

# Codeverse: Um RPG Digital para Iniciantes em Programação

Rayssa M. S. Santana<sup>1</sup>, Claudia B. Rizzi<sup>1</sup>, Clodis Boscarioli<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação (PPGComp) -  
Universidade Estadual do Oeste do Paraná (Unioeste)  
Rua Universitária, 1619 - Faculdade, Cascavel - PR - Brasil

{rayssa.santana, claudia.rizzi, clodis.boscarioli}@unioeste.br

**Abstract.** This work introduces the game Codeverse: Hacking Worlds, a digital game in educational RPG format designed to support the teaching and learning of introductory programming concepts. The game is aimed at students beginning programming and provides a progressive learning experience based on Bloom's revised taxonomy. Each game world represents a main content, such as variables, data types, input and output, conditional and repetition structures, with challenges written in pseudo-code. Codeverse aims not only to promote learning, but also to increase the engagement of students who often find this content abstract and challenging. At the end of the journey, the game provides a personalized learning recommendation based on the student's performance and generates a report so that the teacher can track learning progress.

**Resumo.** Este trabalho apresenta o jogo Codeverse: Hackeando Mundos, um jogo digital no formato de RPG educacional que visa apoiar o ensino e aprendizagem de conceitos introdutórios de programação. Voltado para estudantes iniciantes em programação, o jogo propõe uma experiência progressiva de aprendizagem baseada na Taxonomia de Bloom Revisada. Cada mundo do jogo representa um conteúdo principal, como variáveis, tipos de dados, entrada e saída, estruturas condicionais e de repetição, com desafios redigidos em pseudocódigo. O Codeverse busca não apenas desenvolver a aprendizagem, mas também aumentar o engajamento dos estudantes, que frequentemente percebem esses conteudos como abstratos e desafiadores. Ao final da jornada, o jogo oferece uma recomendação personalizada de estudo com base no desempenho do estudante jogador e gera um relatório para que o professor possa acompanhar a trajetória de aprendizagem.

# Codeverse: Um RPG Digital para Iniciantes em Programação

Rayssa M. S. Santana<sup>1</sup>, Claudia B. Rizzi<sup>1</sup>, Clodis Boscarioli<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação (PPGComp) - Universidade Estadual do Oeste do Paraná (Unioeste)

{rayssa.santana, claudia.rizzi, clodis.boscarioli}@unioeste.br

## Contexto

O ensino de programação, especialmente nas disciplinas introdutórias, tem sido marcado por altos índices de evasão e dificuldades de aprendizagem, muitas vezes relacionadas à abstração exigida e à falta de vínculo entre teoria e prática (SILVA et al., 2018, SOUSA et al., 2022). Estudantes iniciantes frequentemente enfrentam barreiras cognitivas e emocionais, o que compromete seu desempenho e permanência nas disciplinas (MORAIS, 2022). Nesse contexto, o jogo Codeverse: Hackeando Mundos foi desenvolvido para atuar oferecendo uma abordagem lúdica e imersiva para a introdução de conteúdos iniciais de programação. O jogo se ancora na Taxonomia de Bloom Revisada, utilizada aqui como estrutura cognitiva para progressão de desafios. Além disso, está alinhado à BNCC (2018) e à BNCC Computação (2022), especificamente sobre habilidades do eixo Pensamento Computacional.

## Aspectos Técnicos

O jogo foi desenvolvido na engine GDevelop 5, com lógica baseada em eventos. Os desafios são validados por meio de comparação de strings em pseudocódigo, e seguem a sintaxe do Visualg, promovendo familiaridade com o ambiente educacional. O progresso é controlado por variáveis e senhas, e ao final, um arquivo JSON é gerado com os dados de desempenho do jogador para uso docente.

## Especificações

### Público-alvo:

Iniciantes em programação

### Gênero:

RPG

### Plataforma de desenvolvimento:

GDevelop 5

### Conteúdo Principais:

Variáveis, tipos de dados, entrada e saída de dados, estruturas condicionais e estruturas de repetições.

### Disciplina:

Introdução à Programação.

## Narrativa

Em Codeverse, você assume o papel de Codebreaker, um jovem programador escolhido por Zeta, uma mentora virtual, para enfrentar Nullex, um vírus com consciência própria que ameaça destruir o equilíbrio digital de Cibercity. Para salvar a cidade, será preciso explorar cinco mundos – diferentes locais da cidade – resolvendo missões desafiadoras por meio de algoritmos em Portugol. A cada conquista, você ganha experiência e se aproxima do confronto final contra Nullex, onde lógica e estratégia serão suas maiores armas.

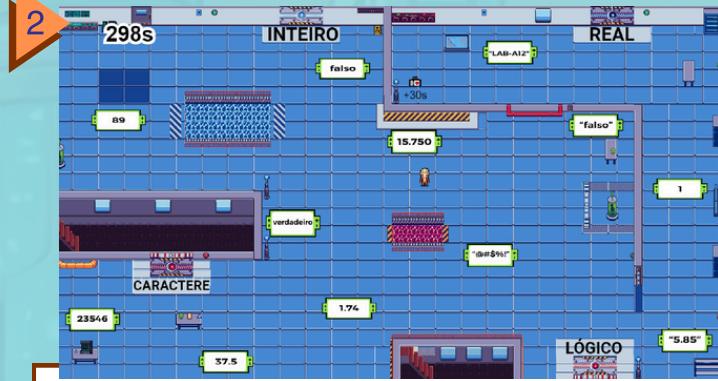


O jogo é dividido em cinco mundos, cada um representando uma área de Cibercity e um conceito essencial da programação. Em cada fase, o jogador deve resolver desafios usando lógica em Portugol, como identificar variáveis, tomar decisões com estruturas condicionais e aplicar laços de repetição.



## Mundo 1 - Cibercity (Variáveis)

→ O jogador explora Cibercity, inicialmente verificando se os nomes dos vírus funcionam como variáveis válidas ou inválidas. Na segunda parte, responde desafios com conceito de variáveis, declaração e atribuição de valores.



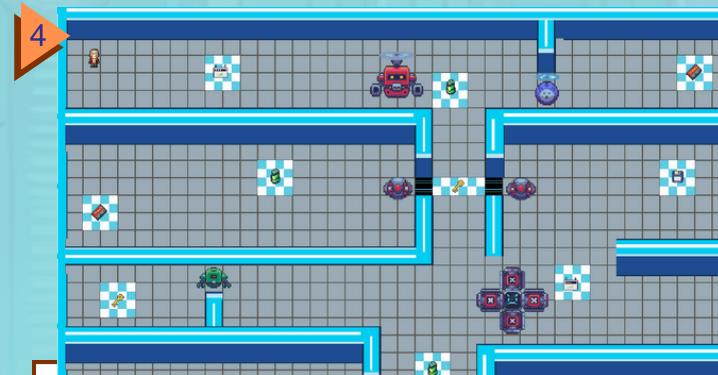
## Mundo 2 - Laboratório de Dados (Tipos de Dados)

→ No laboratório de dados o jogador recolhe as baterias espalhadas no local e deve depositar no seu tipo certo, de acordo com o seu valor. Podendo ser caractere, lógico, real ou inteiro.



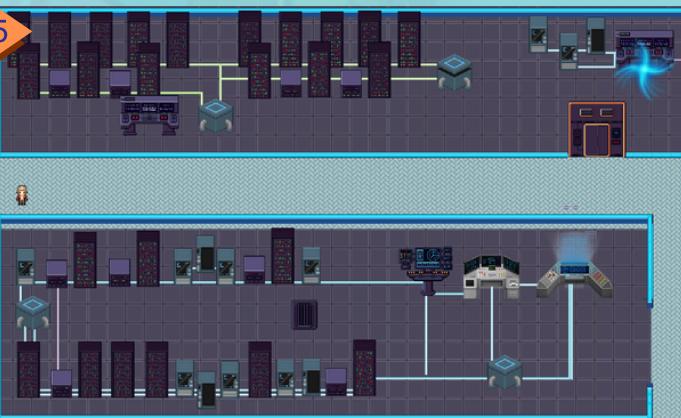
## Mundo 3 - Fábrica de Energia (Entrada e Saída de dados)

→ O jogador deve seguir os sinais luminosos para resolver os desafios. Ao se aproximar do sinal, o desafio é liberado. Nesse mundo também são trabalhados conteúdos secundários de operadores.



## Mundo 4 - Labirinto (Condicionais)

→ Neste mundo, o jogador deve coletar itens espalhados pelo cenário e, para avançar, precisa enfrentar os robôs de Nullex. Cada robô lança um desafio relacionado a estruturas condicionais, e só ao responder corretamente o jogador consegue derrotá-lo e seguir pelo labirinto.



## Mundo 5 - Data Center (Laços de Repetição)

→ Neste mundo, os jogadores devem seguir sinais luminosos que indicam pontos de desafio. Ao se aproximar de cada sinal, um desafio sobre laços de repetição é lançado. No final, ocorre o confronto decisivo com Nullex, composto por uma sequência de perguntas relâmpago e um desafio final que testa todo o conhecimento adquirido.

# Conteúdos Pedagógicos

O Codeverse trabalha conteúdos introdutórios de programação por meio de desafios práticos e progressivos, estruturados a partir da Taxonomia de Bloom Revisada, considerando a natureza das tarefas e o tipo de operação mental exigida, como orienta Krathwohl (2002). Além disso, foram organizados em conteúdos principais, os quais aparecem de forma direta nos desafios e nas mecânicas centrais do jogo e conteúdos secundários, que estão presentes de forma complementar, não sendo o foco central de aprendizagem. Os desafios foram apresentados em pseudocódigo, uma abordagem que favorece a aprendizagem gradual, reduz a sobrecarga cognitiva e fortalece o desenvolvimento do pensamento computacional, ao permitir que os estudantes concentrem-se no raciocínio lógico e na resolução de problemas (HOFF DO AMARAL, 2015; BORBA; MARCHI; REHFELDT, 2021).

## Conceitos principais:

- Variáveis;
- Tipos de dados (inteiro, real, caractere e lógico);
- Entrada e saída de dados;
- Estruturas condicionais (simples, compostas e aninhadas);
- Estruturas de repetição (enquanto, repita até, para).

## Conceitos secundários:

- Operadores aritméticos;
- Operadores relacionais;
- Operadores lógicos;
- Noções de erro e depuração básica (implícitos nos desafios do jogo).

## Processo de Desenvolvimento do Codeverse



## Inovação

O Codeverse é uma abstração lúdica de conteúdos introdutórios de programação, organizados sequencialmente com base na Taxonomia de Bloom Revisada. Cada desafio foi criado associado a um nível cognitivo, promovendo uma progressão pedagógica intencional. O jogo apresenta missões em pseudocódigo dentro de uma narrativa em estilo RPG, favorecendo a aprendizagem. Conta ainda com um sistema de coleta de dados, permitindo que professores acompanhem o percurso de aprendizagem dos estudantes ao final do uso, além de oferecer uma recomendação de estudo personalizada aos alunos.

## Referências

- BORBA, Fabrício Hartmann; MARCHI, Miriam Ines; REHFELDT, Márcia Jussara Hepp. Utilização do Software VisuAlg no Ensino da Lógica de Programação. *Revista de Ensino, Educação e Ciências Humanas*, Londrina, v. 22, n. 3, p. 295–304, 2021. Disponível em: <https://encurtador.com.br/baYwr>. Acesso em: 24 maio 2025.
- BRASIL. Computação na Base Nacional Comum Curricular: Documento Referência. 2022. Disponível em: <https://encurtador.com.br/3z2Au>. Acesso em: 25 mai. 2025.
- BRASIL, Base Nacional Comum Curricular: Ensino Médio. Brasília, DF: MEC 2018. Disponível em: <http://basenacionalcomum.mec.gov.br/>. Acesso em: 25 de maio de 2025.
- HOFF DO AMARAL, Érico Marcelo. Processo de Ensino e Aprendizagem de Algoritmos Integrando Ambientes Imersivos e o Paradigma de Blocos de Programação Visual. 2015. 255f. Tese (Doutorado em Informática na Educação) - Programa de Pós-Graduação em Informática na Educação, Universidade Federal do Rio Grande do Sul - UFRS, Porto Alegre, 2015. Disponível em: <https://encurtador.com.br/rDdY5>. Acesso em: 25 maio 2025.
- KRATHWOHL, D. R. A revision of Bloom's taxonomy: an overview. *Theory into Practice*, v. 41, n. 4, p. 212–218, 2002.
- MORAIS, C. G. B. Ensino e aprendizagem de programação: estudo de caso no Ensino Superior. 2022. 269f. Tese (Doutorado em Ciências da Educação) – Universidade do Minho, Braga, Portugal. Disponível em: <https://encurtador.com.br/YBqY>. Acesso em: 25 mai. 2025.
- SILVA, W. dos Santos et al. Levantamento sobre as dificuldades dos discentes nas disciplinas de programação no curso técnico de informática. *Diversitas Journal*, v. 3, n. 3, p. 761–770, 2018.
- SOUSA, M. K. V. et al. Análise dos fatores de evasão dos alunos dos cursos técnicos da Área de informática no IFPB campus Campina Grande. In: *Anais do Workshop sobre Educação em Computação (WEI)*. Porto Alegre: Sociedade Brasileira de Computação. 2022. p. 369–380.