógica e estruturação teórica.

Lógica de Programação e Algoritmos

Muito indicado para iniciantes.

Entendendo Algoritmos

Didática visual e organizada.

Clean Code

Para organização e estrutura lógica de código.

Vídeos e Cursos

- Curso em Vídeo — Gustavo Guanabara
 - FreeCodeCamp
 - CS50 Harvard
 - Programação Dinâmica (YouTube)
-

Sites

- W3Schools
- GeeksforGeeks
- HackerRank
- Beecrowd
- MDN Web Docs

Plataformas de Exercícios

- Beecrowd;
 - HackerRank;
 - Codewars.
-

Podcasts e Conteúdos Complementares

- Hipsters.tech;
 - Código Fonte TV;
 - Filipe Deschamps.
-

Músicas para Concentração

Lo-fi Beats

Ajuda na concentração durante exercícios lógicos.

Peaceful Piano

Boa para leitura teórica e revisão.

7. Resultado Esperado

Ao final do plano, o estudante deverá ser capaz de:

- interpretar problemas computacionais;
- desenvolver algoritmos;
- escrever programas básicos;
- utilizar estruturas fundamentais da programação;
- criar funções organizadas;
- analisar soluções logicamente.

Além disso, o plano fortalece:

- autonomia intelectual;
- pensamento analítico;

- organização cognitiva;
- reflexão crítica;
- sistematização do conhecimento.

Avaliação

Indicadores	Descrição	Nota
1. As atividades são adaptadas ao estilo de aprendizagem?	Foco em avaliar se o plano cita e aplica conceitualmente o estilo de aprendizagem em sua construção.	1
2. Existem instruções específicas relacionadas ao estilo de aprendizagem?	Foco em verificar se há orientações diretas sobre <i>como</i> o estudante deve executar as atividades de acordo com o seu perfil.	1
3. Não há inclusão de temas que fogem do escopo?	Foco em atestar se o roteiro se mantém estritamente dentro do conteúdo técnico solicitado no <i>prompt</i> , sem desvios.	1
4. O objetivo do plano foi distribuído adequadamente dentro do tempo estimado?	Foco em verificar se a carga de estudos foi bem fracionada no horizonte temporal proposto (2 a 4 semanas).	1
5. As metas são claras e bem estruturadas?	Foco em avaliar se as metas propostas são tangíveis, mensuráveis e de fácil compreensão e aplicação.	1
6. O plano estabelece métodos claros de avaliação de aprendizagem?	Foco em identificar a presença de instrumentos práticos para testar a retenção do que foi estudado.	1
7. O plano inclui formas de revisar o aprendizado?	Foco em verificar a existência de rotinas ou momentos programados para a fixação do conteúdo abordado.	1

8. As tarefas propostas apresentam dinâmicas de execução específicas para os traços do perfil solicitado?	Foco em avaliar se o plano consegue estimular o engajamento por meio de dinâmicas práticas voltadas exclusivamente ao perfil do estudante.	1
9. Há indicação de fontes ou materiais de apoio (links, livros, sites)?	Foco em constatar a inclusão de uma seção dedicada à curadoria de materiais externos complementares.	1
10. O plano incentiva a autonomia, autoavaliação e reflexão do estudante?	Foco em avaliar se o plano estimula o discente a gerenciar o próprio aprendizado e a aplicar métodos de autoanálise.	1

Nota Final: 10

Apêndice C - Teste de Software, Modelo Gemini, Perfil Reflexivo

Documento de Avaliação

LLM: Gemini 3.5 Flash.
componente curricular: Teste de Software
Perfil de Aprendizagem: Reflexivo.

O documento conta com a seguinte estrutura:

- *Prompt*;
- *Output*;
- *Avaliação*.

Prompt

Visão Geral: Seu papel é desenvolver um plano de estudo que consiga atender aos seguintes requisitos com eficiência: o plano deve possuir objetivos claros, descrever bem as atividades a serem realizadas, incentivar a autonomia, autoavaliação e reflexão do discente, e propor formas avaliativas e revisões. O foco deve ser voltado ao estilo de aprendizagem baseado no Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA), onde o seu diferencial é a aplicação desse perfil na construção do plano, desenvolvendo dinâmicas e tarefas de execução específicas que engajem os traços do perfil solicitado.

Objetivo: Desenvolver um plano de estudo para Teste de Software.

Escopo: O plano deve abordar o conteúdo inicial de fundamentos de testes até gerenciamento de testes.

Tempo: O plano deve conter entre 2 a 4 semanas.

Perfil de Aprendizagem: Aplicar o estilo de aprendizagem baseado na teoria de Honey-Alonso: Reflexivo.

Fontes Externas: Com base no perfil de aprendizagem, o plano deve conter uma quantidade razoável de fontes onde possa basear o estudo como Livros, Vídeos, Músicas e Conteúdos diversos.

Output

Aqui está um plano de estudo personalizado de 4 semanas para **Teste de Software**, estruturado especificamente para o perfil **reflexivo** do questionário CHAEA.

Como discente reflexivo, você aprende melhor quando pode observar as situações por diferentes ângulos, coletar dados, ponderar alternativas e refletir antes de agir. Por isso, este plano prioriza a análise de casos, a observação detalhada, o registro em diários de bordo e a autoavaliação, evitando decisões precipitadas ou práticas sem fundamentação teórica.



Visão Geral do Perfil Reflexivo no Plano

- **Postura:** Observador, coletor de dados, analítico e prudente.
- **Estratégia central:** Investigar o "porquê" e o "como" das técnicas de teste antes de sair executando ferramentas.
- **Recurso-chave:** Criação de um **Diário de Aprendizagem Reflexivo (DAR)**, onde você registrará suas impressões, hipóteses e conclusões ao final de cada atividade.



Cronograma de Estudos (4 Semanas)

Semana 1: Fundamentos e Princípios do Teste de Software

Objetivo: Compreender a mentalidade de teste, a psicologia do teste e os 7 princípios fundamentais da área.

- **Atividades de Execução (Foco Reflexivo):**
 - **Análise de Falhas Históricas:** Pesquise sobre uma falha de software famosa (ex: o bug do Milênio, o caso do Therac-25 ou a falha do Ariane 5). Escreva uma breve reflexão de uma página no seu DAR respondendo: *Qual princípio de teste foi violado? Como o teste precoce poderia ter evitado isso?*
 - **Mapeamento de Processo:** Desenhe o ciclo de vida de desenvolvimento de software (SDLC) tradicional e o ágil. Identifique e circule onde o analista de teste deve atuar em cada um, justificando o impacto de sua presença em cada etapa.
- **Fontes de Estudo Recomendadas:**
 - **Livro:** "Fundamentos da Teste de Software" (Rex Black, Erik van Veenendaal, Dorothy Graham) – Capítulos iniciais (Alinhado ao pensador reflexivo).
 - **Vídeo:** Playlist "Curso de Teste de Software" (Iterasys ou Júlio de Lima no YouTube) – Módulo de Introdução.

- *Música para Foco:* Playlist *Deep Focus* ou *Lo-Fi Beats* no Spotify (ajuda na concentração profunda que o perfil exige).

Semana 2: Técnicas de Caixa-Preta e Caixa-Branca

Objetivo: Entender a diferença teórica e a aplicação prática das técnicas de *design* de teste estruturais e funcionais.

- **Atividades de Execução (Foco Reflexivo):**

- **Análise Comparativa:** Crie uma tabela comparativa detalhando as vantagens, desvantagens e os cenários ideais para o uso de *Particionamento de Equivalência* vs. *Análise de Valor Limite*.
- **Estudo de Caso Estático:** Pegue os termos de uso ou a especificação de um sistema simples que você usa no dia a dia (ex: tela de login). Antes de testar, faça uma revisão estática da especificação procurando ambiguidade ou lacunas de lógica. Anote suas conclusões.

- **Fontes de Estudo Recomendadas:**

- *Livro:* "Base de Conhecimento em Teste de Software" (Gustavo Bernardo).
- *Conteúdo Diverso:* Guia do Sílabo CTFL (Certified Tester Foundation Level) do ISTQB – Seção sobre Técnicas de Teste. O rigor conceitual do sílabo atrai o perfil reflexivo.

Semana 3: Níveis, Tipos de Teste e Automação vs. Manual

Objetivo: Diferenciar testes de unidade, integração, sistema e aceitação, além de compreender o papel da automação.

- **Atividades de Execução (Foco Reflexivo):**

- **A Matriz de Decisão:** Diante do cenário "Um e-commerce precisa validar seu fluxo de pagamento em 48h", reflita por escrito: *Quais tipos de teste (Funcional, Desempenho, Segurança, Usabilidade) são prioritários? O que automatizar e o que testar manualmente sob a ótica do Retorno sobre o Investimento (ROI)?*
- **Observação de Código:** Assista a um tutorial de automação de testes (ex: Selenium ou Cypress). Em vez de codificar imediatamente, anote em tópicos os padrões de organização de código utilizados (como o *Page Objects Pattern*) e pondere sobre a sua manutenibilidade.

- **Fontes de Estudo Recomendadas:**

- *Vídeo:* Palestras do canal *Testing Conference* ou *MTS (Mesa Redonda de Teste de Software)* no YouTube, focando em debates conceituais.
- *Livro:* "Automação de Testes de Software" (Elias Nogueira).

Semana 4: Gerenciamento e Processo de Teste

Objetivo: Compreender o planejamento de testes, estimativas, monitoramento, controle e o reporte de defeitos (Bug Report).

- **Atividades de Execução (Foco Reflexivo):**

- **Simulação de Plano de Teste:** Com base em um sistema fictício (ex: Aplicativo de Entrega de Comida), elabore a estrutura de um mini Plano de Teste (Escopo, Critérios de Entrada e Saída, Riscos).
- **Análise de Anatomia de um Bug:** Analise criticamente o que torna um relatório de bug "bom" ou "ruim". Escreva um template ideal de reporte de defeitos focado na clareza e na redução do tempo de triagem do desenvolvedor.

- **Fontes de Estudo Recomendadas:**

- *Livro:* "Gerenciamento de Teste de Software" (Rex Black).
- *Conteúdo Diverso:* Artigos da plataforma Medium (tags: *Software Testing*, *QA*, *Test Management*) para observar como o mercado gerencia crises e métricas.



Revisões, Autoavaliação e Autonomia

Como o estudante reflexivo se beneficia muito da autoanálise, o processo avaliativo deste plano é focado no monitoramento interno do progresso.

1. Ritos de Revisão (Semanal)

Ao final de cada sexta-feira, reserve 45 minutos para revisar suas notas no **Diário de Aprendizagem Reflexivo (DAR)**. Releia suas reflexões das semanas anteriores para consolidar as conexões neurais e verificar se sua percepção sobre os conceitos mudou.

2. Formas Avaliativas

- **Simulados Conceptuais:** Realize simulados oficiais ou gratuitos da certificação CTFL-ISTQB. Para o perfil reflexivo, o valor não está apenas em acertar a resposta, mas em **analisar detalhadamente as alternativas incorretas** e entender exatamente *por que* elas estão erradas.
- **Autoavaliação por Parágrafo:** Ao final do plano, responda às seguintes perguntas-chave de autoavaliação:
 1. *Consigo explicar a diferença entre um Erro, um Defeito e uma Falha com exemplos claros?*
 2. *Diante de um software desconhecido, quais seriam meus primeiros passos analíticos antes de planejar os testes?*
 3. *Quais lacunas teóricas ainda sinto que preciso preencher?*

3. Estímulo à Autonomia

Este plano não diz qual ferramenta exata você deve clicar, mas sim os conceitos que guiam o uso de qualquer ferramenta. Sinta-se livre para ajustar o tempo de estudo de cada módulo se perceber que um conceito (como as técnicas de Caixa-Branca) exige mais tempo de maturação intelectual do que outro. Conconfie no seu ritmo analítico.

Avaliação

Indicadores	Descrição	Nota
1. As atividades são adaptadas ao estilo de aprendizagem?	Foco em avaliar se o plano cita e aplica conceitualmente o estilo de aprendizagem em sua construção.	1
2. Existem instruções específicas relacionadas ao estilo de aprendizagem?	Foco em verificar se há orientações diretas sobre <i>como</i> o estudante deve executar as atividades de acordo com o seu perfil.	1
3. Não há inclusão de temas que fogem do escopo?	Foco em atestar se o roteiro se mantém estritamente dentro do conteúdo técnico solicitado no <i>prompt</i> , sem desvios.	1

4. O objetivo do plano foi distribuído adequadamente dentro do tempo estimado?	Foco em verificar se a carga de estudos foi bem fracionada no horizonte temporal proposto (2 a 4 semanas).	1
5. As metas são claras e bem estruturadas?	Foco em avaliar se as metas propostas são tangíveis, mensuráveis e de fácil compreensão e aplicação.	1
6. O plano estabelece métodos claros de avaliação de aprendizagem?	Foco em identificar a presença de instrumentos práticos para testar a retenção do que foi estudado.	1
7. O plano inclui formas de revisar o aprendizado?	Foco em verificar a existência de rotinas ou momentos programados para a fixação do conteúdo abordado.	1
8. As tarefas propostas apresentam dinâmicas de execução específicas para os traços do perfil solicitado?	Foco em avaliar se o plano consegue estimular o engajamento por meio de dinâmicas práticas voltadas exclusivamente ao perfil do estudante.	1
9. Há indicação de fontes ou materiais de apoio (links, livros, sites)?	Foco em constatar a inclusão de uma seção dedicada à curadoria de materiais externos complementares.	1
10. O plano incentiva a autonomia, autoavaliação e reflexão do estudante?	Foco em avaliar se o plano estimula o discente a gerenciar o próprio aprendizado e a aplicar métodos de autoanálise.	1

Nota Final: 10

Apêndice D - Teste de Software, Modelo GPT, Perfil Reflexivo

Documento de Avaliação

LLM: Chat GPT 5.5.

componente curricular: Teste de Software

Perfil de Aprendizagem: Reflexivo.

O documento conta com a seguinte estrutura:

- *Prompt*;
- *Output*;
- *Avaliação*.

Prompt

Visão Geral: Seu papel é desenvolver um plano de estudo que consiga atender aos seguintes requisitos com eficiência: o plano deve possuir objetivos claros, descrever bem as atividades a serem realizadas, incentivar a autonomia, autoavaliação e reflexão do discente, e propor formas avaliativas e revisões. O foco deve ser voltado ao estilo de aprendizagem baseado no Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA), onde o seu diferencial é a aplicação desse perfil na construção do plano, desenvolvendo dinâmicas e tarefas de execução específicas que engajem os traços do perfil solicitado.

Objetivo: Desenvolver um plano de estudo para Teste de Software.

Escopo: O plano deve abordar o conteúdo inicial de fundamentos de testes até gerenciamento de testes.

Tempo: O plano deve conter entre 2 a 4 semanas.

Perfil de Aprendizagem: Aplicar o estilo de aprendizagem baseado na teoria de Honey-Alonso: Reflexivo.

Fontes Externas: Com base no perfil de aprendizagem, o plano deve conter uma quantidade razoável de fontes onde possa basear o estudo como Livros, Vídeos, Músicas e Conteúdos diversos.

Output

Plano de Estudo — Teste de Software

Perfil de Aprendizagem: Reflexivo (Honey-Alonso)

O perfil reflexivo caracteriza estudantes que aprendem melhor observando, analisando experiências, registrando percepções, comparando abordagens e refletindo antes de agir. Assim, este plano prioriza:

- análise crítica de exemplos;
- registros de aprendizado;
- observação de casos reais;
- comparações entre técnicas;
- resolução gradual de problemas;
- revisões frequentes;
- momentos de autoavaliação e reflexão.

O plano foi estruturado para **4 semanas**, permitindo assimilação progressiva do conteúdo de Teste de Software, desde fundamentos até gerenciamento de testes.

Objetivo Geral

Compreender os principais conceitos, técnicas e práticas de Teste de Software, desenvolvendo capacidade analítica para identificar falhas, estruturar casos de teste e compreender o gerenciamento do processo de testes.

Objetivos Específicos

Ao final do plano, o estudante deverá ser capaz de:

- compreender os fundamentos de qualidade e testes;
- diferenciar tipos e níveis de teste;
- elaborar casos e cenários de teste;
- aplicar técnicas de teste funcional e estrutural;
- compreender automação básica de testes;
- interpretar métricas e documentação de testes;
- entender conceitos de gerenciamento e planejamento de testes;
- refletir criticamente sobre falhas e melhoria contínua.

Estrutura Geral do Plano

- **Duração:** 4 semanas
 - **Carga sugerida:** 1h30 a 2h por dia
 - **Método principal:** leitura analítica + reflexão + observação prática + revisão escrita
 - **Ferramentas recomendadas:**
 - Notion ou Obsidian (diário reflexivo)
 - Trello (organização)
 - Draw.io (mapas mentais)
 - VS Code
 - JUnit ou PHPUnit (contato inicial com testes)
-

Semana 1 — Fundamentos de Teste de Software

Objetivos da Semana

- Entender o conceito de qualidade de software;
 - Compreender o propósito dos testes;
 - Conhecer princípios fundamentais de teste.
-

Conteúdos

- O que é teste de software
 - Qualidade de software
 - Erros, defeitos e falhas
 - Princípios fundamentais de testes
 - Ciclo de vida do teste
 - Níveis de teste:
 - unitário
 - integração
 - sistema
 - aceitação
-

Atividades

1. Leitura Reflexiva

Ler capítulos introdutórios sobre fundamentos de teste e criar um diário contendo:

- principais conceitos;
- dúvidas;
- exemplos reais encontrados;
- comparação entre teoria e experiências pessoais.

2. Mapa Mental Comparativo

Criar um mapa mental comparando:

- erro × defeito × falha;
- tipos de teste;
- qualidade × teste.

3. Observação Crítica

Assistir vídeos sobre falhas famosas em software e responder:

- o que causou a falha?
- qual tipo de teste poderia evitar o problema?
- houve falha de processo ou técnica?

4. Exercício de Reflexão

Escrever uma página sobre:

“Por que testar software é importante além de encontrar bugs?”

Revisão da Semana

Técnica sugerida:

- Revisão escrita + releitura do diário reflexivo.

Autoavaliação

Responder:

- consigo explicar os princípios de teste sem consultar material?
 - consigo diferenciar os níveis de teste?
 - consigo identificar exemplos reais de falhas?
-

Fontes Recomendadas

Livros

- Introdução ao Teste de Software
- Foundations of Software Testing

Vídeos

- Curso de Teste de Software — Curso em Vídeo
- Introdução ao QA — canais de tecnologia no YouTube

Conteúdos Complementares

- Estudos de caso sobre falhas da NASA, Ariane 5 e sistemas bancários.

Música para concentração/reflexão

- Lo-fi instrumental
 - Música clássica leve
 - Ambient focus playlists
-

Semana 2 — Técnicas e Casos de Teste

Objetivos da Semana

- Aprender técnicas de criação de testes;
 - Desenvolver raciocínio analítico;
 - Elaborar casos e cenários de teste.
-

Conteúdos

- Caso de teste
 - Cenário de teste
 - Massa de dados
 - Técnicas caixa-preta:
 - partição de equivalência
 - valor limite
 - tabela de decisão
 - Técnicas caixa-branca
 - Cobertura de testes
-

Atividades

1. Análise Comparativa

Escolher um sistema simples (login, cadastro ou calculadora) e:

- listar funcionalidades;
- criar cenários de teste;
- refletir sobre possíveis falhas.

2. Diário Reflexivo Técnico

Após cada técnica estudada, registrar:

- quando usar;
- vantagens;
- limitações;
- exemplos práticos.

3. Exercício de Observação

Analisar aplicações reais e identificar:

- quais testes parecem ter sido feitos;
- quais problemas ainda existem.

4. Produção de Casos de Teste

Criar casos de teste completos contendo:

- ID
 - objetivo
 - pré-condição
 - passos
 - resultado esperado
-

Revisão da Semana

Método

- Reescrever casos de teste após revisão crítica;
- Comparar versões antigas e novas.

Autoavaliação

Responder:

- consigo criar cenários completos?
 - sei aplicar valor limite?
 - consigo justificar a técnica escolhida?
-

Fontes Recomendadas

Livros

- Software Testing Techniques

Vídeos

- Testes Caixa-Preta e Caixa-Branca
- Exemplos de casos de teste práticos

Plataformas

- [Guru99 QA Tutorials](#)
 - [Ministry of Testing](#)
-

Semana 3 — Testes Automatizados e Documentação

Objetivos da Semana

- Compreender automação de testes;
 - Conhecer ferramentas básicas;
 - Entender documentação de testes.
-

Conteúdos

- Testes automatizados
 - Vantagens e limitações
 - Testes unitários
 - Introdução ao JUnit/PHPUnit
 - Relatórios de teste
 - Evidências de teste
 - Bugs e rastreamento
-

Atividades

1. Observação Guiada

Assistir demonstrações de automação e registrar:

- quais etapas são automatizadas;
- vantagens percebidas;
- dificuldades observadas.

2. Laboratório Reflexivo

Criar testes simples automatizados.

Após executar:

- analisar erros encontrados;
- registrar comportamento do sistema;
- descrever aprendizados.

3. Comparação Crítica

Escrever um comparativo:

Teste Manual	Teste Automatizado
vantagens	vantagens
limitações	limitações
quando usar	quando usar

4. Análise de Bugs

Pesquisar bugs reais em sistemas conhecidos e refletir:

- impacto do erro;
- custo da falha;
- como poderia ter sido evitado.

Revisão da Semana

Método

- Revisão prática:
 - reexecutar testes;
 - analisar logs;
 - revisar documentação criada.

Autoavaliação

Responder:

- compreendo o objetivo da automação?
 - consigo interpretar um relatório de teste?
 - consigo explicar quando automatizar?
-

Fontes Recomendadas

Ferramentas

- [JUnit](#)
- [PHPUnit](#)
- [Selenium](#)

Vídeos

- Introdução ao Selenium
- Primeiros testes automatizados

Livros

- JUnit in Action
-

Semana 4 — Gerenciamento de Testes

Objetivos da Semana

- Entender planejamento de testes;
 - Compreender métricas e gerenciamento;
 - Desenvolver visão estratégica de QA.
-

Conteúdos

- Planejamento de testes
 - Estratégia de testes
 - Métricas
 - Priorização
 - Gestão de defeitos
 - Testes ágeis
 - Papéis da equipe QA
-

Atividades

1. Plano de Teste Reflexivo

Criar um plano de testes simples contendo:

- escopo;
- riscos;
- estratégia;
- cronograma;
- critérios de aceitação.

2. Estudo de Caso

Analisar um projeto fictício e responder:

- quais testes priorizar?
- quais riscos existem?
- como organizar a equipe?

3. Reflexão Final

Produzir um relatório respondendo:

- quais conhecimentos foram mais difíceis?
- quais estratégias funcionaram melhor?
- como o perfil reflexivo ajudou no aprendizado?

4. Autoanálise Profissional

Refletir sobre:

- interesse na área de QA;
 - habilidades desenvolvidas;
 - pontos a melhorar.
-

Revisão Geral Final

Estratégia

- Revisão completa do diário;
 - Revisão dos mapas mentais;
 - Reescrita dos principais conceitos;
 - Simulado teórico.
-

Métodos Avaliativos

Avaliação Formativa

Ocorrerá continuamente através de:

- diário reflexivo;
 - mapas mentais;
 - produção de casos de teste;
 - análises críticas;
 - revisões semanais.
-

Avaliação Somativa

Projeto Final

Criar documentação de testes para um sistema simples contendo:

- cenários;
 - casos de teste;
 - relatório de defeitos;
 - plano de teste resumido.
-

Estratégias Voltadas ao Perfil Reflexivo (CHAEA)

Este plano utiliza abordagens específicas para o perfil reflexivo:

Característica Reflexiva	Aplicação no Plano
-------------------------------------	---------------------------

Observação antes da ação	análise de casos reais
Organização de ideias	mapas mentais e diários
Aprendizagem gradual	divisão semanal progressiva
Reflexão crítica	perguntas analíticas
Comparação de abordagens	testes manuais × automatizados
Registro detalhado	diário de aprendizagem
Tempo para análise	revisões frequentes

Resultado Esperado

Ao final das 4 semanas, o estudante terá desenvolvido:

- base sólida em Teste de Software;
- capacidade crítica para análise de falhas;
- autonomia de estudo;
- habilidade inicial em criação e gerenciamento de testes;
- compreensão prática e teórica da área de QA.

Avaliação

Indicadores	Descrição	Nota
1. As atividades são adaptadas ao estilo de aprendizagem?	Foco em avaliar se o plano cita e aplica conceitualmente o estilo de aprendizagem em sua construção.	1
2. Existem instruções específicas relacionadas ao estilo de aprendizagem?	Foco em verificar se há orientações diretas sobre <i>como</i> o	1

	estudante deve executar as atividades de acordo com o seu perfil.	
3. Não há inclusão de temas que fogem do escopo?	Foco em atestar se o roteiro se mantém estritamente dentro do conteúdo técnico solicitado no <i>prompt</i> , sem desvios.	1
4. O objetivo do plano foi distribuído adequadamente dentro do tempo estimado?	Foco em verificar se a carga de estudos foi bem fracionada no horizonte temporal proposto (2 a 4 semanas).	1
5. As metas são claras e bem estruturadas?	Foco em avaliar se as metas propostas são tangíveis, mensuráveis e de fácil compreensão e aplicação.	1
6. O plano estabelece métodos claros de avaliação de aprendizagem?	Foco em identificar a presença de instrumentos práticos para testar a retenção do que foi estudado.	1
7. O plano inclui formas de revisar o aprendizado?	Foco em verificar a existência de rotinas ou momentos programados para a fixação do conteúdo abordado.	1
8. As tarefas propostas apresentam dinâmicas de execução específicas para os traços do perfil solicitado?	Foco em avaliar se o plano consegue estimular o engajamento por meio de dinâmicas práticas voltadas exclusivamente ao perfil do estudante.	1
9. Há indicação de fontes ou materiais de apoio (links, livros, sites)?	Foco em constatar a inclusão de uma seção dedicada à curadoria de materiais externos complementares.	1
10. O plano incentiva a autonomia, autoavaliação e reflexão do estudante?	Foco em avaliar se o plano estimula o discente a gerenciar o próprio aprendizado e a aplicar métodos de autoanálise.	1

Nota Final: 10

Anexo A - Questionário Honey-Alonso de Estilos de Aprendizagem (CHAEA)

Instruções de preenchimento: O questionário a seguir foi construído para identificar o seu estilo preferido de aprendizagem. Não há respostas certas ou erradas, mas apenas a sua opinião. Leia cada uma das 80 afirmações e assinale apenas aquelas com as quais você se identifica ou está mais de acordo do que em desacordo.

Questionário Honey-Alonso de Estilos de Aprendizagem: CHAEA

1 Tenho fama de dizer o que penso claramente e sem rodeios.

2 A maior parte das vezes, sinto-me seguro(a) do que está correcto e do que está incorrecto.

3 Muitas vezes, actuo sem olhar às consequências.

4 Normalmente, procuro resolver os problemas metodicamente e passo a passo.

5 Creio que o formalismo restringe e limita a actuação livre das pessoas.

6 Interessa-me saber quais são os sistemas de valores dos outros e com que critérios actuam.

7 Penso que o agir intuitivamente pode ser sempre tão válido como agir reflexivamente.

8 Creio que, independentemente, dos métodos o mais importante é que as coisas funcionem.

9 Estou atento a todos os pormenores das disciplinas que frequento (sumários, textos, etc).

10 Agrada-me ter tempo para preparar o meu trabalho e realizá-lo com consciência.

11 Sou adepto(a) da autodisciplina, seguindo uma certa ordem, por exemplo, no regime alimentar, no estudo e no exercício físico, etc.

12 Quando ouço uma ideia nova, começo logo a pensar como poderei pô-la em prática.

13 Prefiro as ideias originais e inovadoras, ainda que não sejam práticas.

14 Só admito e me adapto às normas, se servem para atingir os meus objectivos.

15 Adapto-me melhor às pessoas reflexivas do que às pessoas demasiado espontâneas e imprevisíveis.

16 Escuto com mais frequência do que falo.

17 Prefiro as coisas estruturadas às desordenadas.

18 Preocupo-me em interpretar, cuidadosamente, a informação disponível antes de tirar uma conclusão.

19 Antes de fazer alguma coisa, analiso com cuidado as suas vantagens e inconvenientes.

20 Entusiasma-me ter de fazer de algo novo e diferente.

21 Procuro, quase sempre, ser coerente com os meus princípios, seguindo critérios e sistemas de valores.

22 Quando há uma discussão, não gosto de estar com rodeios.

23 Tenho tendência a relacionar-me de um modo distante, e algo formal com as pessoas com quem trabalho.

24 Gosto mais das pessoas realistas e concretas do que das idealistas.

25 Tenho dificuldade em ser criativo(a) e em romper com as estruturas existentes.

26 Sinto-me bem com pessoas espontâneas.

27 A maior parte das vezes, expresso, abertamente, os meus sentimentos.

28 Gosto de analisar as coisas de todos os ângulos.

29 Incomoda-me que as pessoas não tomem as coisas a sério.

30 Atrai-me experimentar e praticar as últimas técnicas e novidades.

31 Sou cauteloso(a) na hora de tirar conclusões.

32 Prefiro contar com o maior número de fontes de informação, ou seja, quantos mais dados tiver, melhor.

33 Tendo a ser perfeccionista.

34 Prefiro ouvir as opiniões dos outros antes de expor as minhas.

35 Gosto de enfrentar a vida de forma espontânea e não ter que planificar tudo previamente.

36 Nas discussões, gosto de observar como agem os outros participantes.

37 Sinto-me, pouco à vontade, com pessoas demasiado analíticas.

38 Avalio, com frequência, as ideias dos outros pelo seu valor prático.

39 Sinto-me oprimido(a), se me obrigam a acelerar o trabalho para cumprir um prazo.

40 Nas reuniões, apoio as ideias práticas e realistas.

41 É melhor gozar o momento presente do que sentir prazer pensando no passado ou no futuro.

42 Incomodam-me as pessoas que desejam sempre apressar as coisas.

43 Emito ideias novas e espontâneas nos grupos de discussão.

44 Penso que são mais consistentes as decisões fundamentadas numa análise minuciosa que as baseadas na intuição.

45 Detecto, frequentemente, a inconsistência e os pontos débeis nas argumentações dos outros.

46 Creio que me é mais frequente ter de desobedecer às regras do que seguiras.

47 Apercebo-me, frequentemente, de outras formas melhores e mais práticas de fazer as coisas.

48 Em geral, falo mais que escuto.

49 Prefiro distanciar-me dos factos e observá-los de outras perspectivas.

50 Estou convencido(a) que numa situação se deve impor a lógica e o raciocínio.

51 No meu dia a dia procuro novas experiências.

52 Quando ouço falar de uma ideia ou de uma nova abordagem, tento imediatamente encontrar aplicações concretas.

53 Penso que devemos chegar, o mais rapidamente possível, à ideia central dos assuntos.

54 Esforço-me sempre por conseguir conclusões e ideias claras.

55 Prefiro discutir questões concretas e não perder tempo com ideias

abstractas.

56 Impaciento-me, quando me dão explicações irrelevantes ou incoerentes.

57 Verifico, sempre, com antecedência, se as coisas funcionam como deve ser.

58 Faço vários rascunhos antes da redacção definitiva de um trabalho.

59 Estou consciente de que, nas discussões, ajudo a manter os outros centrados no tema, evitando divagações.

60 Observo que sou, com frequência, uma das pessoas mais objectivas e desapaixonadas nas discussões.

61 Quando algo corre mal, tento logo fazer melhor.

62 Rejeito ideias originais se me parecem impraticáveis.

63 Pondero sempre diversas alternativas, antes de tomar uma decisão.

64 É frequente eu tentar prever o futuro.

65 Nos debates e discussões prefiro desempenhar um papel secundário em vez de ser o(a) líder ou o(a) que mais participa.

66 Incomodam-me as pessoas que não agem com lógica.

67 Incomoda-me ter de planificar e prever as coisas.

68 Penso que, muitas vezes, os fins justificam os meios.

69 Costumo pensar, profundamente, sobre os assuntos e os problemas. 1 2 3

70 O trabalhar consciente enche-me de satisfação e orgulho.

71 Perante os acontecimentos, tento descobrir os princípios e as teorias que os fundamentam.

72 Desde que possa atingir os meus fins, sou capaz de ferir os sentimentos de outros.

73 Não me importo de fazer tudo o que seja necessário para que o meu trabalho seja eficiente.

74 Sou com frequência umas das pessoas que mais animam as festas.

75 Aborreço-me, rapidamente, com o trabalho metódico e minucioso.

76 As pessoas costumam pensar que sou insensível aos seus sentimentos.

77 Costumo deixar-me levar pela minha intuição.

78 Se faço parte de um grupo de trabalho, procuro que se siga um plano e uma metodologia.

79 Interessa-me, com frequência, descobrir o que pensam as pessoas.

80 Evito os assuntos subjectivos, ambíguos e pouco claros.

Metodologia de Correção e Pontuação:

A pontuação do questionário é obtida de forma quantitativa. O respondente soma 1 ponto para cada resposta afirmativa assinalada. O total de pontos é dividido nos quatro grupos de estilos de aprendizagem, obedecendo ao seguinte mapeamento de itens no questionário:

Estilo Ativo: Somar 1 ponto para cada resposta afirmativa nas questões 3, 5, 7, 9, 13, 20, 26, 27, 35, 37, 41, 43, 46, 48, 51, 61, 67, 74, 75, 77.

Estilo Reflexivo: Somar 1 ponto para cada resposta afirmativa nas questões 10, 16, 18, 19, 28, 31, 32, 34, 36, 39, 42, 44, 49, 55, 58, 63, 65, 69, 70, 79.

Estilo Teórico: Somar 1 ponto para cada resposta afirmativa nas questões 2, 4, 6, 11, 15, 17, 21, 23, 25, 29, 33, 45, 50, 54, 60, 64, 66, 71, 78, 80.

Estilo Pragmático: Somar 1 ponto para cada resposta afirmativa nas questões 1, 8, 12, 14, 22, 24, 30, 38, 40, 47, 52, 53, 56, 57, 59, 62, 68, 72, 73, 76.

O estilo que obtiver o maior somatório final é classificado como o perfil cognitivo predominante do estudante.

Fonte: Alonso, Gallego e Honey (2007), com tradução e validação para a língua portuguesa adaptada por Miranda e Moraes (2008).