

Análise do uso do TypeScript na aprendizagem de programação

**Nilson Rodrigo Borges de Sousa¹, Wesley Tiago Medeiros Lustosa¹,
Ana Rosa Pereira Chaves¹, Iallen Gábio de Sousa Santos¹**

¹Curso de Análise e Desenvolvimento de Sistemas – Instituto Federal do Piauí (IFPI)
Caixa Postal 64.260-000 – Piripiri – PI – Brasil

{nilsonrodrigues818, wesleyt584, rosaaanapc1000}@gmail.com, iallen@ifpi.edu.br

Abstract. *Programming education is essential for the development of logical thinking and problem-solving skills. However, beginners often choose dynamically typed languages, such as JavaScript, due to the lower initial effort required. This paper aims to analyze how TypeScript, a statically typed extension of JavaScript, can facilitate the programming learning process. The study adopts a qualitative approach, based on literature review and a questionnaire applied to students of the Analysis and Systems Development program at the Federal Institute of Piauí – Piripiri Campus. The results show that TypeScript's static typing promotes better code organization and clarity, contributing to early error detection and the development of good programming practices from the beginning.*

Resumo. *O ensino de programação é fundamental para o desenvolvimento do pensamento lógico e da resolução de problemas. Entretanto, iniciantes costumam optar por linguagens de tipagem dinâmica, como o JavaScript, por demandarem menor esforço inicial. Este artigo tem como objetivo analisar como o TypeScript, uma extensão do JavaScript com tipagem estática, pode facilitar o processo de aprendizagem de programação. A pesquisa adota uma abordagem qualitativa, baseada em revisão bibliográfica e aplicação de questionário com estudantes de Análise e Desenvolvimento de Sistemas do Instituto Federal do Piauí – Campus Piripiri. Os resultados evidenciam que a tipagem estática do TypeScript favorece a organização e a clareza do código, contribuindo para a identificação precoce de erros e o desenvolvimento de boas práticas desde os primeiros contatos com a programação.*

1. INTRODUÇÃO

A utilização das linguagens de programação é essencial na construção de plataformas digitais e na criação de sites modernos, desenvolvidos para atender às necessidades de clientes e empresas. Dentro da área de tecnologia, a compreensão e o domínio da Linguagem de codificação são uma das habilidades mais requisitadas dos profissionais. Mesmo para quem atua fora do setor, esse conhecimento é considerado um diferencial por causa do avanço de ferramentas tecnológicas e da digitalização nas organizações (FUNDAÇÃO ESTUDAR, 2022).

Nesse contexto, os programadores ou desenvolvedores assumem o papel de protagonistas na criação de sistemas automatizados, contribuindo significativamente para a

inclusão digital e o avanço da globalização. No entanto, torna-se fundamental o desenvolvimento sólido de competências e habilidades técnicas para programar, pois somente por meio do conhecimento é possível criar soluções funcionais, seguras e bem estruturadas.

Diante disso, Segundo a REDAÇÃO NA PRÁTICA (2022), as linguagens de programação ”são conjuntos de regras e instruções que um programador constrói para gerar programas e softwares que serão processados por um computador, dispositivo móvel ou outro equipamento”, podendo ser classificadas em baixo nível — comunicação direta com o processador, utilizando instruções simples e específicas, de difícil compreensão para humanos — e alto nível — com sintaxe próxima da linguagem humana, e que adotam tipagem estática ou dinâmica. Iniciantes, em geral, optam por linguagens de tipagem dinâmica, como o JavaScript, por exigirem menos esforço inicial, embora possa facilitar o manuseio, em códigos maiores, pode resultar em erros mais difíceis de identificar.

Sob esse ponto de vista, ”Todas as pessoas deveriam aprender a programar computadores, porque isso ensina a pensar- Steve Jobs ”(SILVEIRA;KNIRSCH, 2016). Dessa forma, conclui-se que a programação torna-se essencial não apenas para quem quer seguir carreira na área, mas também para o desenvolvimento do raciocínio lógico e da criatividade. No entanto, aprender a programar exige muito esforço e muitos estudantes enfrentam dificuldades, especialmente quando começam por linguagens de tipagem dinâmica, que apesar de parecerem mais simples por exigirem menos código, podem gerar erros e atrapalhar a aprendizagem.

Nesse sentido, a linguagem TypeScript ou TS, surge como uma alternativa interessante no processo de aprendizagem, facilitando a leitura e a compreensão do código, tornando o caminho da assimilação do conteúdo mais fácil e permitindo identificar erros de tipo em tempo de desenvolvimento, reduzindo a ocorrência de bugs e facilitando a manutenção do código Mororó (2024, p.19). Baseada em JavaScript (JS), o TS é conhecido como um superset, ou superconjunto do JavaScript — ou seja, todo código JavaScript é válido em TypeScript, mas o contrário não é verdade. A principal diferença está na tipagem estática do TS, o que ajuda iniciantes a criarem variáveis não mutáveis e praticarem boas práticas de programação desde o começo.

O presente artigo discute como a linguagem TypeScript — um superset do JavaScript que adiciona tipagem estática — pode contribuir para o processo de aprendizagem, especialmente nos estágios iniciais da formação em tecnologia. O estudo focaliza os alunos do 1º e do 3º módulo do curso de Análise e Desenvolvimento de Sistemas (ADS) do Instituto Federal de Educação, Ciência e Tecnologia do Piauí — *campus* Piripiri. Os dados sobre as percepções dos estudantes acerca do TypeScript e seus desafios foram coletados por meio de um questionário aplicado aos alunos.

A escolha de concentrar o ensino na linguagem TypeScript se justifica pelo enfoque do curso, que prioriza o domínio aprofundado de uma linguagem específica. Embora aprender várias linguagens ofereça vantagens e desvantagens, o conhecimento sólido de uma única linguagem é considerado mais relevante para o desenvolvimento acadêmico e profissional dos estudantes, permitindo que eles construam uma base consistente para futuros aprendizados e aplicações práticas.

2. REFERENCIAL TEÓRICO

2.1. Fundamentos do JavaScript

Antes de falar sobre o TS, é necessário entender de onde surgiu e em que está embasada a sua história entendendo quais foram os caminhos até a criação. Assim, o JS é uma linguagem de programação criada em 1995 por Brendan Eich, originalmente com o nome de LiveScript. Sua proposta era permitir maior interatividade nas páginas web, executando-se diretamente no navegador sem necessidade de compilação. Com o passar dos anos, tornou-se uma das principais tecnologias do desenvolvimento front-end, sendo usada para manipular o comportamento de elementos HTML, realizar requisições assíncronas e construir aplicações web completas .

2.2. Desafios do Javascript

A linguagem JavaScript impõe certos desafios aos desenvolvedores, sobretudo por adotar uma tipagem dinâmica, que pode ocasionar comportamentos inesperados e dificultar a manutenção do código. Conforme destaca Mororó (2024, p.25):

”A tipagem dinâmica também pode apresentar desafios, especialmente em projetos grandes e complexos, nos quais o controle estrito dos tipos pode ser desejado para evitar erros. É importante ter cuidado ao lidar com tipos em JavaScript para garantir que as operações sejam realizadas corretamente e que os valores sejam coerentes com o esperado”.

Dessa forma, torna-se necessário discutir os desafios associados à linguagem. Embora o JavaScript seja amplamente utilizado e apresente vantagens como a portabilidade, sua aplicação em sistemas de grande porte pode ser problemática. A ausência de tipagem estática aumenta a tendência a erros inesperados, o que impacta tanto o desenvolvimento quanto a experiência de aprendizado de programadores iniciantes.

2.3. Fundamentos do TypeScript

TypeScript é uma linguagem de programação de código aberto baseada em JavaScript, criada pela Microsoft. Ela adiciona tipagem estática e recursos avançados à linguagem JavaScript, ajudando os desenvolvedores a escrever código mais seguro, robusto e fácil de manter sendo amplamente utilizada em desenvolvimentos de aplicações Web, tanto no front-end quanto no back-end (MORORÓ, 2024)

Foi lançado em 2012 como um projeto de código aberto embasada nas especificações da *ECMAScript*⁵, sendo a mesma base do JavaScript. Ao escrever um código em TS ele compila todo Script para JS, e o compilador acrescenta no código o comando *”use strict”* impedindo que variáveis sejam usadas sem antes delas serem declaradas. O TypeScript é amplamente adotado em projetos de grande escala, nos quais a manutenção e a escalabilidade são essenciais (MORORÓ,2024).

Em termos de analogia, pense no JS como um carro antigo, funcional, mas que exige manutenção constante. Já o TS seria um carro moderno, com maior conforto e segurança. Ele auxilia os programadores na adoção de boas práticas e evita erros antes mesmo que o código seja executado.

2.4. Vantagens e desafios

Utilizar TypeScript no ensino de programação traz benefícios significativos. A linguagem oferece:

- **Tipagem estática** - cada variável tem seu tipo definido e verificado durante a compilação, permitindo a detecção antecipada de inconsistências.
- **Estabilidade na execução dos programas** - os erros são identificados antes da execução, reduzindo falhas inesperadas em tempo de runtime.
- **Clareza e legibilidade do código** - a definição explícita de tipos facilita a compreensão e manutenção dos programas.
- **Supporte à programação orientada a objetos** - possibilita o uso de classes, interfaces e herança, auxiliando na organização de sistemas complexos.
- **Manutenção facilitada em projetos grandes** - a tipagem e as interfaces tornam mais simples modificar, atualizar ou expandir o código sem introduzir novos erros.
- **Integração com bibliotecas JavaScript existentes** - permite utilizar bibliotecas populares de JavaScript com suporte a tipos, mantendo compatibilidade e segurança.
- **Autocompletar e auxílio do editor** - ferramentas como VS Code oferecem sugestões inteligentes e documentação inline, aumentando a produtividade.
- **Melhora na colaboração em equipe** - o código tipado é mais previsível, facilitando o entendimento entre diferentes programadores.

Um desafio relevante do TypeScript está na necessidade de sua compilação para JavaScript antes da execução do programa. Esse processo pode tornar o desenvolvimento mais lento e o fluxo de trabalho mais complexo, especialmente em projetos de pequeno porte ou protótipos. Adicionalmente, a integração com bibliotecas JavaScript existentes nem sempre é simples, já que muitas delas não fornecem definições de tipos completas. Nesses casos, é necessário que o desenvolvedor crie ou adapte manualmente declarações de tipo para assegurar a correta tipagem durante a compilação.

3. METODOLOGIA

Este trabalho adota uma abordagem qualitativa e exploratória, com base em duas estratégias principais: a revisão bibliográfica e a aplicação de um questionário. Inicialmente, foi realizada uma pesquisa bibliográfica sobre o ensino de programação, o uso de linguagens de tipagem estática e dinâmica, e o papel do TypeScript no processo de aprendizagem. Foram consultados artigos acadêmicos, materiais técnicos e fontes digitais confiáveis para fundamentar a discussão teórica e contextualizar a proposta do estudo.

Além disso, com o objetivo de compreender a percepção dos estudantes sobre o uso do TS no aprendizado de programação, foi elaborado e aplicado um questionário online, composto por 9 (nove) questões objetivas e uma questão aberta. A pesquisa contou com 15 (quinze) participantes, sendo a maioria composta por iniciantes na área de programação. Os dados coletados abordaram temas como a linguagem de entrada, experiência com o TS, comparação com o JS, dificuldades enfrentadas e opinião sobre sua aplicabilidade no processo de aprendizagem.

Sob essa perspectiva, buscou-se evidenciar como a linguagem TS pode contribuir para uma aprendizagem mais estruturada focada na organização de código, facilitando a vida de um programador e minimizando erros comuns, com o objetivo de promover boas

práticas desde o início da formação. Os resultados foram organizados em gráficos, que estão distribuídos ao longo do artigo para facilitar a visualização e análise das respostas. As interpretações obtidas por meio do questionário foram utilizadas para reforçar os argumentos apresentados nas seções de análise e discussão.

4. RESULTADOS E DISCUSSÕES

4.1. Percepções dos Estudantes sobre o Uso do TypeScript

A maioria dos participantes (53,3%) indicou o TS como a primeira linguagem de programação aprendida, seguido por Python com 33,3% (Figura 1). Esse dado reforça a crescente adoção do TypeScript por iniciantes, ainda que não seja tradicionalmente ensinado como linguagem introdutória.

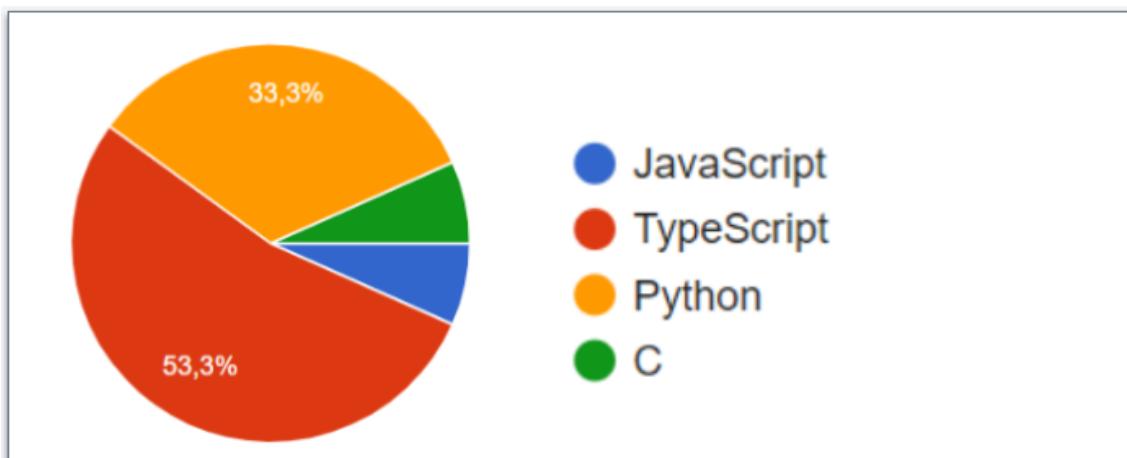


Figura 1. Primeira linguagem de programação

Fonte: Elaborado pelos autores (2025).

Além das informações descritivas coletadas, uma das questões objetivas do questionário buscou avaliar diretamente a percepção dos participantes quanto ao papel da tipagem estática no processo de aprendizagem. A pergunta formulada foi: *Em sua opinião, a tipagem estática do TypeScript facilita ou dificulta o aprendizado?*. A Figura 2 apresenta a distribuição

Os dados revelam que 80% dos participantes consideram que a tipagem estática **facilita** o aprendizado, enquanto 20% acreditam que ela tanto facilita quanto dificulta. Nenhum dos participantes indicou que a tipagem dificulta ou que não saberia opinar. Esse resultado aponta para uma percepção predominantemente positiva da tipagem estática como um recurso didático, capaz de favorecer a clareza, a organização e a detecção precoce de erros durante o processo de aprendizagem em programação.

Dessa forma, com intuito de analisar se o TypeScript ajuda a entender melhor os erros no código foi elaborado outra pergunta - *Comparado ao JavaScript, o TypeScript ajuda a entender melhor os erros no código?*. Os dados foram que 80% dos participantes afirmaram que ajuda bastante. Esse resultado reforça a ideia de que a tipagem estática e os mecanismos de verificação antecipada de erros do TypeScript oferecem maior clareza ao programador durante o desenvolvimento. Ao contrário do JavaScript, que permite maior



Figura 2. Opiniões dos alunos sobre se o TypeScript facilita o aprendizado

Fonte: Elaborado pelos autores (2025).

flexibilidade mas pode ocultar erros até o momento da execução, o TypeScript se destaca por alertar sobre inconsistências ainda durante a codificação, o que pode beneficiar especialmente iniciantes que estão desenvolvendo suas habilidades de depuração e lógica.

Tais características tornam o TypeScript uma linguagem potencialmente mais instrutiva para estudantes em fase inicial, pois contribui para o desenvolvimento de boas práticas de programação e estimula o raciocínio em torno da estrutura e coerência do código.

No entanto, por meio do formulário aplicado, buscou-se compreender, a partir da perspectiva dos estudantes, em quais partes do código eles enfrentavam (ou imaginavam enfrentar) maiores dificuldades ao utilizar TypeScript. Com esse intuito, foi elaborada a seguinte pergunta: *“Quais dificuldades você enfrentou (ou imagina enfrentar) ao utilizar TypeScript?”* A Figura 3 ilustra os principais desafios mencionados, os quais serão discutidos a seguir.

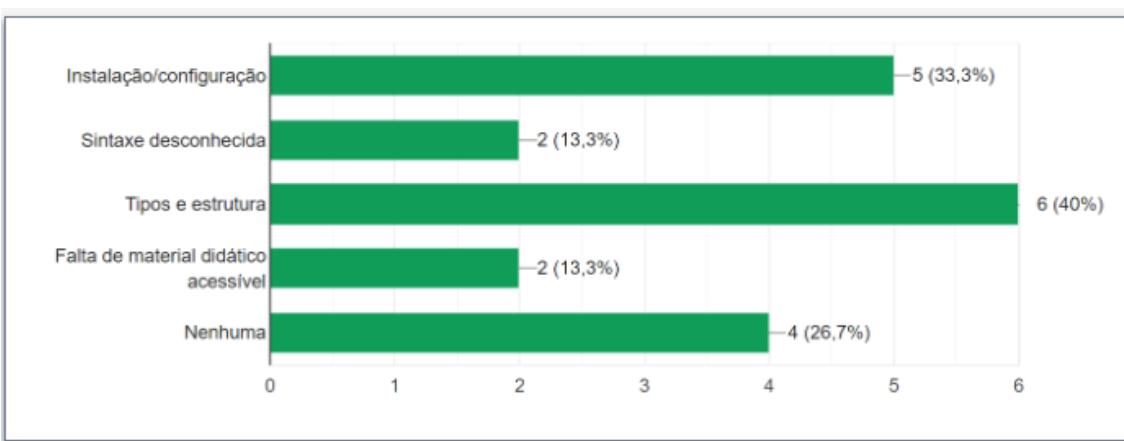


Figura 3. Dificuldades na utilização do TypeScript

Fonte: Elaborado pelos autores (2025).

4.2. Análise das dificuldades encontradas

Outro ponto abordado no questionário buscou identificar as principais dificuldades enfrentadas (ou previstas) pelos estudantes ao utilizarem TypeScript. Conforme demonstrado na Figura 3, a dificuldade mais mencionada foi relacionada a “Tipos e estrutura”, com 40% das respostas. Esse resultado é coerente com o fato de o TypeScript introduzir conceitos de tipagem estática e estruturas mais rígidas que o JavaScript, exigindo maior compreensão da sintaxe e da organização do código.

A “Instalação e configuração” também foi apontada como uma barreira significativa por 33,3% dos participantes. Esse dado revela um desafio técnico inicial que pode desmotivar iniciantes, principalmente quando não há acompanhamento ou materiais claros de apoio. Outros fatores mencionados foram a “Sintaxe desconhecida” e a “Falta de material didático acessível”, cada um representando 13,3% das respostas. Tais aspectos indicam que, apesar de TS oferecer recursos poderosos para o desenvolvimento, sua curva de aprendizagem pode ser acentuada se o estudante não contar com recursos didáticos adequados.

Por outro lado, 26,7% dos respondentes afirmaram não ter encontrado dificuldades com a linguagem, o que pode estar relacionado à familiaridade prévia com JavaScript ou ao uso de ferramentas que facilitam a transição entre as duas linguagens. Esse dado reforça a importância de se considerar o contexto de aprendizagem e o nível de experiência dos estudantes ao introduzir o TypeScript em ambientes educacionais.

4.3. Sugestões dos Alunos sobre o Uso do TypeScript no Ensino de Programação

Além das percepções e dificuldades relatadas, o questionário aplicado também trouxe importantes sugestões e considerações dos alunos sobre a utilização do TypeScript no processo de aprendizagem de programação.

Os estudantes destacaram que o TypeScript oferece uma base sólida para o ensino, ao combinar a flexibilidade do JavaScript com a segurança da tipagem estática. Eles ressaltaram que, no contexto educacional, o TypeScript facilita a compreensão de conceitos fundamentais como tipos de dados, interfaces, orientação a objetos e modularização, promovendo uma organização mais clara e estruturada do código. Segundo os alunos, essa combinação incentiva um pensamento mais cuidadoso sobre a estrutura do programa e as possíveis falhas de execução, promovendo desde o início uma cultura de prevenção de erros.

Outro ponto citado é que o TS, por compilar para JavaScript, também possibilita que os estudantes entendam melhor o processo de compilação e a transformação do código para execução no ambiente real, ampliando a visão sobre o funcionamento das linguagens de programação. Os participantes também enfatizaram que o TypeScript é uma linguagem dinâmica e tipada que promove um aprendizado crítico, ao exigir que o programador seja organizado e saiba estruturar seu código, o que é fundamental para a formação de profissionais preparados para o mercado.

Alguns consideraram o TypeScript uma excelente porta de entrada para o mundo da programação, especialmente pela didática simples e eficaz adotada em sala de aula. Ainda, foi sugerido que o uso do TS poderia ser ampliado no currículo, com a inclusão de mais conteúdos que possibilitem aos alunos compreenderem melhor a dimensão e

aplicabilidade da linguagem no desenvolvimento web e em projetos reais. De modo geral, os relatos dos alunos refletem uma percepção positiva quanto ao uso do TypeScript no ensino de programação, destacando sua eficiência e importância para a construção de uma base sólida no aprendizado.

5. CONSIDERAÇÕES FINAIS

A utilização do TypeScript no ensino de programação mostrou-se uma alternativa eficaz para o desenvolvimento de habilidades essenciais entre estudantes iniciantes. Ao impor a tipagem estática, a linguagem favorece a organização e a legibilidade do código, além de auxiliar na identificação precoce de erros, promovendo um aprendizado mais estruturado e reflexivo.

Os dados obtidos a partir do questionário aplicado evidenciam que a maioria dos estudantes reconhece os benefícios do TypeScript, especialmente no que diz respeito à depuração de código e ao entendimento da estrutura de dados e funções. Embora tenham sido relatadas dificuldades iniciais com configuração do ambiente e sintaxe, essas barreiras são superadas à medida que o estudante comprehende o funcionamento da linguagem e passa a enxergar suas vantagens em relação ao JavaScript puro.

Portanto, os resultados reforçam a ideia de que a introdução do TypeScript desde as fases iniciais da formação acadêmica pode contribuir significativamente para a formação de profissionais mais preparados para o mercado de trabalho, com maior domínio de boas práticas e maior autonomia no desenvolvimento de sistemas.

Referências

- Microsoft (2025). Typescript playground. Site, disponível em: <https://www.typescriptlang.org/>. Acesso em: 11 jul. 2025.
- Mororó, J. F. (2024). Um estudo comparativo entre javascript e typescript. Relatório técnico / manuscrito, autor independente.
- REDAÇÃO NA PRÁTICA (2022). Linguagem de programação: conheça os principais conceitos e ferramentas desse universo. Site, disponível em: <https://napratica.org.br/noticias/linguagem-de-programacao>. Acesso em: 21 out. 2025.
- Silveira, L. P. N. and Knirsch, J. F. (2016). Como a programação pode auxiliar no desenvolvimento do raciocínio lógico em crianças, adolescentes e jovens. In *Uma Nova Pedagogia para a Sociedade Futura*, pages 734–737.