

# Desenvolvimento de uma Aplicação Web para Criação de Tabelas de Decisão com Inteligência Artificial Integrada

Trabalho de Conclusão do Curso de Tecnologia em Sistemas Para Internet

Ueber Syemmer Austria dos Santos  
Orientadora: Fabrícia Py Tortelli Noronha

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS) Campus  
Porto Alegre

Av Cel Vicente, 281, Porto Alegre – RS – Brasil

uebersyemmer@gmail.com, fabricia.noronha@poa.ifrs.edu.br

**Resumo.** *A tabela de decisão é uma técnica essencial para especificar regras de negócios de forma estruturada e clara, facilitando a implementação e verificação da lógica de negócios e testes de software. No entanto, a criação manual de tabelas pode ser um processo trabalhoso e propenso a erros, especialmente quando envolve múltiplas condições e ações de uma determinada funcionalidade. Este trabalho culminou no desenvolvimento de uma aplicação web integrada com inteligência artificial, chamada Tabela.AI, que auxilia na criação de tabelas de decisão aplicáveis aos testes de software. A ferramenta visa representar de maneira precisa e organizada as múltiplas condições e ações, otimizando o processo de criação de cenários de teste e assegurando maior precisão na validação de sistemas. A metodologia se baseia em pesquisa exploratória, incluindo levantamento bibliográfico e análise de ferramentas semelhantes. Os testes foram realizados com profissionais de desenvolvimento e qualidade de software e os dados foram produzidos por meio de um questionário. Os resultados indicam que a Tabela.AI atingiu seu objetivo de automatizar a criação de tabelas de decisão e cenários de teste, proporcionando melhorias significativas, como eficiência no processo, economia de tempo e maior confiabilidade e precisão, garantindo uma cobertura mais abrangente nos testes. A Tabela.AI mostra-se uma ferramenta essencial para profissionais da área de desenvolvimento e qualidade de software, evidenciando o potencial da inteligência artificial em otimizar processos de forma eficiente.*

**Palavras-chave:** *Tabela de decisão. Teste de software. Inteligência Artificial.*

## 1. Introdução

O teste é uma etapa essencial no desenvolvimento de sistemas e essa é uma área que está em constante evolução. Entretanto, para que os testes contemplem todas as funcionalidades que foram desenvolvidas, faz-se necessária a geração de cenários de teste abrangentes e bem definidos. Nesse contexto, as tabelas de decisão surgem como uma ferramenta para estruturar e visualizar cenários de teste fazendo com que os softwares desenvolvidos alcancem um maior nível de confiabilidade.

A tabela de decisão fornece uma comunicação efetiva sobre suas condições e ações entre os profissionais da área de desenvolvimento. Além de uma fácil leitura, as tabelas de decisão, ao serem examinadas, podem minimizar as probabilidades de ocorrência de erros. No entanto, a criação manual dessas tabelas pode se tornar uma tarefa desafiadora à medida que haja cenários que contenham múltiplas condições e ações envolvidas.

Este estudo apresenta o desenvolvimento de uma aplicação web integrada com inteligência artificial (IA), projetada para facilitar a criação de tabelas de decisão. A ferramenta Tabela.AI busca solucionar os desafios associados a esse processo, otimizando a geração de tabelas de decisão e, assim, economizando tempo, reduzindo custos e fornecendo resultados padronizados e objetivos.

Para atingir esse objetivo, adotou-se a pesquisa exploratória, visando entender como os profissionais lidam com a geração de cenários de teste, identificando seus principais obstáculos. Juntamente, foi conduzida uma pesquisa bibliográfica para sistematizar o conhecimento existente relacionado à melhoria na escrita de cenários de teste.

O artigo está organizado da seguinte forma: a fundamentação teórica é apresentada na seção 2; os trabalhos relacionados são discutidos na seção 3; a metodologia é descrita na seção 4; a modelagem da aplicação é abordada na seção 5; o desenvolvimento é tratado na seção 6; os testes com usuários são apresentados na seção 7; e, por fim, as considerações finais estão na seção 8.

## **2. Fundamentação teórica**

Nos últimos anos, o desenvolvimento de sistemas cada vez mais complexos impulsionou o aumento de investimentos em testes de software, por parte das empresas de tecnologia da informação, a fim de garantir o funcionamento correto das aplicações (Hoppe, 2015). Nesse sentido, tornou-se fundamental incluir as atividades relacionadas ao teste de software no processo de desenvolvimento, além de capacitar profissionais para validar os requisitos e verificar o funcionamento dos sistemas, assegurando a qualidade do software (Hoppe, 2015).

Um cenário de teste é uma representação de um conjunto específico de condições sob as quais um sistema é testado. Esses cenários são essenciais para os profissionais da área de desenvolvimento de software porque permitem a validação sistemática de cada funcionalidade do sistema em diferentes situações simuladas. Com base nas tabelas de decisão, os cenários de teste são elaborados para cobrir todas as possíveis interações entre entradas e saídas, garantindo que o software funcione conforme esperado nas mais diversas circunstâncias operacionais e de uso.

A inteligência artificial (IA) assume um papel importante na geração de tabelas de decisão, automatizando e otimizando o processo. De acordo com Lima (2023), através de algoritmos avançados de aprendizado de máquina, a IA é capaz de analisar grandes volumes de dados e cenários complexos, identificando padrões e relações entre variáveis. Essa automação não apenas acelera a criação das tabelas de decisão, mas também aprimora sua precisão e eficiência, permitindo que os desenvolvedores se concentrem em aspectos mais estratégicos e de alto valor durante o ciclo de desenvolvimento (Lima, 2023).

As tabelas de decisão são fundamentais para a elaboração de cenários de teste uma vez que fornecem uma base estruturada e objetiva para a definição de cada caso de teste. Ao mapear explicitamente todas as possíveis combinações de entrada e saída, as tabelas de decisão garantem que nenhum aspecto crítico do software seja negligenciado nos testes. Isso é essencial para minimizar riscos e assegurar que todas as funcionalidades do sistema se comportem conforme o esperado em diferentes contextos operacionais.

As tabelas de decisão são amplamente utilizadas em testes de software para avaliar requisitos de sistema e como diversas combinações de condições resultam em diferentes ações. Segundo Teixeira (p. 18, 2024), “[...] estas tabelas são constituídas por colunas, que correspondem às regras de decisão que definem combinações únicas de condições, e por linhas, que representam as ações resultantes do sistema”. Essa técnica, apesar de eficaz na seleção de casos de teste, pode apresentar desafios relacionados à sua complexidade e custo de manutenção, especialmente em cenários com múltiplas combinações de entradas e ações.

A tabela de decisão ajuda a verificar todas as combinações possíveis de condições para teste, além disso, os testadores também podem identificar facilmente as condições perdidas, sendo estas condições indicadas como valores True (T) e False (F)." (MADRUGA, 2021, p. 26). Condições perdidas referem-se a combinações que não foram consideradas durante o processo de planejamento, ou seja, passaram despercebidas por alguma razão, como falhas na análise inicial dos requisitos, o que pode resultar em falhas não identificadas no sistema.

A otimização da geração de tabelas de decisão é essencial para facilitar a criação de cenários de teste de forma eficiente e escalável. Ao desenvolver ferramentas ou aplicativos que automatizam este processo, os profissionais da área de desenvolvimento de software podem reduzir significativamente o tempo necessário para criar e revisar tabelas de decisão. Isso não apenas aumenta a produtividade da equipe de teste, mas também melhora a qualidade geral do software ao garantir uma cobertura abrangente de testes desde as fases iniciais do desenvolvimento.

Em resumo, o desenvolvimento de uma aplicação web integrada à inteligência artificial visa simplificar e acelerar a criação de tabelas de decisão. Desta forma, se destina a auxiliar os profissionais da área de desenvolvimento de software na geração de cenários de teste de modo mais eficiente e preciso, contribuindo para a redução de erros e economia de tempo, além de promover uma maior confiabilidade dos produtos desenvolvidos.

### **3. Trabalhos Relacionados**

Para a elaboração desta seção, foram realizadas buscas por ferramentas com propósitos semelhantes à aplicação web desenvolvida pelo presente trabalho. A partir das pesquisas, foram encontradas duas ferramentas para geração de tabela de decisão e geração de cenários de testes, entre elas: x-decision e teste.ai.

O **x-decision** é uma ferramenta de testes que permite a criação de tabelas de decisão, simplificando o processo de validação de requisitos e garantindo um maior controle sobre a qualidade do produto final.

Figura 1 - Ferramenta x-decision



Fonte: Capturado pelo autor (2024).<sup>1</sup>

A criação de uma tabela de decisão com o x-decision é realizada por meio de cinco passos. Iniciando pela inserção do requisito específico que é desejado testar, como ilustrado na Figura 1, em seguida, na tela seguinte, são inseridos todos os atributos de entrada relacionados ao requisito, depois os atributos de saída do requisito, que são os resultados esperados com base nos atributos de entrada fornecidos. O próximo passo é definir as classes de equivalência para cada atributo de entrada e saída. Por fim, a tabela de decisão é gerada, oferecendo uma visão organizada de como diferentes combinações de atributos de entrada resultam em diferentes atributos de saída, facilitando a identificação de cenários de teste e a verificação da conformidade com os requisitos.

Conforme pode ser observado na Figura 2, a plataforma **Teste.ai** conta com um conjunto de recursos, tais como: geradores de casos de teste, cenários e massa de dados, além de estratégias para testes de usabilidade, segurança e desempenho. A ferramenta possui integração com inteligência artificial e permite a geração automática de casos de teste com base na documentação do software ou em requisitos específicos. O modelo de linguagem é o OpenAI, combinados com prompts desenvolvidos para garantir respostas precisas.

---

<sup>1</sup> Site da captura: <http://www.juliodelima.com.br/xdecision/>

Figura 2 - Teste.ai



Fonte: Capturado pelo autor (2024).<sup>2</sup>

As ferramentas mencionadas foram analisadas considerando as funcionalidades iniciais vinculadas ao propósito deste trabalho. Sendo assim, as comparações se deram a partir de alguns critérios estabelecidos, tais como:

- **Criação de tabelas de decisão:** a criação de tabelas de decisão facilita a visualização clara das combinações de condições e ações, garantindo cobertura completa dos casos de teste. Isso minimiza erros e melhora a comunicação entre equipes de desenvolvimento.
- **Software livre/acessível:** ferramentas de custos reduzidos minimizam custos iniciais e tornam tecnologias avançadas acessíveis a pequenas empresas. Elas promovem flexibilidade e customização, beneficiando-se de atualizações contínuas.
- **Integração com IA:** a integração com IA automatiza a criação de cenários de teste e identifica padrões de falhas, aumentando a precisão e eficiência. Isso permite que testadores foquem em áreas críticas e estratégicas, melhorando a qualidade dos testes.
- **Interface amigável:** uma interface amigável garante que a ferramenta seja fácil de usar para todos, independentemente da experiência técnica. Isso melhora a produtividade, reduz a curva de aprendizado e minimiza erros operacionais, assegurando eficiência na criação de testes.

Embora a Tabela.AI não seja totalmente gratuita, seus custos são mínimos e acessíveis. Os principais custos estão associados ao uso da API da OpenAI, que calcula valores com base no número de tokens processados. Já a infraestrutura de hospedagem AWS oferece um período inicial gratuito de 750 horas para novas contas, permitindo a utilização sem custos adicionais no início do projeto.

<sup>2</sup> Site da captura: <https://www.teste.ai/>

A seguir, o Quadro 1 apresenta, de forma resumida, o comparativo entre as ferramentas analisadas e a Tabela.AI, ferramenta desenvolvida neste estudo.

Quadro 1 - Comparativo dos Trabalhos Relacionados

Critérios	Tabela.AI	x-decision	Teste.ai
Criação de tabelas de decisão	X	X	
Software livre/acessível	X	X	
Integração com IA	X		X
Interface amigável	X		X
Geração de cenários de teste	X		X

Fonte: Elaborado pelo autor (2024).

Além de reunir os critérios listados nas duas ferramentas e apresentar custos mínimos, a Tabela.AI se destaca pela integração da inteligência artificial, que otimiza a criação de cenários de teste de forma ágil e eficiente. Diferentemente dos métodos tradicionais, a IA analisa dados rapidamente, gerando cenários coerentes e abrangentes, minimizando possíveis erros humanos e garantindo maior cobertura de teste. Isso resulta em validações mais confiáveis e um processo de teste mais eficaz e preciso.

#### 4. Metodologia

De natureza exploratória, a pesquisa adotou uma metodologia estruturada em cinco etapas principais para o desenvolvimento da Tabela.AI, conforme descrito a seguir.

Inicialmente, realizou-se a pesquisa bibliográfica por meio de uma revisão de literatura em artigos científicos, materiais acadêmicos sobre a utilização de tabelas de decisão no contexto de testes de software e inteligência artificial, com ênfase em fontes disponíveis no Google Acadêmico.

A partir das strings de busca: “tabela de decisão” AND “tomada de decisão”, “inteligência artificial” AND “análise de requisitos em projetos de software”, “tabelas de decisão” AND “geração e execução de casos de teste” AND “tabela de decisão”, “tabela de decisão” AND “software” foram encontrados materiais que possibilitaram estudar temas, como: a importância da automação no desenvolvimento de cenários de teste, a aplicabilidade de IA na otimização desses processos e o impacto na qualidade do software. Essa etapa permitiu compreender o contexto em que o sistema seria desenvolvido e identificar as principais necessidades dos desenvolvedores e testadores de software.

Na segunda etapa, conduziu-se uma pesquisa sobre soluções semelhantes disponíveis no mercado, como sistemas que utilizam IA para automatizar a criação de cenários de teste ou ferramentas que geram tabelas de decisão de forma automática. Uma das ferramentas

mencionadas no artigo, a x-decision, Figura 1, já era do conhecimento do autor, enquanto a outra, Teste.ai, Figura 2, foi descoberta durante as pesquisas realizadas para este estudo.

Essa pesquisa possibilitou identificar as principais funcionalidades presentes nessas ferramentas e também lacunas que poderiam ser preenchidas com uma abordagem mais focada em automação e inteligência artificial. Além disso, ajudou a entender como outras soluções lidam com desafios semelhantes no campo de testes de software.

Vale ressaltar que não foram encontradas outras ferramentas relevantes que atendam aos critérios estabelecidos nesta pesquisa, indicando uma oportunidade para explorar soluções no campo da automação com inteligência artificial para testes de software.

Na terceira etapa, com base nos resultados das etapas anteriores, foi definida a lista de requisitos do sistema, incluindo funcionalidades essenciais, como a inserção de condições e ações, a comunicação com o modelo GPT-4-0-mini da OpenAI e a geração automática da tabela de decisão. Esses requisitos foram mapeados em um diagrama de casos de uso (Figura 3), oferecendo uma visão clara das operações que o sistema deve realizar, desde o recebimento de dados até a geração e visualização das tabelas de decisão.

Na quarta etapa, foi realizada a prototipagem de alta fidelidade da interface gráfica da Tabela.AI (Apêndice A). Essa prototipagem permitiu validar os requisitos e funcionalidades definidos na etapa anterior, servindo como base para nortear o desenvolvimento do front-end da aplicação Tabela.AI. O processo focou em proporcionar uma experiência de usuário simplificada para a inserção de condições e ações e na visualização clara das tabelas de decisão geradas automaticamente.

Na quinta etapa, realizou-se a implementação do sistema. Para o desenvolvimento do back-end, foi utilizado o Node.js e o framework Express, permitindo a comunicação entre o servidor e o modelo GPT-4-0-mini. O front-end foi desenvolvido com ReactJS, uma biblioteca JavaScript amplamente reconhecida, para permitir a inserção de dados e a visualização das tabelas geradas. A comunicação entre o front-end e o back-end ocorreu de maneira eficiente, garantindo que o sistema funcione de forma integrada e responsiva, proporcionando uma boa experiência aos usuários ao gerar tabelas de decisão. O código-fonte da aplicação está disponível no GitHub e pode ser acessado no seguinte endereço: <https://github.com/uebersantos/tcc-ueber>. Para completar, a aplicação foi implantada em uma instância EC2 na AWS (Apêndice B), permitindo que a Tabela.AI seja acessada em qualquer dispositivo com conexão à internet.

Por fim, na sexta e última etapa, ocorreu a validação da aplicação Tabela.AI (Apêndice C), conduzida com profissionais das áreas de desenvolvimento e qualidade de software. O objetivo desta etapa foi avaliar os potenciais benefícios da solução proposta, destacando a eficiência na criação de tabelas de decisão e na geração de cenários de teste, a qualidade do produto final e a aceitação da ferramenta pelos especialistas.

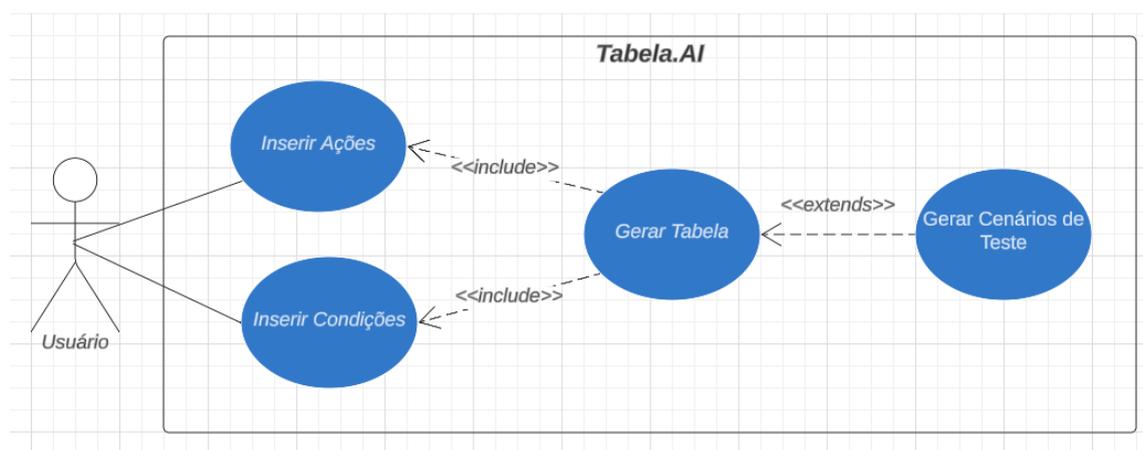
A partir da análise dos resultados, foram identificados aspectos relevantes sobre a usabilidade, precisão e organização das tabelas de decisão, além da utilidade da integração com inteligência artificial. Com base nesses resultados, foi realizado o aprimoramento do sistema, visando melhorar sua performance e eficiência.

Esses passos tiveram como objetivo assegurar que o desenvolvimento da aplicação fosse orientado por uma abordagem sólida, desde a pesquisa teórica até a implementação técnica, garantindo que as funcionalidades atendam às necessidades identificadas.

## 5. Modelagem

Para a modelagem da aplicação foi selecionado o diagrama de casos de uso que visa determinar quais funcionalidades integram o sistema. Assim, após realizar a análise dos trabalhos relacionados ao aqui desenvolvido, foi elaborado o diagrama de casos de uso, esquematizado a seguir.

Figura 3 - Diagrama de casos de uso



Fonte: Elaborado pelo autor (2024).

De acordo com a ilustração da Figura 3, o usuário é o responsável por inserir condições e ações para criar uma tabela de decisão que represente suas regras de negócios. Ele pode validar essas condições e ações na tabela de decisão para garantir o correto funcionamento. Além disso, é possível gerar uma tabela de decisão a partir das condições e ações inseridas.

Os cenários de teste, por sua vez, são derivados dessas tabelas de decisão, permitindo uma análise estruturada do comportamento do sistema. Eles representam combinações de condições e ações que verificam se o sistema funciona conforme o esperado, cobrindo tanto situações previstas, quanto possíveis falhas. Esses cenários desempenham um papel fundamental na garantia de qualidade, assegurando que o sistema atenda aos requisitos especificados e seja confiável para o usuário final.

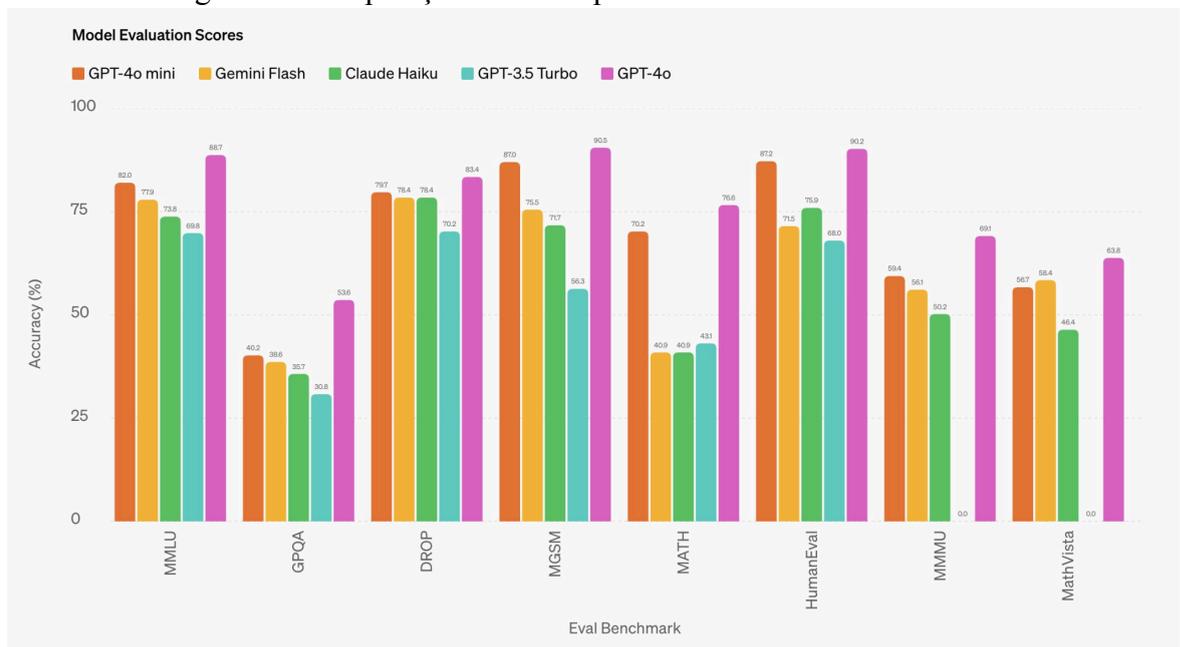
## 6. Desenvolvimento

O desenvolvimento da aplicação web está fundamentado na arquitetura Node.js para o back-end, utilizando o framework Express. Esse framework será responsável pelo gerenciamento dos endpoints da aplicação, facilitando a comunicação entre o servidor e o modelo GPT-4-0-mini da OpenAI. A principal funcionalidade da aplicação será permitir que os usuários insiram condições e ações relacionadas a regras de negócio. Com base nessas informações, a aplicação utilizará inteligência artificial para gerar automaticamente uma tabela de decisão, o que irá otimizar e facilitar a criação de cenários de teste de software de maneira eficiente.

A funcionalidade central da aplicação reside na integração com o modelo GPT-4o-mini, que processará os dados inseridos pelos usuários (condições e ações) e gerará uma tabela de decisão automatizada. Essa tabela será utilizada por desenvolvedores e testadores para construir cenários de teste de maneira mais rápida e precisa, eliminando a necessidade de criar esses cenários manualmente.

A seguir, a Figura 4 apresenta uma comparação do desempenho de cinco modelos de IA em diversos benchmarks: GPT-4o mini, Gemini Flash, Claude Haiku, GPT-3.5 Turbo e GPT-4o. Cada barra representa a precisão alcançada por cada modelo em tarefas específicas.

Figura 4 - Comparação de desempenho de modelos de IA



Fonte: Capturado pelo autor (2024).<sup>3</sup>

Ao analisar o gráfico da Figura 4, divulgado pelo Twitter oficial da OpenAI Developers (2024), e o artigo escrito por Roberto Dias Duarte (2024), o qual detalha a comparação entre modelos de IA de última geração, os principais fatores que motivaram a escolha do GPT-4o mini incluem sua alta precisão em benchmarks relevantes. No benchmark MMLU, que avalia a compreensão de linguagem em diversas tarefas e tópicos, o GPT-4o mini (representado pela barra laranja) demonstrou boa capacidade de compreensão, essencial para modelos voltados para testes de software. No HumanEval, que mede a eficácia na geração de código e resolução de problemas de programação, o desempenho do modelo destaca-se como uma habilidade importante para auxiliar na criação de cenários de teste.

No teste MGSM, que avalia a generalização matemática e a manipulação simbólica, o GPT-4o mini mostrou-se competente para resolver problemas matemáticos e realizar manipulações simbólicas, características relevantes para aplicações que envolvem lógica e condições complexas. Além disso, o GPT-4o mini oferece o melhor custo-benefício entre

<sup>3</sup> Site da captura: <https://twitter.com/OpenAIDevs/status/1813990750851612830>

os modelos analisados, sendo significativamente mais acessível em todas as categorias de tokens, o que é vantajoso para projetos que exigem processamento constante de dados, como a geração de tabelas de decisão.

O fluxo de comunicação com a OpenAI segue etapas bem definidas. Inicialmente, o sistema recebe as condições e ações fornecidas pelos usuários, que descrevem estados possíveis e definem os comportamentos esperados para cada combinação dessas condições. Após o recebimento, o sistema realiza uma filtragem e validação dos dados, verificando se todas as informações necessárias foram corretamente fornecidas. Caso algum dado esteja incorreto ou incompleto, a aplicação emite mensagens de erro ou solicita informações adicionais.

Quando a validação é concluída, os dados são enviados para a API da OpenAI, que, por meio do modelo GPT-4-0-mini, gera uma tabela de decisão organizada com base nas condições e ações fornecidas. A tabela resultante exibe combinações de condições e suas respectivas ações. Essa resposta está estruturada em formato JSON e um processo de tratamento remove quaisquer caracteres indesejados, assegurando que a tabela gerada seja objetiva. Em seguida, ela é formatada para ser facilmente apresentada no front-end da aplicação.

Os custos da API da OpenAI são calculados com base no número de tokens consumidos, que podem ser palavras inteiras, partes de palavras ou até caracteres individuais. Com o dólar cotado a mais ou menos R\$ 5,94 (cinco reais e noventa e quatro centavos), o custo de uma requisição completa é de R\$ 0,0000041025 (aproximadamente quatro milionésimos de real). Os tokens são divididos de duas formas: entrada, que representa os tokens consumidos pelo texto enviado ao modelo; e saída, que são os tokens gerados pelo modelo em resposta à entrada.

Assim, o custo de uma requisição completa é o resultado da soma dos tokens consumidos na entrada e na saída. Para iniciar o desenvolvimento da Tabela.AI, foi feito um pagamento inicial de 10 dólares (aproximadamente sessenta reais) para ativar a comunicação com a API da OpenAI. Esse valor funciona como um crédito pré-pago, do qual o custo de cada requisição é automaticamente debitado.

O front-end, construído utilizando ReactJS, é responsável por proporcionar uma interface interativa e amigável para os usuários. Ele permite a inserção de condições e ações de forma dinâmica, além de exibir a tabela de decisão gerada automaticamente. Uma das funcionalidades principais do front-end é o formulário dinâmico, no qual os usuários podem adicionar ou modificar condições e ações conforme a necessidade.

Após o processamento no back-end, a tabela de decisão gerada é exibida no front-end, organizada de forma a facilitar o entendimento dos resultados. O layout responsivo da aplicação garante que os usuários possam interagir de maneira eficiente, seja em desktops ou dispositivos móveis, oferecendo uma visão clara das combinações de condições e ações e dos cenários de teste correspondentes.

## **7. Testes com usuários**

A fase de testes foi acompanhada por um questionário (Apêndice C), enviado junto a Tabela.AI, realizado com nove usuários, sendo sete profissionais da área de qualidade de software e dois desenvolvedores de software. Entre os participantes da pesquisa, seis

compartilharam comentários ou sugestões sobre a aplicação, dentre essas destacam-se duas sugestões, particularmente, relevantes:

1º inclusão de um ícone de ajuda para facilitar a inserção de condições e ações, fornecendo orientações claras aos usuários;

2º possibilidade de compartilhar os resultados gerados por meio de diferentes canais, como e-mail, aplicativos de mensagens instantâneas e redes sociais.

Ambas as sugestões foram avaliadas como pertinentes e, portanto, implementadas, visando o aprimoramento da experiência do usuário e otimização da usabilidade da ferramenta, além de atender de forma mais eficaz às necessidades dos usuários e garantir um desempenho mais eficiente do sistema.

## **8. Considerações finais**

O desenvolvimento da Tabela.AI alcançou os objetivos propostos, demonstrando que é possível otimizar de forma significativa o processo de criação de tabelas de decisão e cenários de teste. A aplicação permitiu a automação dessas etapas, proporcionando eficiência, economia de tempo e maior confiabilidade, garantindo uma cobertura mais abrangente nos testes.

A Tabela.AI mostrou-se como uma ferramenta valiosa para profissionais da área de qualidade de software, oferecendo agilidade e precisão, além de contribuir para a melhoria dos processos no dia a dia desses profissionais.

Os aprendizados adquiridos ao longo do desenvolvimento deste trabalho reforçam a importância de aliar tecnologia e usabilidade para resolver problemas práticos, destacando o potencial da inteligência artificial em otimizar processos de forma eficiente. A aplicação abre ainda possibilidades para o desenvolvimento de trabalhos futuros, como a integração com outras ferramentas e o aprimoramento de algoritmos para lidar com casos mais complexos, ampliando seu impacto e relevância na área de desenvolvimento e qualidade de software.

## REFERÊNCIAS

TCHEMRA, Angela Hum. **Tabela de Decisão Adaptativa na Tomada de Decisão Multicritério**. 2009. Tese (Doutorado em Engenharia de Computação e Sistemas Digitais) - Universidade de São Paulo, São Paulo, 2009. Disponível em: [https://www.teses.usp.br/teses/disponiveis/3/3141/tde-14082009-172031/publico/Te se\\_Tabela\\_de\\_Decisao\\_Adaptativa\\_na\\_Tomada\\_de\\_Decisao\\_Mul.pdf](https://www.teses.usp.br/teses/disponiveis/3/3141/tde-14082009-172031/publico/Te se_Tabela_de_Decisao_Adaptativa_na_Tomada_de_Decisao_Mul.pdf). Acesso em: 20 jun. 2024.

LIMA, Nathaelly da Silva. **Desenvolvimento de um Sistema de Suporte à Decisão com Base em Lógica Fuzzy e Inteligência Artificial para Análise de Risco em Projetos de Software**. 2023. Tese (Doutorado em Ciência da Computação) – Universidade Federal de Santa Catarina, Florianópolis, 2023. Disponível em: [https://repositorio.ufscar.br/bitstream/handle/ufscar/19407/TCC\\_Nathaelly\\_2023\\_VERSAO\\_FINAL.pdf?sequence=1&isAllowed=y](https://repositorio.ufscar.br/bitstream/handle/ufscar/19407/TCC_Nathaelly_2023_VERSAO_FINAL.pdf?sequence=1&isAllowed=y). Acesso em: 20 jun. 2024.

LACHTERMACHER, Luana. **O uso de tabelas de decisão para a automação da geração e da execução de casos de teste**. 2010. Dissertação (Mestrado em Informática) - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2010. Disponível em: [https://www.maxwell.vrac.puc-rio.br/16193/16193\\_1.PDF](https://www.maxwell.vrac.puc-rio.br/16193/16193_1.PDF). Acesso em: 02 jul. 2024.

HOPPE, Desireé Fernanda. **Ferramenta de Apoio ao Teste de Software Baseado na Tabela de Decisão**. 2015. 101 f. Dissertação (Mestrado em Sistemas de Informação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau. Disponível em: [https://www.furb.br/dsc/arquivos/tccs/monografias/2015\\_2\\_desiree-fernanda-hoppe\\_monografia.pdf](https://www.furb.br/dsc/arquivos/tccs/monografias/2015_2_desiree-fernanda-hoppe_monografia.pdf). Acesso em: 10 abr. 2024.

TEIXEIRA, Eva Mariana Pires. **Testes Funcionais em Sistemas Críticos**. 2024. Relatório de Estágio (Licenciatura em Engenharia Informática) – Departamento de Engenharia Informática, Universidade de Coimbra, Coimbra. Disponível em: <https://estudogeral.uc.pt/retrieve/275603/Relat%C3%B3rio%20de%20Est%C3%A1gio.pdf>. Acesso em: 20 nov. 2024.

OpenAI Developers. **GPT-4O mini scores highly on textual intelligence and reasoning (82% on MMLU), math, coding, and multimodal reasoning**. São Francisco, 18 jul. 2024. Twitter: @OpenAIDevs. Disponível em: <https://twitter.com/OpenAIDevs/status/1813990750851612830>. Acesso em: 28 nov. 2024.

DUARTE, Roberto Dias. **Desempenho do GPT-4.0: comparação entre modelos de IA de última geração**. Blog Roberto Dias Duarte. Belo Horizonte, 14 mai. 2024. Disponível em: <https://robertodiasduarte.com.br/desempenho-do-gpt-4o-comparacao-entre-modelos-de-ia-de-ultima-geracao/#gsc.tab=0>. Acesso em: 28 nov. 2024.

MADRUGA, Michel Pinheiro. **Investigação do uso de aprendizagem de máquina no fluxo de ataques a canais laterais em sistemas criptográficos**. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pelotas, Pelotas, 2024. Disponível em: [https://guaiaca.ufpel.edu.br/bitstream/handle/prefix/7719/Dissertacao\\_Michel\\_Pinheiro\\_Ma](https://guaiaca.ufpel.edu.br/bitstream/handle/prefix/7719/Dissertacao_Michel_Pinheiro_Ma)

druga.pdf?sequence=1&isAllowed=y. Acesso em: 29 nov. 2024.

## APÊNDICE A – PROTOTIPAGEM DA TABELA.AI

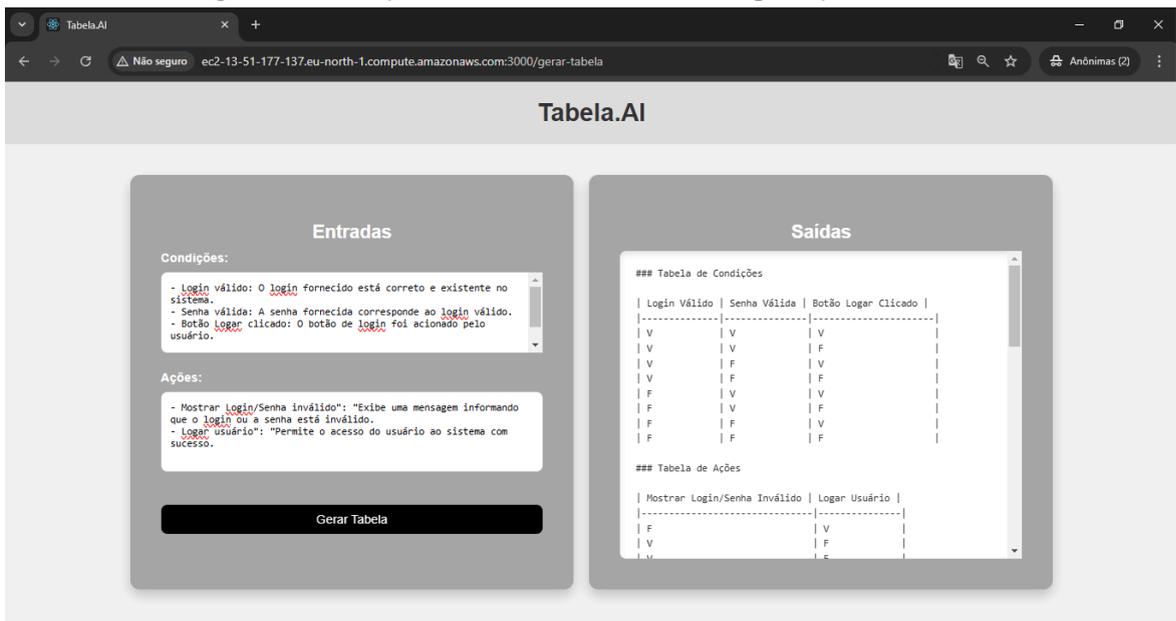
Figura 5 - Prototipagem de alta fidelidade da Tabela.AI



Fonte: Elaborado pelo autor (2024).

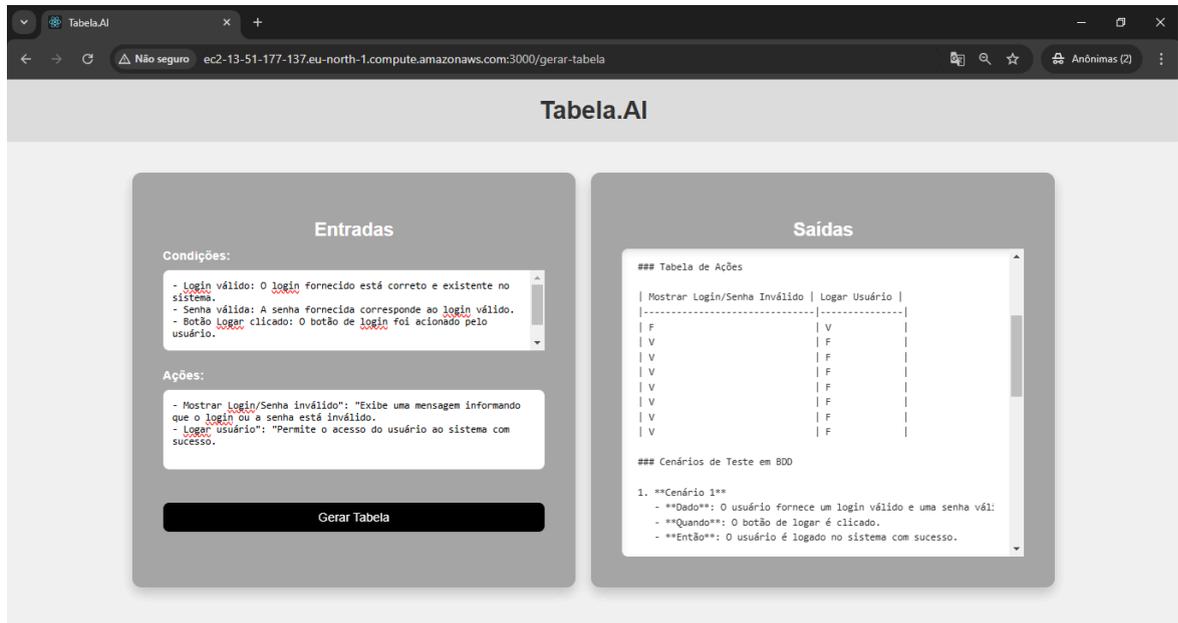
## APÊNDICE B – TABELA.AI

Figura 6 - Geração da tabela de decisão na aplicação Tabela.AI



Fonte: Capturado pelo autor (2024).

Figura 7 - Geração de cenários de testes na aplicação Tabela.AI



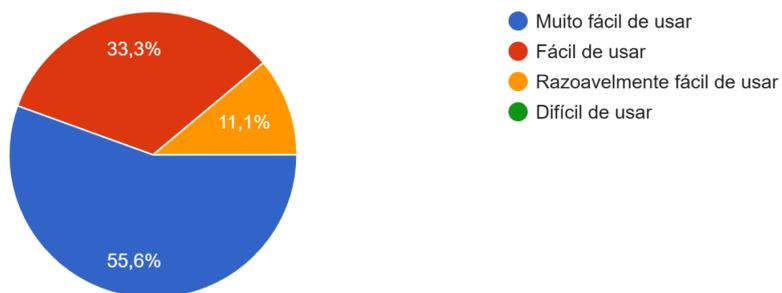
Fonte: Capturado pelo autor (2024).

## APÊNDICE C – Questionário e pesquisa

Figura 8 - Pesquisa de avaliação sobre a interface da Tabela.AI

Como você percebe a interface da aplicação Tabela.AI na inserção de condições e ações?

9 respostas

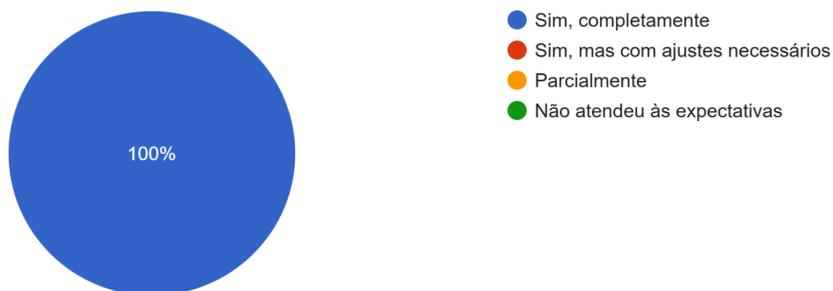


Fonte: Capturado pelo autor (2024).

Figura 9 - Pesquisa de avaliação sobre a precisão e organização da Tabela.AI

A tabela de decisão gerada pela Tabela.AI atendeu às suas expectativas em termos de precisão e organização?

9 respostas

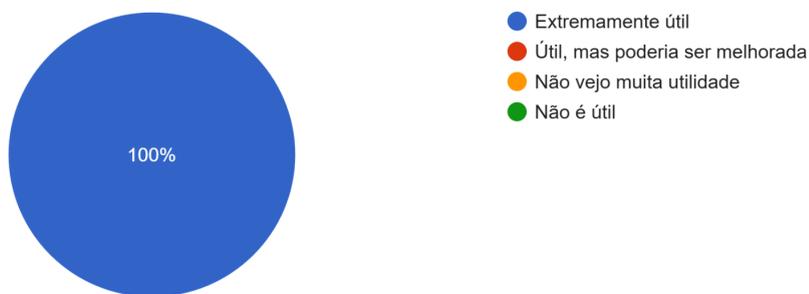


Fonte: Capturado pelo autor (2024).

Figura 10 - Pesquisa de avaliação sobre a integração com inteligência artificial na Tabela.AI

Como você avalia a integração com inteligência artificial na Tabela.AI para a criação de cenários de teste?

9 respostas

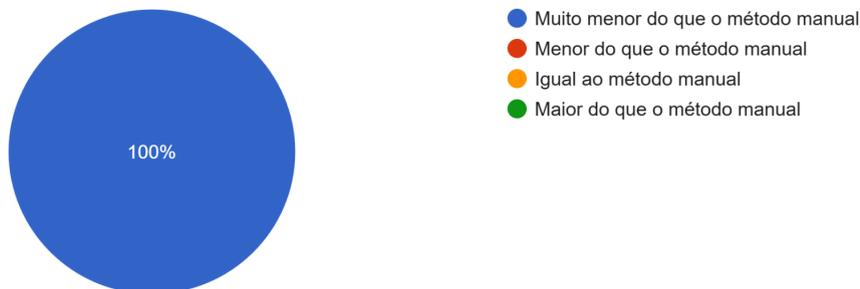


Fonte: Capturado pelo autor (2024).

Figura 11 - Pesquisa de avaliação sobre o tempo para criar tabelas de decisão na Tabela.AI

O tempo necessário para criar tabelas de decisão com a Tabela.AI é:

9 respostas



Fonte: Capturado pelo autor (2024).

Figura 12 - Comentários e sugestões da pesquisa de avaliação sobre a Tabela.AI

Você gostaria de compartilhar algum comentário ou sugestão sobre sua experiência? Fique à vontade para escrever.

6 respostas

A plataforma oferece uma facilidade incrível, proporcionando um processo dinâmico e ágil na criação de cenários. Com sua estrutura BDD (Banco de Dados Baseado em Dimensões), ela torna a elaboração de cenários muito mais eficiente e prática. A agilidade no desenvolvimento economiza tempo. Parabéns!!

Sugiro adicionar um "i" nos campos condições e ações dando exemplos de entradas de forma a orientar melhor o usuário.

A ferramenta agrega muito ao trabalho de criação de cenário de testes. É uma combinação inovadora, não apenas por otimizar o processo, mas também por abrir portas a um universo de novas possibilidades, trazendo soluções dinâmicas e eficientes.

Ficou bem fácil de usar

A ferramenta demonstrou-se excelente, sendo intuitiva e atendendo de forma eficaz à crescente demanda de empresas que adotam princípios ágeis em seus processos de desenvolvimento. Como sugestão de aprimoramento, recomendaria a inclusão de uma funcionalidade que permita compartilhar os resultados gerados por meio de diferentes canais, como e-mail, mensagens instantâneas, WhatsApp e redes sociais, a exemplo do Facebook. Parabenizo todos os envolvidos no desenvolvimento pela qualidade apresentada.

Excelente trabalho! Parabéns pela atenção aos detalhes e pela funcionalidade apresentada!

Fonte: Capturado pelo autor (2024).