

# Cyberspace CodeX: Um Jogo Sobre Tecnologia da Informação e Lógica de Programação

Arison Heltami Rodrigues Uchôa\* Thyago Bezerra Lima† Yarick Ivens da Silva Santos‡  
 Emílio Thalles Rocha§ André Lucas da Silva Rodrigues¶ Luan Souza Frota||  
 Glaudiney Moreira Mendonça Junior\*\*

Universidade Federal do Ceará, Sistemas e Mídias Digitais, Brasil



Figura 1: Imagens do jogo *Cyberspace CodeX*

## RESUMO

O objetivo desse artigo é analisar as metáforas criadas e desenvolvidas no jogo *Cyberspace CodeX* pensadas para, além de uma experiência lúdica significativa, embasar conteúdos relevantes ao ensino básico de programação, fomentando princípios seminais como os conceitos de variáveis, operações lógicas, operações condicionais e estruturas de repetição. A completude do projeto, eficiência e eficácia na elaboração das metáforas foi repassada ao longo de iterações, tanto no aspecto de design, para a criação de um produto final satisfatório, quanto com orientadores que estabeleceram as métricas relacionadas ao embasamento na área de Tecnologia da Informação e Cibercultura, as quais foram trabalhadas para unir, do melhor modo, o aspecto lúdico com o incentivo cognitivo e educacional. Essas mecânicas, suas concepções e a maneira pelas quais elas atendem aos requisitos propostos tornaram-se o objeto de estudo deste artigo.

**Palavras-chave:** Educação, Design de Jogos, Mecânicas de Jogo, Aprendizado de Programação.

## 1 INTRODUÇÃO

Na busca por estudar o Design de Jogos e a construção de uma experiência lúdica, um dos primeiros obstáculos se dá nas competências atribuídas ao *Game Designer* como figura criadora, pondo em xeque especificamente o que ele é compelido de criar. Segundo Zimmerman e Salen, a atribuição do Designer e a própria criação do design de um jogo podem ser descritos como “o processo pelo qual um designer cria um contexto a ser encontrado por um participante, a partir do qual o significado emerge” [7].

\*e-mail: arisonheltami@gmail.com

†e-mail: thyagobezerra8608@gmail.com

‡e-mail: yarickivens2@hotmail.com

§e-mail: emiliothalles@gmail.com

¶e-mail: andreh\_rukasu@hotmail.com

||e-mail: luanfrota@gmail.com

\*\*e-mail: glaudiney@virtual.ufc.br

Essa definição resumida trata exatamente sobre a atribuição: criar um contexto propício para a emergência de significados. Para tal o *Game Designer* utiliza de diversos artifícios ao escrever narrativas, projetar experiências e planejar mecânicas [8], transmitidas ao jogador de forma a permitir a formação desse significado e estabelecer uma relação de comunicação de suas ideias com o usuário final.

Para a criação e a manutenção desses contextos experienciais, o *Game Designer* deve permitir a percepção do usuário, assim como oferecer subsídios e conhecimentos para que possa manusear e interagir com os entes que foram projetados a fim de propiciar a experiência. Dá-se o nome de metáfora e analogias a esses objetos que agem de forma a referenciar elementos de conhecimento do usuário, assim como guardam semelhanças com esses referenciais, apesar das diferenças inerentes a forma de transmissão, material, construção ou uso [6].

Metáforas são ferramentas extremamente importantes na construção de um design de jogo, proporcionando uma interação mais intuitiva mediada pelos modelos mentais já formulados pelos usuários, facilitando o processamento de informações [2] e, por sua vez, apoiando a construção do estado de fluxo [3]. Dessa forma, metáforas podem auxiliar na compreensão e na resolução de problemas, potencializando a capacidade criativa, enriquecendo e facilitando o entendimento sobre os assuntos apresentados [4].

Observadas essas características nas metáforas e sua importância na construção de uma experiência lúdica, esse artigo busca analisar tais capacidades, relacioná-las com o contexto apresentado para a geração de uma interação lúdica significativa [7] e a produção de um estado de fluxo no jogo *Cyberspace CodeX*, especificamente em suas mecânicas construídas para apoiar a compreensão de conceitos de lógica de programação.

## 2 TRABALHOS RELACIONADOS

Associar jogos ao ensino de programação ou usar conceitos de ambientes virtuais para contextualizar a narrativa em jogos não é uma novidade na indústria. Diversos jogos abordam o conceito ou a temática para trabalhar cenários virtuais, lógica de programação e tecnologia da informação em seus escopos.

Há jogos relevantes que se propõem a essa atividade como o *Serious Game Code Combat* (CodeCombat Inc., 2013) que busca trabalhar o ensino de linguagens de programação diversas em um

cenário de fantasia. Há jogos que se propõem ao ensino de lógica estruturada aplicável a programação como o *Lightbot* (Armorgames, 2008) (Figura 2).



Figura 2: Telas de jogo do *Code Combat* (esquerda) e do *Lightbot* (direita)

Fora do escopo de *Serious Games*, existem diversos jogos que abordam conceitos de programação em seu escopo como *Hack'n'Slash* (Double Fine, 2014) que insere conceitos de atributos e lógica de programação em suas mecânicas, e *Glitchspace* (Space Budgie, 2016) que trabalha a lógica em diversos quebra-cabeças como meio de progressão no jogo (Figura 3).



Figura 3: Telas de jogo do *Hack'n'Slash* (esquerda) e do *Glitchspace* (direita)

### 3 CONCEPÇÃO DO PROJETO

*Cyberspace CodeX* (Figura 4) é um projeto desenvolvido durante dois semestres do curso de Sistemas e Mídias Digitais da Universidade Federal do Ceará. Criado por alunos do segundo (posteriormente terceiro) semestre, é um jogo de Estratégia em Turnos do gênero Tático, ambientado em um mundo virtual paralelo ao mundo real.



Figura 4: Logo do jogo *Cyberspace CodeX*

Com uma narrativa onde os personagens, antes de um mundo virtual, têm noção de que há uma pessoa em um mundo paralelo auxiliando-os (no caso, o jogador, em uma narrativa metalinguística). Eles devem enfrentar inimigos misteriosos que estão invadindo seu mundo, tendo o jogador como estrategista, desvendando uma trama de conspiração enquanto ajuda as entidades virtuais desse mundo.

Para auxiliar esse mundo virtual, o jogador comanda seus aliados em diversas missões que surgem ao redor do *Metaverse*, o mundo virtual do jogo, combatendo inimigos e gerenciando seus aliados através de uma interface que emula um sistema operacional. Diversos aliados vão se unindo ao grupo liderado pelo jogador durante a narrativa, enquanto possibilita novas estratégias como customizações de habilidades e missões cada vez mais complexas geradas proceduralmente.

O MVP (*Minimum Viable Product* - Produto Mínimo Viável) de *Cyberspace CodeX* foi desenvolvido inicialmente utilizando a tecnologia *Javascript* com o auxílio da biblioteca *CreateJS* para a criação da lógica do jogo. O design da interface, assim como dos personagens, ambientes e *sprites* foram desenvolvidos utilizando as ferramentas *Adobe Photoshop CS6*, *PaintTool Sai* e *Spriter Pro*. Posteriormente, a lógica foi transferida para a *Unity 5.3* com o objetivo de resolver problemas de compatibilidade e eficiência, assim como permitir um escalonamento adequado ao produto final.

O processo de desenvolvimento de *Cyberspace CodeX*, desde a sua concepção, foi acompanhado pelos professores orientadores como parte de trabalhos específicos da faculdade, em avaliações multidisciplinares e em disciplinas voltadas especialmente para o acompanhamento da criação do produto. Através desses acompanhamentos, diversos pontos relacionados ao design da interface, construção narrativa e aspectos cognitivos foram avaliados e passaram por processos de iteração para gerar o MVP do jogo.

### 4 METODOLOGIA DA ANÁLISE

Diversas mecânicas em *Cyberspace CodeX* foram fruto da necessidade de abordar, em seu escopo, aspectos relacionados a conceitos de lógica de programação, trabalhados no ensino de disciplinas voltadas a introdução à programação. Inseridas na forma de mecânicas secundárias [5], elas servem de apoio para a progressão do jogador na narrativa e na evolução de seus personagens ao longo do jogo.

Para basear a análise dessas mecânicas, foi utilizado a metodologia proposta por Cezarroto, Battaiola e Alves que trata de metáforas e objetos de aprendizagem [1], o qual se propõe a decompor características ao estruturar uma análise em relação ao conteúdo abordado, à construção da atividade e à metáfora proposta.

### 5 PROBLEMÁTICA TRABALHADA

O processo de desenvolvimento de parte das mecânicas secundárias de *Cyberspace CodeX* baseou-se na necessidade de trabalhar conceitos relacionados ao contexto criado para a narrativa e o universo do jogo, e a proposta apresentada por um cliente, definido na disciplina de Projeto Integrado I, de abordar de forma lúdica vários conceitos relacionados a lógica de programação.

O cliente, professor Ernesto Trajano, docente responsável pela disciplina de Programação I na matriz curricular do curso de Sistemas e Mídias Digitais, estabeleceu problemáticas principais, as quais formaram a base para a construção de um sistema de programação que focasse nas dificuldades apresentadas e pudesse ser utilizado como ferramenta auxiliar de ensino.

Tratando-se de um jogo ambientado em um mundo virtual, com diversos conceitos representando, de maneira icônica, elementos das Tecnologias da Informação e Comunicação e da Cibercultura, era interessante e adequado o desenvolvimento de mecânicas que se baseassem na imersão e fossem relacionadas com esses conceitos, dando ao jogador uma maior sensação de aproximação com o mundo e inserindo a proposta de aprendizagem requerida pelo cliente.

Para isso, foi pensado em um sistema que guarda diversas semelhanças com a programação em blocos utilizada em programas simples de criação de jogos como o *Stencyl* e *Scratch*, onde o jogador pudesse intervir nas habilidades das personagens, modificando atributos relacionados, estruturas de comparação e de atribuição, de maneira a dar liberdade para a interação com um código para que ele pudesse ver o resultado direto de suas ações no ambiente de jogo.

As recomendações passadas pelo professor Ernesto foram voltadas para as dificuldades de compreensão e utilização de determinados elementos de lógica de programação e funcionamento de estruturas básicas. As principais diretrizes passadas foram:

- **Apresentar o conceito de variável como um valor relacionado a um nome específico e a diversidade de tipos de variáveis.**

Parte dos estudantes que estão em seu primeiro contato com noções de programação demonstram ter problemas para assimilar o conceito de variáveis como valores associados a um nome atribuído. Dessa forma, noções como a de operações com variáveis tornam-se de difícil entendimento. Outro ponto relevante para a compreensão é a diversidade de tipos de variáveis, as operações possíveis entre esses tipos e a melhor utilização de cada um deles.

- **Trabalhar a diferença entre variáveis globais e locais.** Outra ideia cuja compreensão é relativamente difícil é o conceito de variáveis que são manipuladas localmente e que podem ser manipuladas globalmente. Devido a várias operações lógicas necessitarem dessa manipulação para funcionarem, é necessário fixar o conteúdo relativo a distinção e ao funcionamento de operações com esses tipos de variáveis.
- **Trabalhar condições lógicas, estruturas comparativas, condicionais e de repetição.** Operadores lógicos e comparações são complexos de compreender ao se trabalhar pela primeira vez a ideia da lógica na programação. Ao trabalhá-los em um âmbito mais complexo, como estruturas condicionais e de repetição, o aluno necessita do embasamento para gerar corretamente essas estruturas, assim como a melhor forma de utilizá-las em um programa.
- **Abordar o conceito de Lista, Vetor, operações de Busca e Ordenação.** Sendo uma ideia mais avançada, vista no final da ementa da disciplina, trabalhar com esses conceitos é, apesar de necessário, um aspecto secundário dos requerimentos apresentados. A construção do conhecimento para esses tópicos é válida para a criação de aplicações mais complexas, mas também é um trabalho mais elaborado que os conceitos básicos de lógica de programação.

Cada um desses tópicos foi avaliado e inserido no conceito do jogo na forma de uma mecânica secundária, orbitando e auxiliando as interações primárias do jogador com o jogo e baseadas principalmente no sistema de programação proposto.

## 6 DESENVOLVIMENTO DA ANÁLISE

As mecânicas foram analisadas através dos critérios sobre os conteúdos transmitidos, as atividades que transmitem esse conteúdo e a metáfora que relaciona o conteúdo e o contexto do jogo à atividade proposta. Dessa forma, as mecânicas relacionadas aos assuntos buscam ser elementos facilitadores da compreensão de determinados aspectos da lógica de programação.

### 6.1 Conceito de variável como um valor relacionado a um nome e a diversidade de tipos de variáveis

Através dos atributos das personagens (características como Ataque, Defesa, Vitalidade, etc.) o jogo aborda a relação entre valores e nomes, o que é a base da dificuldade apontada ao se trabalhar com variáveis (Figura 5). No sistema de programação, os nomes das variáveis são utilizados em vez de seus valores numéricos armazenados, dando ao jogador a noção de trabalhar com o nome da variável em operações matemáticas e lógicas.

Dentro do sistema de programação, as variáveis são exibidas de forma esteticamente diferente. Diferença de cores e formas dão um retorno de que determinadas variáveis são diferentes entre si, tanto no aspecto formal (variáveis numéricas ou textual), quanto nas suas possibilidades de utilização (operações matemáticas, lógicas, comparação de textos). Caso a forma das variáveis não se encaixe



Figura 5: Tela de atributos do jogo *Cyberspace CodeX*

no método construído, devido a sua utilização errada, uma mensagem de erro é retornada para o jogador.

Há uma analogia que baseia a relação entre os valores numéricos e os atributos, com essa relação sendo a mensuração das capacidades de cada personagem em termos numéricos. Da mesma forma, essas variáveis correspondentes as capacidades das personagens são um paralelo com a ideia de variáveis guardando características utilizadas em classes e em funções de linguagens de programação.

Relacionada a diversidade das variáveis existentes, a metáfora trata do conceito de códigos programados que podem sofrer problemas de acordo com o mau uso dos tipos de variáveis em sua composição. A dinâmica da interação é relacionada com a estética e a estrutura da programação em blocos que também utilizam de cores e formas para diferenciar seus elementos.

### 6.2 Trabalhar a diferença entre variáveis globais e locais

Através de um sistema chamado *PermaData*, um repositório de pontos que podem ser redistribuídos entre todas as personagens, o jogador pode modificar os atributos delas, construindo novas estratégias e possibilidades para seus aliados. Esses pontos podem ser alterados entre os combates e são compartilhados pelas personagens, de forma que cabe ao jogador controlar a distribuição desses pontos.

A metáfora associada é baseada na aplicação das variáveis globais em programação. Elas são, por exemplo, utilizadas na modificação das variáveis locais. A própria alteração da variável global associada ao *PermaData* é perceptível pelo jogador, fazendo um paralelo com a mudança de variáveis globais em um código de programação.

### 6.3 Trabalhar condições lógicas, estruturas comparativas, condicionais e de repetição

As condições lógicas e estruturas condicionais fazem parte da formação básica de códigos de programação, compondo parte significativa da lógica programada. Dentro do sistema de programação do jogo, elas compõem uma parte integrante das habilidades das personagens.

As estruturas comparativas fazem parte do Sistema de Reação. Cada personagem possui um Tipo específico que permite reagir a habilidades lançadas contra ela (por exemplo, caso uma personagem tenha o Tipo “Cinético”, ela reagirá a habilidades que sejam do mesmo tipo).

As estruturas de repetição, por sua vez, são trabalhadas na combinação de ataques das personagens, permitindo que vários ataques sejam utilizados em um único turno, determinado por condições, como a relação entre o poder do ataque e a quantidade de golpes desferidos, ou o poder do ataque e a chance de acerto dele.

A metáfora que baseia a presença das condições lógicas e estruturas condicionais é a própria aplicação corriqueira dessas estruturas na programação, inseridos em *Cyberspace CodeX* na forma da presença constante desses elementos no sistema de programação planejado.

Para basear o uso das estruturas comparativas, o Sistema de Reação funciona relacionando comparações entre as personagens

e os ataques dos quais elas são alvo. Dessa forma, é permitido ao jogador realizar edições e relacionar as comparações para gerar as reações dos personagens.

A combinação de ataques é uma analogia às estruturas “for” e “while”, comuns em programação. As combinações de ataques podem utilizar um dos dois arquétipos de iteração. O primeiro é baseado em uma quantidade de ataques determinada com um poder menor, de forma análoga a estrutura “for”. O segundo é baseado em uma quantidade de ataques indeterminada, que ocorrerá até que o ataque falhe, análogo a estrutura “while”.

#### 6.4 Abordar o conceito de Lista, operações de Busca e Ordenação

Para trabalhar com conceito de listas, vetores, busca e ordenação, uma abordagem mais avançada é necessária para que se alcance o resultado desejado. Por ser uma parte do conteúdo mais complexa, abordá-la tornou-se mais um exercício de demonstração do que de prática para o jogador.

A abordagem para o conceito de busca envolve a criação de uma estrutura chamada “scan”, presente no sistema de programação do jogo e tendo a função de realizar buscas nas listas de aliados e de inimigos baseado em um determinado critério.

Aliados e inimigos são organizados em combate na forma de listas, capazes de serem ordenadas pelos critérios definidos (por Vitalidade, pelo Maior ou Menor Ataque etc.), facilitando a visualização do jogador para o estado do combate.

Há habilidades das personagens que se valem de mecanismos de ordenação para funcionarem. Em conjunto com a estrutura “scan”, habilidades reordenam a lista de inimigos e aliados para gerar determinados efeitos, curando os aliados em relação a primeira posição da lista ordenada a partir da Maior Vitalidade ou causando efeitos diferentes em inimigos de acordo com a sua posição nela.

A metáfora expressa para a criação do “scan” refere-se a existência de várias estruturas de busca nas linguagens de programação, apresentada aqui de forma simplificada. As listas são trabalhadas a partir de sua principal característica de diferenciação em relação aos vetores. Listas reordenam-se caso um de seus elementos seja removido, sendo uma ideia implementada na lista de inimigos e aliados, quando algum deles é derrotado em combate.

Para apresentar os conceitos de ordenação, as habilidades de determinadas personagens funcionam como uma espécie de “sort”, reajustando as listas existentes para que elas atendam aos seus critérios e gerem os efeitos desejados.

Em especial, essas metáforas, por não serem tão interativas quanto as anteriores, necessitam de um apoio maior para o usuário, utilizando de um retorno visual que demonstre o funcionamento desses conceitos na prática de programação.

#### 7 CONCLUSÕES E TRABALHOS FUTUROS

O desenvolvimento das mecânicas de *Cyberspace CodeX* representou um grande desafio, principalmente no aspecto de realizar a tradução de conceitos relevantes ao ensino em um jogo não alinhado a proposta de um *Serious Game*. A construção dessas mecânicas, porém, mostrou-se de grande valor para a criação do jogo em si, reforçando a narrativa e adicionando uma nova camada de interação significativa para o jogador.

Ainda em processo de conclusão, todas as análises e iterações feitas sobre o jogo são de grande ajuda para compreender a vasta gama de possibilidades que podem ser alcançadas, não somente com esse projeto, mas pelo potencial dos jogos em si como uma mídia transformadora, capaz de transmitir informações, gerar engajamento e propor experiências enriquecedoras.

Testes preliminares com o MVP de *Cyberspace CodeX* foram realizados e demonstraram que as metáforas propostas eram facilmente compreendidas em relação a jogabilidade, com elementos como o a analogia entre atributos de programação e as pontuações

dos personagens, relacionando um valor a um nome específico, sendo bem compreendidos, assim como a diferença entre variáveis locais e globais presente no sistema *PermaData*.

Problemas apresentados nos testes preliminares partiram da dificuldade de compreensão de algumas metáforas visuais utilizadas, como a movimentação feita através de uma interface auxiliar, em vez de selecionar o destino do personagem diretamente. Outros problemas relacionados com a implementação do MVP também atrapalharam a obtenção de dados mais conclusivos em testes mais longos, o que, no entanto, não impediu que a compreensão das metáforas pretendidas e implementadas fosse executada.

Nesse artigo, parte das mecânicas de *Cyberspace CodeX* foram apresentadas, focando especialmente as criadas para a atual versão do projeto e compreendendo os aspectos que orbitam o sistema de programação construído. Diversas outras mecânicas e metáforas, construídas tanto para a narrativa quanto para a interface do jogo, também embasam diversos aspectos das Tecnologias da Informação e Comunicação, e são extremamente valiosas para a composição da experiência geral proposta.

Como trabalhos futuros, espera-se o lançamento da versão completa de *Cyberspace CodeX*, o aprofundamento das mecânicas desenvolvidas e de sua narrativa ao longo do ano de 2016.

#### 8 AGRADECIMENTOS

Agradecimentos especiais ao orientador do artigo, professor Glau-diney Mendonça, às professoras Georgia Cruz e Inga Saboia por incentivarem o desenvolvimento do projeto e cobrarem mudanças valiosas para a melhoria do produto final, ao professor Ernesto Trajano por auxiliar nas iterações do trabalho como Cliente e ao professor Artur de Oliveira por sua prestatividade e auxílio desde o início.

#### REFERÊNCIAS

- [1] M. A. Cezarotto, A. L. Battaiola, and M. M. Alves. Contribuições na criação de metáforas e analogias para objetos de aprendizagem. In *Proceedings of SBGames 2015 - Simpósio Brasileiro de Jogos e Entretenimento Digital*, 2015.
- [2] J. Cheon and M. Grant. A cognitive load approach to metaphorical interface design: Reconsidering theoretical frameworks. In *Society for Information Technology & Teacher Education International Conference*, volume 2008, pages 1054 – 1059, 2008.
- [3] M. Csikszentmihalyi. *Flow: the Psychology of Optimal Experience*. Harper Perennial Modern Classics Edition, 1990.
- [4] W. Hudson. The whiteboard: Metaphor: a doubleedged sword. *Interactions*, May 2000.
- [5] A. Järvinen. *Games without Frontiers: Theories and Methods for Game Studies and Design*. Tampere University Press, 2008.
- [6] L. Niemeyer. *Elementos de Semiótica Aplicados ao Design*. 2AB, 2016.
- [7] K. Salen and E. Zimmerman. *Rules of Play. Game Design Fundamentals*. MIT Press, 2004.
- [8] P. Schuytema. *A Practical Approach*. Charles River Media, 2008.