

PortuJava - Uma ferramenta de auxílio a aprendizagem de programação

Matheus P. G. Santos¹, Carlos A. Silva¹

¹Departamento de Informática – Instituto Federal de Minas Gerais (IFMG)
CEP 34.590-390 – Sabará, MG – Brasil

matpirespires@gmail.com, carlos.silva@ifmg.edu.br

Abstract. *This paper describes the development of the PortuJava tool, designed to assist programming learning by facilitating the transition from Portugol to Java. Given the gap between the demand for IT professionals and the number of graduates, there is a growing need for methods that ease programming education. PortuJava allows the conversion of code written in Portugol into Java, addressing syntax and language differences often encountered by students when using professional languages. The tool is presented as a web page, developed with the Angular framework, and integrates the JDoodle IDE Plugin for Java code visualization and execution. Initial results, obtained through a questionnaire, indicate that the tool is well-received by students, being highlighted as a practical solution for adapting to Java syntax and development within professional IDEs.*

Resumo. *O presente trabalho descreve o desenvolvimento da ferramenta PortuJava, criada para auxiliar a aprendizagem de programação por meio da transição entre as linguagens Portugol para Java. Dada a discrepância entre a demanda de profissionais de TI e a quantidade de formandos, há uma crescente necessidade de métodos que facilitem o ensino de programação. O PortuJava permite converter códigos escritos em Portugol para Java, abordando as diferenças de sintaxe e idioma, situação comumente ocorrida a alunos que se deparam com linguagens profissionais. A ferramenta é apresentada como uma página web, desenvolvida com o framework Angular, e utiliza o JDoodle IDE Plugin para visualização e execução dos códigos em Java. Os resultados, obtidos por meio de um questionário, indicam que a ferramenta é bem recebida pelos estudantes, sendo destacada como uma solução prática para adaptação à sintaxe de Java e ao desenvolvimento em IDEs profissionais.*

1. Introdução

O ensino de programação tem recebido bastante atenção nos últimos anos. Com o avanço da tecnologia e com a ampliação do acesso a dispositivos e a internet, o impacto da programação de computadores tem se acentuado tanto no mercado de trabalho quanto na área educacional. De acordo com a Associação das Empresas de Tecnologia da Informação e Comunicação e de Tecnologias Digitais (Brasscom) haverá uma demanda de quase 800 mil profissionais de área de Tecnologia da Informação (TI) no Brasil de 2021 a 2025, sendo que o país forma apenas 53 mil profissionais da área por ano [Brasscom 2021].

Diante da crescente demanda por profissionais da área de tecnologia e da importância da programação para o desenvolvimento de diversas habilidades, a busca por métodos inovadores para o ensino dessa disciplina se torna cada vez mais urgente. No entanto, as metodologias tradicionais muitas vezes encontram dificuldades em atender às necessidades dos atuais alunos.

Nesse contexto, o desenvolvimento de plataformas digitais interativas desponta como uma alternativa promissora para tornar o aprendizado de programação mais eficaz e acessível, proporcionando aos estudantes um ambiente dinâmico e personalizado. Diversas pesquisas e iniciativas têm buscado atenuar as dificuldades enfrentadas pelos alunos, com o desenvolvimento de ferramentas específicas para o ensino de programação, como a IDE proposta por [Noschang et al. 2014] para auxiliar na aprendizagem de algoritmos. Embora essas ferramentas representem um avanço, ainda há a necessidade contínua de inovação e adaptação das metodologias de ensino.

A diversidade de ferramentas e linguagens de programação, cada uma com sua sintaxe e paradigmas específicos, representa um desafio considerável para os aprendizes. A necessidade de adaptar-se constantemente a novos ambientes e comandos, muitas vezes em inglês, impacta diretamente na capacidade dos alunos de aplicar os conhecimentos adquiridos em contextos diferentes.

Além da dificuldade em adaptar o conhecimento adquirido em ferramentas introdutórias, outros obstáculos como a compreensão de erros no código e a complexidade das IDEs (*Integrated Development Environments*) dificultam significativamente o aprendizado de novas linguagens de programação. Esses desafios, frequentemente encontrados na transição para linguagens como Java, podem desencorajar muitos estudantes.

Estudos como [Oliveira et al. 2014, Fernandes and Junior 2016, Medeiros 2019] evidenciam as dificuldades enfrentadas pelos estudantes de programação, com taxas de reprovação superiores a 45% e altas taxas de evasão. Apesar do apoio de ferramentas introdutórias, os estudantes ainda enfrentam desafios consideráveis, como a compreensão de laços de repetição e estruturas condicionais. No entanto, o uso dessas ferramentas é fundamental, pois contribui significativamente para a aprendizagem e reduz as taxas de reprovação, como apontado por [Junior and França 2017].

Com o objetivo de facilitar a transição dos estudantes de programação do Portugol para Java, uma linguagem amplamente utilizada no mercado de trabalho, desenvolvemos uma ferramenta online que converte automaticamente códigos escritos em Portugol para sua versão equivalente em Java. Essa ferramenta permite aos alunos comparar as duas linguagens e entender as diferenças sintáticas e semânticas entre elas.

O Portugol, presente em diversos materiais didáticos, como aponta a pesquisa literária de [Zanini and Raabe 2012], é uma linguagem de programação propícia na formação inicial de profissionais da área. Sua simplicidade e a utilização da língua portuguesa facilitam a compreensão dos conceitos fundamentais da programação, contribuindo para a construção de uma base sólida para o aprendizado de linguagens mais complexas, conforme corroborado por [Cristovão 2008].

A linguagem Java, desenvolvida pela Sun Microsystems e lançada em 1995, destaca-se por sua versatilidade, sendo empregada em diversas áreas como desenvolvimento de jogos, computação em nuvem, big data e inteligência artificial. Conforme

descrito em [Indrusiak 1996], o Java é caracterizado por sua completude, simplicidade, facilidade de uso e poder. Essa avaliação é corroborada por pesquisas como a do IEERanking [Cass 2022], que a posiciona como a quinta linguagem mais utilizada em 2022 e a segunda com maior demanda no mercado de trabalho mundial.

O trabalho está organizado da seguinte forma. A seção 2 apresenta a justificativa para o desenvolvimento da ferramenta proposta. Em seguida, a seção 3 descreve os objetivos gerais e específicos da pesquisa. A seção 4 expõe uma revisão da literatura científica relacionada. A seção 5 aborda a metodologia empregada na construção da ferramenta, detalhando as etapas e as ferramentas utilizadas. Na seção 6, o desenvolvimento da ferramenta é descrito, abordando as fases de entrada de dados, processamento e visualização do código convertido. A seção 7 analisa os resultados obtidos a partir do uso e da avaliação da ferramenta, enquanto a seção 8 conclui o trabalho, ressaltando os benefícios da ferramenta para o ensino de programação e propondo direções futuras para seu aprimoramento e ampliação.

2. Justificativa

Ao analisar casos como o de [Oliveira et al. 2014], nota-se que os alunos enfrentam grandes dificuldades no aprendizado de programação, especialmente em estruturas de repetição e condicionais, com uma taxa de reprovação média de 46%. O estudo de [Fernandes and Junior 2016] apresenta resultados semelhantes: entre 2010 e 2015, as disciplinas de Linguagem de Programação I e II no Campus Sombrio, do Instituto Federal Catarinense, obtiveram apenas 58% de aprovação, com índices de evasão e reprovação de 26% e 16%, respectivamente. Na Universidade do Ceará, a taxa de aprovação também é alarmante, atingindo apenas 49,6% dos estudantes. De forma similar, [Medeiros 2019] aponta elevadas taxas de reprovação em disciplinas de programação, reforçando a necessidade de alternativas pedagógicas que amenizem esses índices.

Embora os estudos de [Oliveira et al. 2014, Fernandes and Junior 2016, Medeiros 2019] indiquem altas taxas de reprovação em disciplinas de programação, é importante destacar que diferentes ferramentas e metodologias podem influenciar positivamente o aprendizado dos alunos. Conforme demonstrado em [Gomes et al. 2008], a utilização de recursos adequados pode levar a melhorias no desempenho, mesmo que as taxas de reprovação e repetência não apresentem reduções significativas.

Diante das dificuldades enfrentadas no aprendizado de disciplinas introdutórias de programação e das elevadas taxas de reprovação, este trabalho propõe uma ferramenta para auxiliar estudantes que iniciaram seus estudos com Portugol a realizar a transição para a linguagem Java. Segundo [Indrusiak 1996], Java é uma linguagem robusta, acessível e versátil, que apresenta alta demanda e ampla aplicação no mercado, ocupando posição de destaque entre as linguagens mais requisitadas [Cass 2022].

3. Objetivos

3.1. Geral

Desenvolver uma ferramenta que auxilie estudantes na transição de seus conhecimentos da linguagem Portugol para a linguagem Java, facilitando o aprendizado e adaptação a essa nova linguagem.

3.2. Específicos

- Implementar uma ferramenta que apoie os alunos na adaptação e aprendizado de uma linguagem profissional.
- Estabelecer correspondências entre os principais comandos das linguagens Portugol e Java, simplificando a compreensão das diferenças sintáticas.
- Aplicar conhecimentos adquiridos no curso de Sistemas de Informação em um projeto prático e útil para o ensino de programação.

4. Revisão Bibliográfica

No artigo de [Noschang et al. 2014], é apresentado um ambiente integrado de desenvolvimento (IDE) voltado para iniciantes em programação. Os autores destacam que muitas ferramentas profissionais podem ser desafiadoras para os estudantes, pois frequentemente são desenvolvidas em inglês, apresentam configurações avançadas e exigem conhecimentos prévios para utilização eficaz. Para mitigar essas dificuldades, foi desenvolvida uma IDE específica para a linguagem Portugol, utilizando um compilador criado em um trabalho anterior. Desde seu lançamento, em 2013, até a escrita do artigo, a ferramenta já havia registrado mais de 2000 downloads, mesmo com divulgação limitada a pequenos grupos acadêmicos. A IDE também estava em uso em duas universidades, com diversos trabalhos de conclusão de curso e dissertações de mestrado baseados neste projeto.

No artigo de [Medeiros 2015], é apresentado um ambiente de desenvolvimento com interpretador integrado. Diferente do trabalho mencionado anteriormente, esta ferramenta não exige instalação, pois é executada diretamente pela internet, embora também ofereça uma versão para uso *offline*. A publicação do site, divulgada amplamente nas redes sociais, permitiu aos autores avaliarem a experiência dos usuários, que, em geral, se mostrou positiva. Inclusive, houve o relato de um defeito, prontamente corrigido pelo autor.

Uma proposta didático-pedagógica para o ensino de programação é apresentada por [Wanderley et al. 2020], utilizando um jogo de tabuleiro estruturado em três modalidades. Duas dessas modalidades exploram conceitos técnicos e o uso das linguagens Portugol e Java, enquanto a terceira foca no desenvolvimento de lógica, abstração e sistematização. O tabuleiro foi projetado para ser flexível, permitindo a alteração das casas de ação e possibilitando adaptações conforme o conteúdo abordado pelo docente. Os testes indicaram que o jogo oferece uma experiência dinâmica e lúdica, que se mostrou eficaz em motivar os alunos. Com esses resultados, os autores projetam uma possível redução nas taxas de reprovação e evasão, reforçando a importância de alternativas pedagógicas para apoiar o aprendizado dos discentes.

A pesquisa de [Junior and França 2017] evidencia a necessidade de aprimorar as ferramentas utilizadas no ensino de programação. Ao comparar diferentes ferramentas, os autores identificaram algumas deficiências comuns, como a limitação linguística e a falta de recursos para despertar o interesse e a motivação dos alunos. Apesar de reconhecerem o potencial das ferramentas analisadas, os autores sugerem que futuras pesquisas se concentrem no desenvolvimento de ferramentas mais completas e eficazes, capazes de atender às necessidades específicas dos iniciantes em programação.

Tanto o e-book de [Chiara 2014] quanto o material de [Carmona 2014] abordam a equivalência entre os comandos do Portugol e do Java. Enquanto o material de Car-

mona oferece uma visão geral do tema, o e-book de Ramon se aprofunda na comparação, apresentando uma descrição mais completa e detalhada. A obra de Ramon abrange uma variedade de tópicos, como tipos de dados, operações e estruturas de controle, tornando-se um material de referência indispensável para quem deseja dominar a relação entre as duas linguagens.

O processo de aprendizado de programação pode ser desafiador, especialmente no que diz respeito à compreensão das diferentes sintaxes das linguagens de programação. A ferramenta desenvolvida neste trabalho tem como objetivo otimizar esse processo ao oferecer uma ponte entre o Portugol, uma linguagem comumente utilizada em cursos introdutórios, e o Java, uma linguagem amplamente utilizada no mercado de trabalho. Ao apresentar de forma comparativa as estruturas e comandos das duas linguagens, a ferramenta facilita a transição do aluno para um nível mais avançado de programação.

5. Metodologia

A fim de alcançar os objetivos estabelecidos, foi realizado o desenvolvimento de um site utilizando o framework Angular. A construção da interface do usuário foi baseada em HTML, CSS e JavaScript, proporcionando uma experiência visualmente agradável e interativa. O framework Angular foi utilizado para estruturar o código e gerenciar a lógica da aplicação.

5.1. Configurações de ambiente

Nessa seção são descritas e listadas as tecnologias utilizadas no desenvolvimento do site/plataforma. Como ferramenta de desenvolvimento foi utilizado o Angular, um framework robusto que facilita a criação de interfaces dinâmicas e responsivas. Para o controle de versão do código, utilizou-se o Git, que permite um gerenciamento eficiente das alterações e colaborações no projeto. Também foi utilizado o *plugin* da IDE de Java disponibilizado pelos criadores do site JDoodle, recurso essencial para a visualização e execução do código Java, proporcionando uma experiência prática aos usuários ao permitir que eles testem suas implementações diretamente na plataforma. Também será apresentado o diagrama do produto, que ilustra a arquitetura e os componentes principais da aplicação, facilitando a compreensão do fluxo de dados e das interações entre as diversas partes do sistema.

- **JDoodle IDE Plugin:** Uma solução prática e gratuita para integrar a experiência de programação diretamente em páginas web. Com ele, é possível editar e executar códigos Java utilizando diversas versões do JDK e bibliotecas do Apache Maven, tudo isso sem sair do navegador. O *plugin* oferece um ambiente de desenvolvimento rápido e eficiente, permitindo visualizar os resultados da execução e identificar erros de forma imediata. Embora algumas funcionalidades como salvar, baixar ou carregar arquivos localmente não estejam disponíveis diretamente no *plugin*, os usuários podem facilmente enviar suas edições para o site oficial do JDoodle para acessar essas opções adicionais
- **Angular 18.1.0:** Angular é uma plataforma de desenvolvimento construída em *TypeScript*. Sua estrutura é baseada em componentes para criar aplicativos web escalonáveis. Além disso, é possível adicionar diversas bibliotecas que abrangem uma ampla variedade de recursos, como roteamento, gerenciamento de formulários,

comunicação entre cliente e servidor, dentre outros [Google]. Essa ferramenta foi escolhida por sua simplicidade, facilidade de uso e excelente desempenho.

- **Git 2.33.0:** Git é um sistema de controle de versão distribuído gratuitamente e de código aberto. Pode-se utilizá-lo em diversos tipos de projetos, sejam pequenos ou mais complexos com um alto desempenho, rapidez e eficiência. Para facilitar o armazenamento das versões e servir também com uma forma de backup do projeto, foi utilizado o GitHub em conjunto com o Git.

5.2. Diagrama de produto

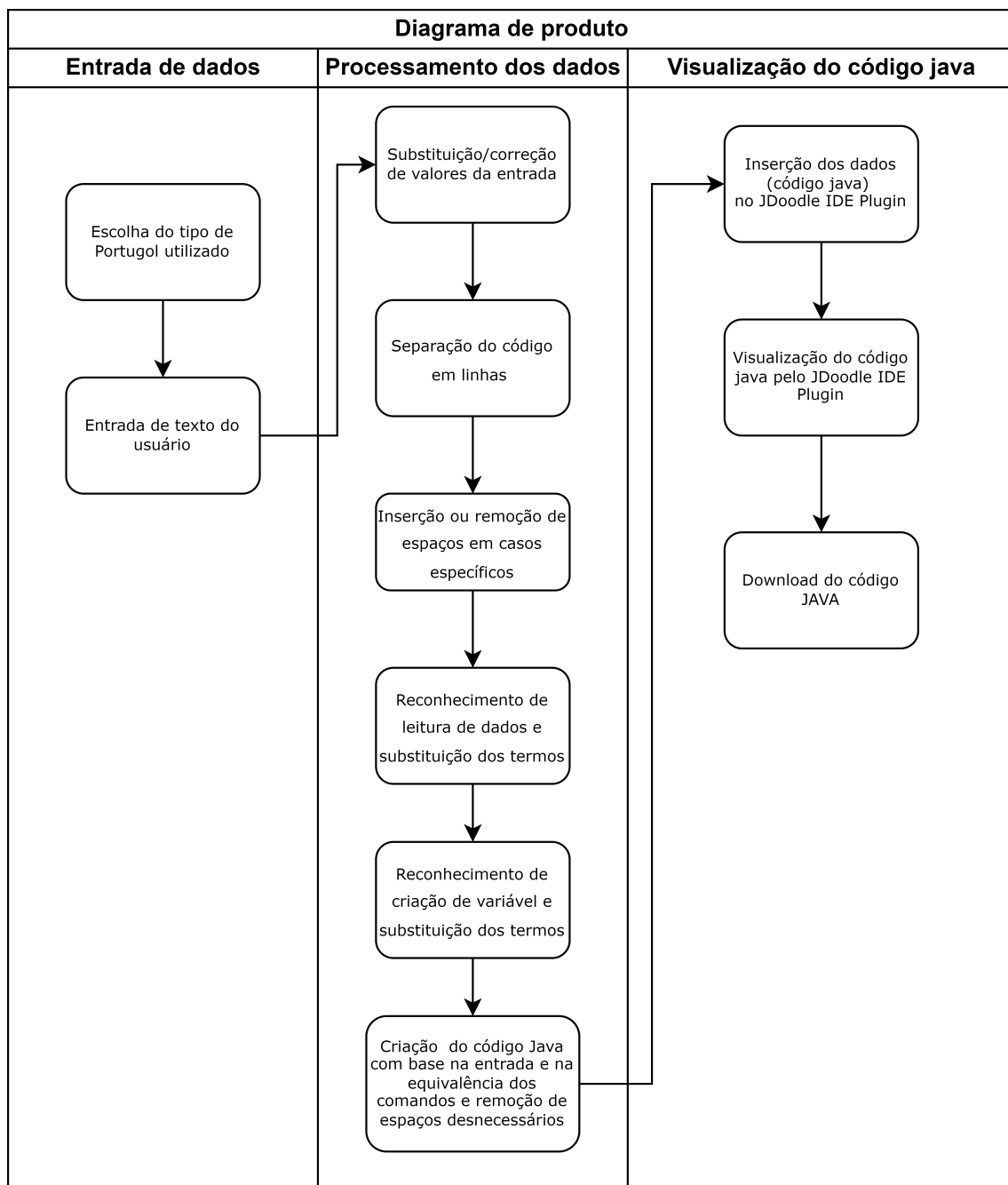


Figura 1. Diagrama do produto. Fonte: O próprio autor.

Um diagrama de fluxo visualiza as etapas envolvidas na criação e desenvolvimento de um software. A Figura 1 apresenta o diagrama do produto desenvolvido. A solução proposta automatiza essa visualização, dividindo o processo em três fases: entrada de dados (código em Portugol), processamento (análise sintática, substituição de palavras-chave, formatação) e saída (código em Java). As onze etapas envolvidas garantem a precisão e a eficiência da conversão, facilitando a transição entre as duas linguagens de programação.

6. Desenvolvimento

Esta seção descreve os passos do diagrama de produto apresentado na seção 5.2, cujas etapas e interações podem ser visualizadas na Figura 1.

6.1. Entrada de dados

Nesta etapa, o usuário poderá escolher o tipo de sintaxe do Portugol desejado e escrever o código correspondente. A diferenciação entre tipos de Portugol é necessária, pois mesmo sendo essencialmente a mesma linguagem, os dois tipos dispostos na ferramenta diferem em alguns comandos como: forma de criação e atribuição de variáveis e o uso de chaves para isolar determinados comandos. A diferença entre as duas sintaxes pode ser vista na Figura 2. Para a escrita do código, o usuário utilizará um campo de texto do tipo *textarea*. Depois de escrever o código, o usuário deve apertar o botão “Transformar” para iniciar o processo de conversão do código em Portugol para Java.

<pre> 1 algoritmo "Exemplo visualg" 2 // Exemplo visualg 3 var 4 n1, n2, o: inteiro 5 r: real 6 7 inicio 8 escreva(" Escreva um numero ") 9 leia(n1) 10 11 escreva(" Escreva outro numero ") 12 leia(n2) 13 14 escreval(" Digite uma opção ") 15 escreval(" opção 1: Adção ") 16 escreval(" opção 2: Subtração ") 17 leia(o) 18 19 escolha o 20 caso 1 21 r<-n1+n2 22 23 caso 2 24 r<-n1-n2 25 26 outrocaso 27 escreva("Nao temos essa opção! ") 28 r<-0 29 30 fimescolha 31 32 escreva("O resultado é ",r) 33 34 finalgoritmo 35 36 </pre>	<pre> 1 programa{ 2 // Exemplo Portugol Studio 3 funcao inicio(){ 4 inteiro n1, n2, o 5 real r 6 7 escreva("Escreva um numero") 8 leia(n1) 9 10 escreva("Escreva um numero") 11 leia(n2) 12 13 escreva(" Digite uma opção ") 14 escreva(" opção 1: Adção ") 15 escreva(" opção 2: Subtração ") 16 leia(o) 17 18 escolha (o){ 19 caso 1: 20 n1 = n1 + n2 21 pare 22 23 caso 2: 24 n2 = n2 - n1 25 pare 26 caso 3: 27 brancos = brancos + 1 28 pare 29 30 caso contrario: 31 escreva("Nao temos essa opção! ") 32 r = 0 33 } 34 escreva("O resultado é ",r) 35 36 } </pre>
--	---

Figura 2. Diferença entre sintaxe do VisuAlg (esquerda) e do Portugol Studio (direita). Fonte: O próprio autor.

6.2. Processamento de dados

Nesta fase, são realizadas as preparações e ajustes necessários para converter o código de Portugol para Java. Inicialmente, o código passa por um processo de preparação, onde

algumas palavras são substituídas para facilitar a transformação ou para alinhar termos ao seu equivalente em Java. Em seguida, o bloco de texto do usuário é dividido por linhas, e cada linha é processada para adicionar espaços em determinados caracteres, o que ajuda no funcionamento correto do código. Após essa etapa, as linhas são separadas com base nos espaços presentes.

Um dos processos realizados nesta seção é a detecção das variáveis e seus tipos. Essa fase é crucial, pois a leitura da entrada de dados em Java depende do conhecimento do tipo de variável que está sendo lida. Durante esse processamento, os nomes das variáveis são armazenados em uma estrutura chamada *Object*, onde as propriedades do objeto correspondem aos nomes das variáveis e os valores dessas chaves representam os tipos de cada variável. Além disso, são feitas as substituições necessárias para declarar um *array* ou matriz.

Em seguida, é definido o que acontece quando uma entrada de dados, identificada pela palavra reservada “leia” em Portugol, é detectada. Nesse momento, são realizadas as operações necessárias para recuperar o tipo da variável, que já foi armazenado pelo programa, e inserir o método correspondente da classe *Scanner* em Java.

Posteriormente o sistema identifica e processa estruturas de controle como o laço *for* e a condicional *switch/case*. Nessa etapa, são realizadas substituições e ajustes para garantir a correta conversão para o código Java. Por exemplo, a palavra-chave ‘passo’, usada em Portugol para incrementar o contador em um laço, é substituída pelo equivalente em Java.

A última etapa dessa fase envolve a geração do código em Java. Neste momento, os comandos escritos em Portugol são convertidos para seus equivalentes em Java, utilizando um objeto que armazena as propriedades correspondentes aos comandos em Portugol e os valores que representam os comandos equivalentes na linguagem da Oracle. Se uma palavra não estiver presente nesse objeto, ela é transcrita diretamente no código resultante. Ao identificar o final de uma instrução, um ponto e vírgula é inserido, marcando o término da linha executável. Em seguida, uma nova linha é iniciada. Contudo, para estruturas como laços ou condicionais, a delimitação é feita por uma quebra de linha seguida de uma tabulação, melhorando a legibilidade do código gerado.

6.3. Visualização do código JAVA

Após a etapa de processamento de dados, o código Java gerado é integrado ao JDoodle IDE Plugin. Esta integração possibilita a visualização imediata do código em uma interface intuitiva, conforme ilustrado na Figura 3. A partir dessa interface, o usuário dispõe de diversas funcionalidades, como a execução direta do código, o download do mesmo para uso posterior e o acesso ao site oficial do plugin para explorar opções de personalização e outras ferramentas.

6.4. Formulário On-line

Foi desenvolvido um formulário online utilizando a ferramenta *Google Forms*, com o objetivo de validar as expectativas dos usuários em relação a uma ferramenta de conversão. O formulário busca entender o que os usuários esperam da ferramenta e como foram suas experiências anteriores com outras soluções similares. Além disso, visa avaliar possíveis melhorias e novas funcionalidades que poderiam ser implementadas na ferramenta.

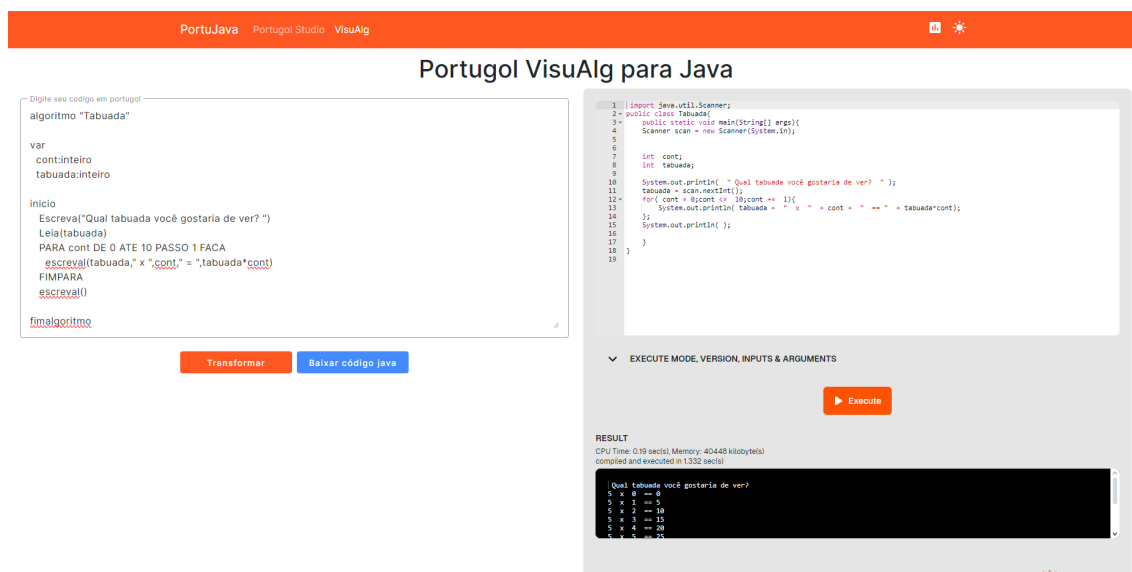


Figura 3. Visualização do código java utilizando o site Fonte: O próprio autor.

O formulário consiste em 18 perguntas organizadas em 5 seções. A primeira seção coleta informações sobre o participante e sua experiência com programação e linguagens. A segunda seção tem como objetivo entender as motivações e expectativas dos participantes em relação à ferramenta desenvolvida. As seções 3 e 4 exploram as experiências dos usuários com outras ferramentas de conversão, analisando como foram essas interações. Por fim, a última seção busca receber sugestões de melhorias e novas funcionalidades para a ferramenta.

O formulário foi divulgado em grupos de WhatsApp da instituição de ensino e por e-mail enviado pela Secretaria Acadêmica aos alunos, e também para pessoas e grupos relacionados à informática, utilizando o WhatsApp. O formulário permaneceu disponível para respostas de 21 de junho de 2024 até 29 de julho de 2024, recebendo um total de 26 respostas. A principal limitação desta pesquisa foi a dificuldade em obter um número maior de respostas para as questões propostas, o que pode ter impactado a profundidade da análise.

7. Análise dos Resultados

Esta seção descreve os resultados obtidos a partir do formulário mencionado na seção 6.4 e também introduz a ferramenta de conversão desenvolvida.

7.1. Formulário

A maioria das pessoas que participaram desta pesquisa possui entre 16 e 24 anos, representando 84,6% do total, e estão cursando o ensino superior. Grande parte dos participantes tem entre 0 e 4 anos de experiência em programação, e muitos não utilizaram a linguagem Portugol em seus estudos e projetos. No entanto, a grande maioria já utilizou a linguagem Java, totalizando 92,3% (ver Figura 4).

A maioria dos respondentes aprendeu Java com o objetivo de estudar, aprimorar habilidades e desenvolver aplicações. Entre os participantes, 53,8% desejam compreender melhor a sintaxe do Java ao utilizar a ferramenta, enquanto 46,2% buscam aprender novas

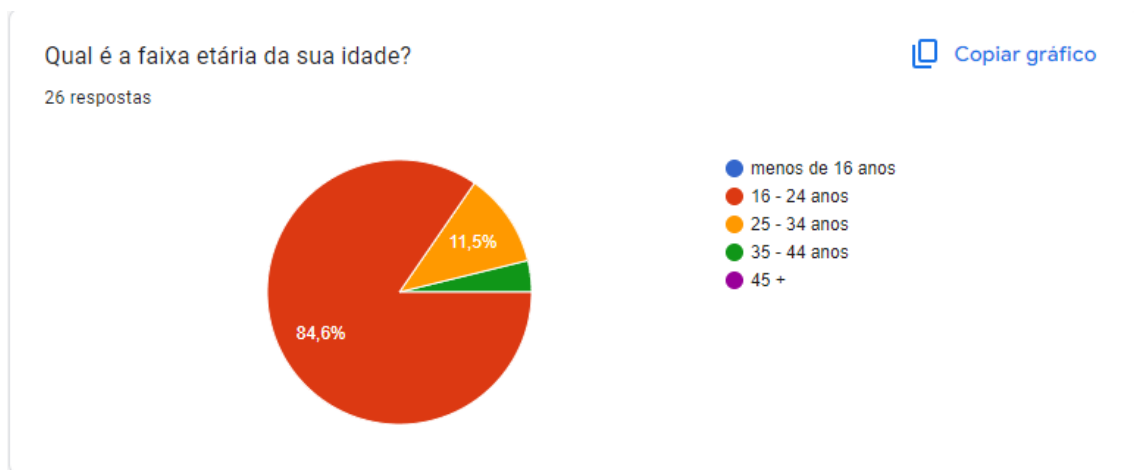


Figura 4. Percentual da faixa etária dos respondentes. Fonte: O próprio autor.

funcionalidades da linguagem criada pela Sun Microsystems. Além disso, 34,6% dos respondentes pretendem facilitar a migração de seus projetos desenvolvidos em Portugol (conforme ilustrado na Figura 5).

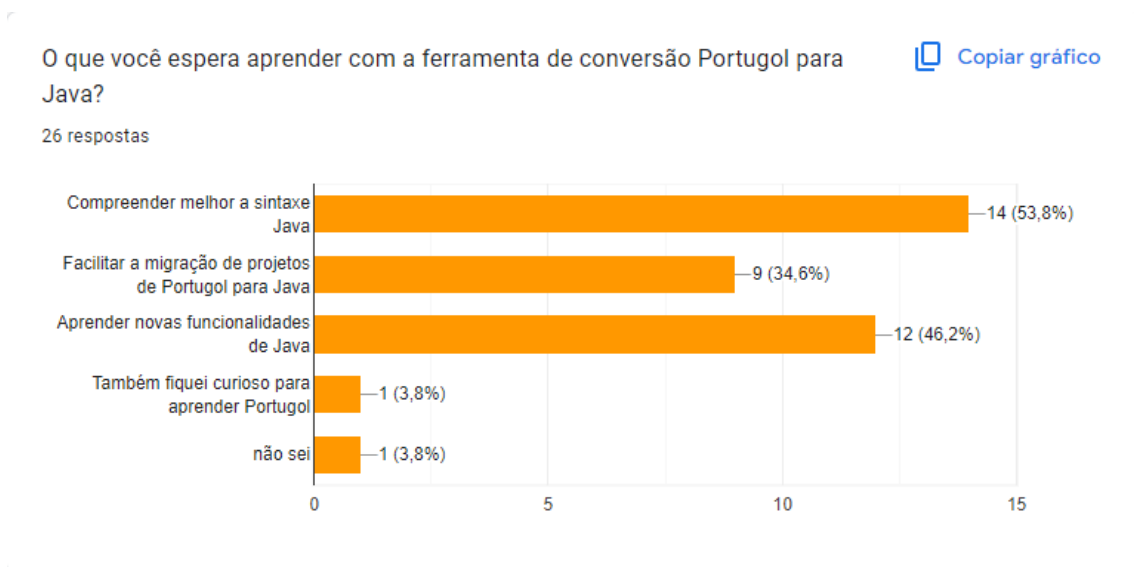


Figura 5. Resultado a respeito da pergunta sobre expectativa de aprendizado. Fonte: O próprio autor.

Grande parte dos participantes expressou o desejo de encontrar uma conversão automática de código, além de explicações sobre as diferenças entre as duas linguagens. Eles também gostariam de exemplos e tutoriais de códigos, mas acreditam que desafios como a sintaxe diferente, a manipulação de tipos de dados e principalmente a diferença entre os paradigmas das linguagens — estruturada versus orientada a objetos — podem ser os principais obstáculos ao realizar a conversão (ver Figura 6).

Embora muitos dos participantes nunca tenham utilizado uma ferramenta de conversão de linguagens de programação, 65,4% acreditam que a ferramenta de conversão de Portugol para Java seja útil. Os participantes que já utilizaram alguma ferramenta utiliza-

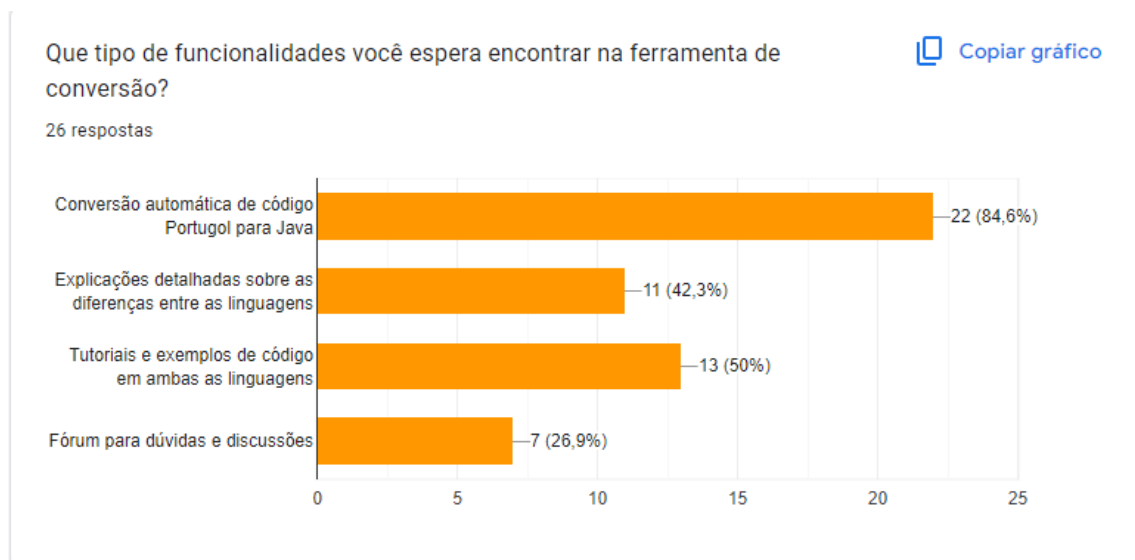


Figura 6. Funcionalidades esperadas. Fonte: O próprio autor.

ram ferramentas de IA, como o chat GPT, e apontaram que tiveram uma boa experiência e que foram ferramentas úteis (ver figura 7 e 8).

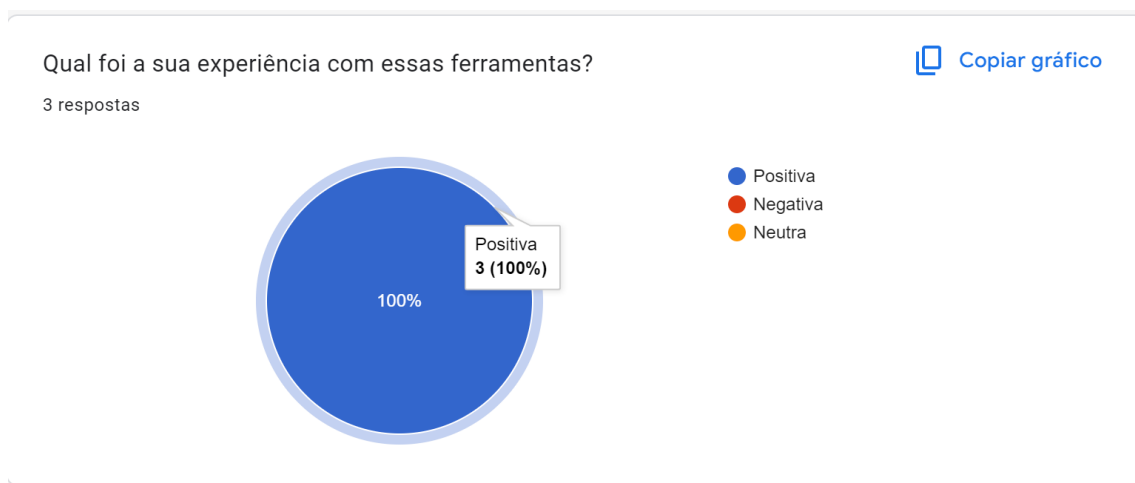


Figura 7. Funcionalidades esperadas. Fonte: O próprio autor.

No entanto, os participantes esperam alguns recursos adicionais nessa ferramenta, como conexão com o Google Drive para salvamento automático na nuvem, um sistema que identifique erros no código e que mostre qual expressão é equivalente em Java, destacando as diferenças de sintaxe entre ambas. Além disso, sugeriram o uso de inteligências artificiais para demonstrar outras formas de implementar o mesmo código e para modificar partes do código durante a conversão.

7.2. Programa de Conversão

7.2.1. Deploy e publicação

Após o desenvolvimento do código, foi necessário encontrar uma forma de publicar o site. Para isso, utilizou-se uma ferramenta disponibilizada pelo próprio GitHub: o GitHub

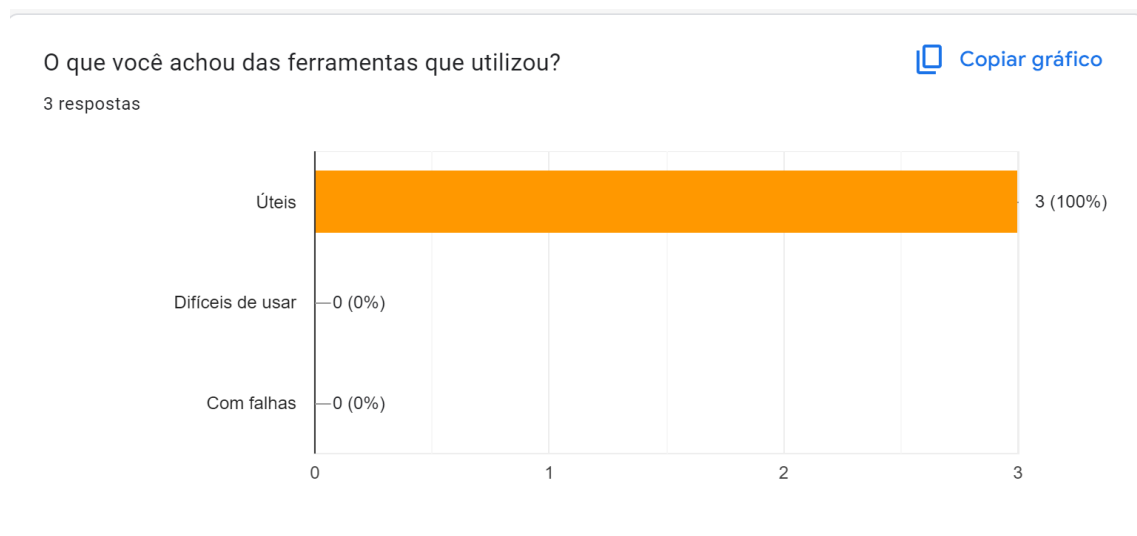


Figura 8. Funcionalidades esperadas. Fonte: O próprio autor.

Pages. Essa ferramenta foi escolhida por ser gratuita e de fácil utilização, permitindo uma publicação simples e eficiente.

Com isso, o site que contém a ferramenta de conversão foi publicado e está disponível no link <https://matpires02.github.io/portujava/>. Na tela inicial, os usuários podem, por meio da barra de navegação, selecionar o tipo de Portugol a ser escrito e também alterar o esquema de cores da aplicação, como ilustrado na Figura 9.

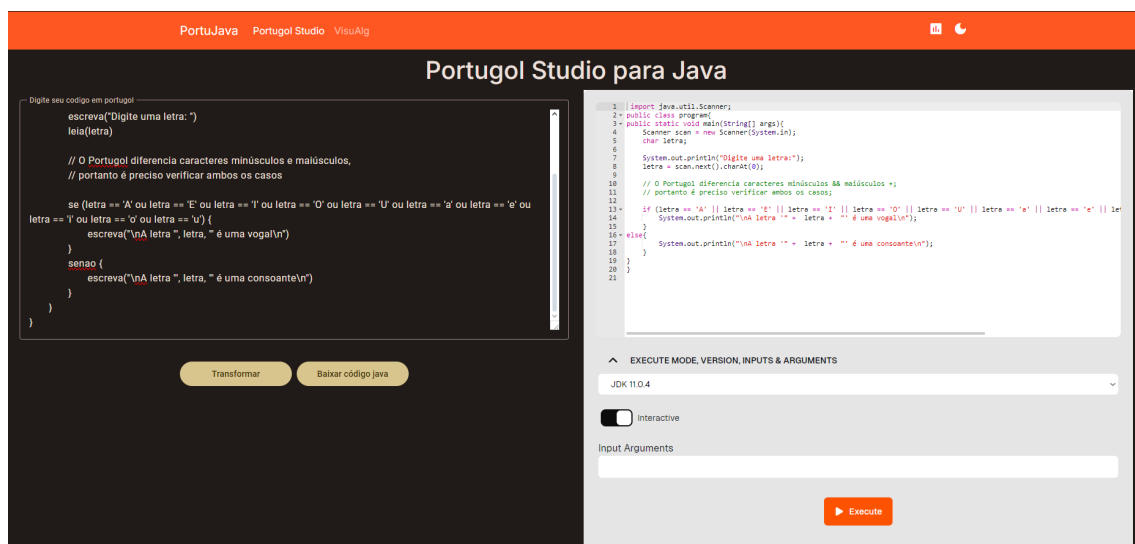


Figura 9. Visualização da ferramenta PortuJava após conversão. Fonte: O próprio autor.

7.2.2. Estruturas suportadas

A ferramenta de conversão foi desenvolvida para suportar as principais estruturas básicas, incluindo estruturas condicionais como *if* e *switch case*, além de laços de repetição como

do *while* e *for*. Também oferece suporte a vetores. Esta seção exemplifica as estruturas *if* e *for*, assim como o uso de vetores, para cada um dos tipos de conversão.

Para ilustrar a conversão de estruturas condicionais, foi utilizado um programa cujo objetivo é verificar se o caractere digitado é uma vogal ou uma consoante. A Figura 10 apresenta a conversão dessa estrutura no Portugol Studio, enquanto a Figura 11 mostra a mesma conversão no VisualG.

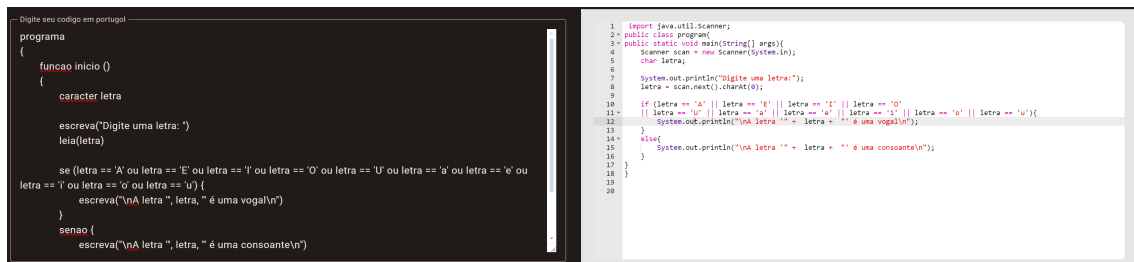


Figura 10. Conversão com estrutura condicional - Portugol Studio. Fonte: O próprio autor.

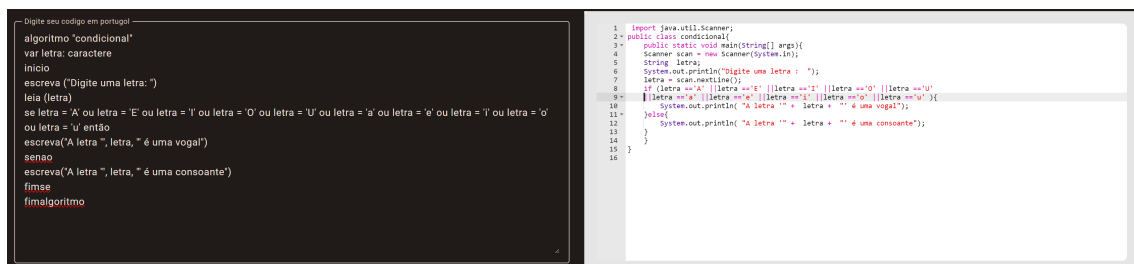


Figura 11. Conversão com estrutura condicional - VisualG. Fonte: O próprio autor.

Para exemplificar a conversão de laços de repetição, foi utilizado um programa que calcula a tabuada de um número digitado pelo usuário. A Figura 12 ilustra a conversão desse laço no Portugol Studio, enquanto a Figura 13 apresenta a mesma conversão no VisualG.

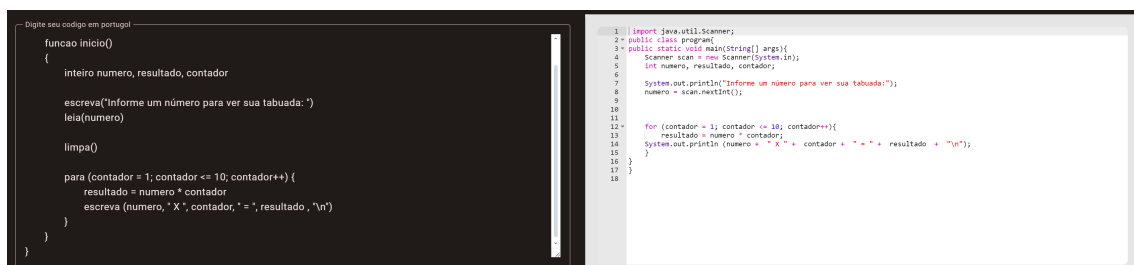


Figura 12. Conversão com laço de repetição - Portugol Studio. Fonte: O próprio autor.

Exemplificando o uso de vetores, foi desenvolvido um programa que popula e exibe os valores de um vetor, tanto na ordem de inserção quanto do último para o primeiro. É importante ressaltar que, para este caso específico, foi necessário alterar o código gerado, pois não há suporte à importação, e esse exemplo requer a biblioteca Random para

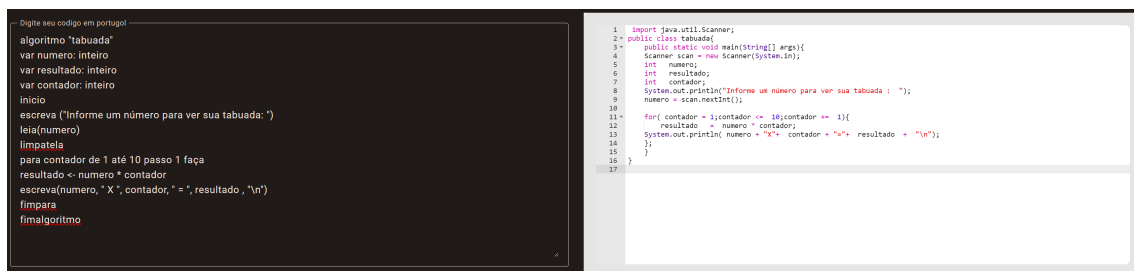


Figura 13. Conversão com laço de repetição - VisualG. Fonte: O próprio autor.

a geração de números aleatórios. A Figura 14 mostra a conversão utilizando vetores no Portugol Studio, enquanto a Figura 15 apresenta a mesma conversão no VisualG.

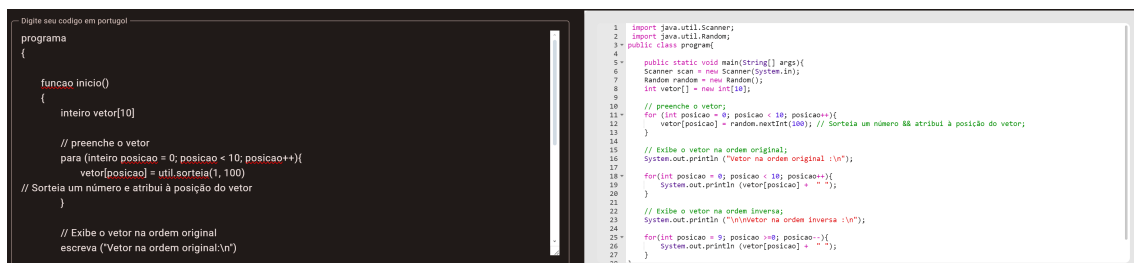


Figura 14. Conversão com utilização de vetores - Portugol Studio. Fonte: O próprio autor.

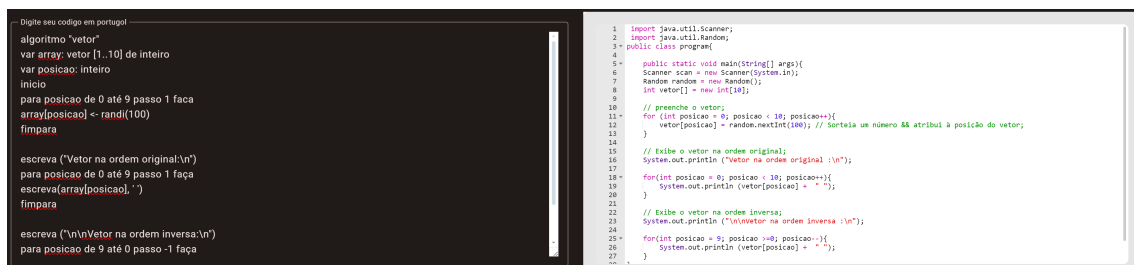


Figura 15. Conversão com utilização de vetores - VisualG. Fonte: O próprio autor.

8. Conclusão

O desenvolvimento da ferramenta PortuJava visa suprir a necessidade de ferramentas educacionais que facilitem a transição de estudantes de linguagens introdutórias, como o Portugol, para linguagens amplamente aplicadas no mercado, como o Java. Através de uma interface acessível, a ferramenta possibilita a conversão de códigos entre essas linguagens, abordando dificuldades comuns, como as diferenças de sintaxe e o entendimento de erros, fatores que costumam dificultar o aprendizado e aumentar as taxas de evasão em cursos de programação. Os educadores podem utilizar a ferramenta para exibir na prática as diferenças de sintaxe entre as linguagens. A ferramenta também pode ser utilizada para fazer a transcrição de códigos e projetos feitos anteriormente. Os resultados iniciais indicam que a ferramenta é bem recebida, sendo considerada pelos estudantes uma solução prática para o desenvolvimento de competências em programação em um contexto profissional.

Contudo, o trabalho apresenta limitações, dentre as quais destaca-se o pequeno número de respondentes ao formulário de avaliação, o que restringe a representatividade dos resultados obtidos. Além disso, a ferramenta ainda carece de funcionalidades mais avançadas para atender a uma gama mais ampla de necessidades de aprendizado, como a correção de erros automatizada e a expansão do suporte a outras linguagens.

Como trabalhos futuros, planeja-se aprimorar o PortuJava com novas funcionalidades, incluindo a exibição detalhada de erros de código e sugestões de correção automática, promovendo uma aprendizagem mais autônoma. Também é proposta a integração de um sistema de inteligência artificial, capaz de sugerir alternativas de codificação e refatoração, bem como a expansão para o suporte a outras linguagens de programação. Essas melhorias visam transformar o PortuJava em uma plataforma robusta e adaptável, capaz de acompanhar e auxiliar o desenvolvimento de habilidades de programação em diferentes níveis de proficiência.

Referências

- [Brasscom 2021] Brasscom (2021). Demanda de talentos em tic e estratégia σ tcem. <https://brasscom.org.br/pdfs/demanda-de-talentos-em-tic-e-estrategia-tcem/>.
- [Carmona 2014] Carmona, H. (2014). Aula 10 - equivalência java x portugol studio - parte 1. <https://www.slideshare.net/henriquecarmona/aula-10-equivalencia-java-x-portugol-studio-parte-1>.
- [Cass 2022] Cass, S. (2022). Top programming languages 2022. <https://spectrum.ieee.org/top-programming-languages-2022>.
- [Chiara 2014] Chiara, R. (2014). O senhor dos algoritmos. <https://www.ramon.pro.br/ebooks/O-Senhor-dos-Algoritmos-v1.0.pdf>.
- [Cristovão 2008] Cristovão, H. M. (2008). Aprendizagem de algoritmos num contexto significativo e motivador: um relato de experiência. *SBC*, 30.
- [Fernandes and Junior 2016] Fernandes, V. d. S. and Junior, V. F. (2016). Evasão e reprovação nas disciplinas de lógica e programação: Informações preliminares no campus sombrio, do instituto federal catarinense. In *5º SICT-SUL-Simpósio de Integração Científica e Tecnológica do Sul Catarinense*.

- [Gomes et al. 2008] Gomes, A., Henriques, J., and Mendes, A. J. (2008). Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. *Educação, Formação & Tecnologias*, 1(1):93–103.
- [Google] Google. What is angular? <https://angular.io/guide/what-is-angular>.
- [Indrusiak 1996] Indrusiak, L. S. (1996). Linguagem java. *Grupo JavaRS JUG Rio Grande do Sul*, page 19.
- [Junior and França 2017] Junior, S. d. S. and França, S. (2017). Programação para todos: Análise comparativa de ferramentas utilizadas no ensino de programação. In *Anais do XXV Workshop sobre Educação em Computação*, Porto Alegre, RS, Brasil. SBC.
- [Medeiros 2015] Medeiros, A. V. M. (2015). Um interpretador online para a linguagem portugal. *Trabalho de Conclusão de Curso (graduação)–Universidade Federal de Sergipe, Curso de Ciência da Computação*, pages 23–26.
- [Medeiros 2019] Medeiros, R. P. (2019). *Hello, world: uma análise sobre dificuldades no ensino e na aprendizagem de introdução à programação nas universidades*. PhD thesis, Universidade Federal de Pernambuco.
- [Noschang et al. 2014] Noschang, L. F., Pelz, F., de Jesus, E., and Raabe, A. (2014). Portugal studio: Uma ide para iniciantes em programação. In *Anais do XXII Workshop sobre Educação em Computação*, pages 1–10. SBC.
- [Oliveira et al. 2014] Oliveira, M., De Souza, A., Ferreira, A., and Barreiros, E. (2014). Ensino de lógica de programação no ensino fundamental utilizando o scratch: um relato de experiência. In *Anais do XXII Workshop sobre Educação em Computação*, pages 239–248. SBC.
- [Wanderley et al. 2020] Wanderley, E. G., Wanderley, R. G., Melo, M. G. R., Dias, C. L. D. T., and Sales, Y. A. d. S. (2020). Jogo de tabuleiro “programming”—uma nova estratégia pedagógica para o ensino e a aprendizagem de algoritmo. *Brazilian Journal of Development*, 6(5):32006–32015.
- [Zanini and Raabe 2012] Zanini, A. S. and Raabe, A. L. A. (2012). Análise dos enunciados utilizados nos problemas de programação introdutória em cursos de ciência da computação no brasil. In *Anais do XXXII Congresso da Sociedade Brasileira de Computação, XX WEI–Workshop sobre Educação em Computação*.