CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE CAMPINAS

CURSO DE TECNOLOGIA EM GESTÃO DA TECNOLOGIA DA INFORMAÇÃO

MATHEUS CHRISTIAN OLIVEIRA

**VIRTUAL TABLE: UMA ABORDAGEM VISUAL PARA O ENSINO DE LÓGICA DE PROGRAMAÇÃO**

CAMPINAS/SP

2018

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA

FACULDADE DE TECNOLOGIA DE CAMPINAS

CURSO DE TECNOLOGIA EM GESTÃO DA TECNOLOGIA DA INFORMAÇÃO

MATHEUS CHRISTIAN OLIVEIRA

**VIRTUAL TABLE: UMA ABORDAGEM VISUAL PARA O ENSINO DE LÓGICA DE PROGRAMAÇÃO**

Trabalho de Graduação apresentado como pré-requisito para a conclusão do Curso Superior de Tecnologia em Gestão da Tecnologia da Informação, da Faculdade de Tecnologia de Campinas, elaborado sob a orientação da Profª. Mônica Frigeri.

CAMPINAS/SP

2018

AGRADECIMENTOS

Agradeço primeiramente а Deus que permitiu que tudo isso acontecesse ao longo de minha vida, е não somente nestes anos como universitário, e meus pais, pelo amor, incentivo е apoio incondicional. Agradeço imensamente à minha orientadora Prof. Monica, pelo suporte no pouco tempo que lhe coube, pelas suas correções е incentivos. Não poderia deixar de agradecer todos os professores por me proporcionaram о conhecimento não apenas racional, mas а manifestação do caráter е afetividade da educação no processo de formação profissional. Por fim, a todos que direta ou indiretamente fizeram parte da minha formação, minha eterna gratidão.

RESUMO

Este trabalho foi realizado no intuito de desenvolver um protótipo de uma ferramenta baseada na web chamada Virtual Table, que auxilie alunos no processo de ensino de algoritmos e lógica de programação. A falta de certas competências por parte dos alunos, métodos de ensino inadequados e uma conotação negativa associada aos profissionais da área, estão entre algumas razões que influenciam no índice de desistências dos cursos que contém disciplinas de programação em sua grade curricular, que é um dos mais elevados. A referida ferramenta tem por objetivo facilitar a compreensão de como funciona um algoritmo, oferecendo como diferencial o *gamification*. A partir desse objetivo foi desenvolvida uma pesquisa bibliográfica de caráter exploratório, buscando estabelecer uma relação entre jogos e aprendizado. Essa relação foi a base para a utilização do gamification aplicado ao desenvolvimento do protótipo. O protótipo foi então desenvolvido utilizando a linguagem javascript e o *framework* *Phaser*, tornando possível a execução da ferramenta de qualquer navegador web.

**Palavras-chave:** algoritmos; programação; aprendizagem; gamificação.

ABSTRACT

This project was developed in order to build a web-based prototype tool called Virtual Table, which helps both students in the learning process of algorithms and programming logic. Due to the many distinct factors, such as the lack of some skills required from the students, inadequate teaching methods and a negative connotation associated with professionals in that field, the dropout rate in courses that contains programming subjects in their curriculum is very large. The purpose of this tool is to auxiliate the understanding of how an algorithm works, ofering the gamification as a "game changer". This purpoese led to a exploratory research, which seek for a relationship between games and learning. This relationship was the foundation to the use of gamification applied to the prototype development. The prototype was then developed using the javascript language and the Phaser Framework, making it possible to be acessed from any web browser.

**Keywords**: algorithms; programming; learning; gamification.

LISTA DE FIGURAS

[Figura 1: Exemplo de fluxograma. 12](#_Toc516792241)

[Figura 2: Caixa de Skinner. 16](#_Toc516792242)

[Figura 3: O círculo mágico. 22](#_Toc516792243)

[Figura 4: Quadrante de Deterding. 24](#_Toc516792244)

[Figura 5: Barra de progresso utilizada pelo Linkedin. 25](#_Toc516792245)

[Figura 6: Interface do Duolingo. 27](#_Toc516792246)

[Figura 7: Metodologia MDA aplicada. 30](#_Toc516792247)

[Figura 8: Pong, o primeiro videogame. 32](#_Toc516792248)

[Figura 9: Donkey Kong, 1980. 32](#_Toc516792249)

[Figura 10: Interface do Inkscape. 40](#_Toc516792250)

[Figura 11: Exemplo de Sprite. 41](#_Toc516792251)

[Figura 12: Interface do Atom 41](#_Toc516792252)

[Figura 13: Interface da ferramenta de desenvolvimento do chrome 42](#_Toc516792253)

[Figura 14: Tela inicial do Virtual Table. 43](#_Toc516792254)

[Figura 15: Tela de cadastro do Virtual Table. 44](#_Toc516792255)

[Figura 16: Tela de seleção de níveis do Virtual Table. 44](#_Toc516792256)

[Figura 17: Seleção de sub-níveis do Virtual Table. 45](#_Toc516792257)

[Figura 18: Tela inicial do Virtual Table. 46](#_Toc516792258)

[Figura 19: Contorno indicando que um objeto não pode ser movido. 47](#_Toc516792259)

[Figura 20: Contorno indicando que um objeto pode ser movido. 47](#_Toc516792260)

LISTA DE SIGLAS

TI Tecnologia da Informação

BPO Business Process Outsourcer

MDA Mechanics, Dynamics and Aesthetics

HTML Hypertex Markup Language

W3C World Wide Web Consourtium

CSS Cascade Style Sheet

IDE Integrated Develpment Environment

SUMÁRIO

[1. INTRODUÇÃO 10](#_Toc516792450)

[1.1. JUSTIFICATIVA 11](#_Toc516792451)

[1.2. CONTEXTUALIZAÇÃO 12](#_Toc516792452)

[1.3. OBJETIVOS 13](#_Toc516792453)

[1.3.1. Objetivos gerais 13](#_Toc516792454)

[1.3.2. Objetivos específicos 13](#_Toc516792455)

[2. REVISÃO BIBLIOGRÁFICA 15](#_Toc516792456)

[2.1. ASPECTOS TEÓRICOS DA APRENDIZAGEM 15](#_Toc516792457)

[2.1.1. Aprendizagem segundo a teoria behaviorista 16](#_Toc516792458)

[2.1.2. Aprendizagem segundo a teoria cognitivista 17](#_Toc516792459)

[2.2. GAMIFICATION 19](#_Toc516792460)

[2.2.1. Origem e definição do termo gamification 19](#_Toc516792461)

[2.2.2. Definição de jogo, de acordo com Johan Huizinga 20](#_Toc516792462)

[2.2.3. Características fundamentais dos jogos 21](#_Toc516792463)

[2.2.4. O círculo mágico 22](#_Toc516792464)

[2.2.5. Diferenças entre gamification, jogos e jogos sérios 23](#_Toc516792465)

[2.2.6. Aplicabilidade do gamification 25](#_Toc516792466)

[2.2.7. Gamification na educação 26](#_Toc516792467)

[2.3. DESENVOLVIMENTO DE SOLUÇÕES GAMIFICADAS 27](#_Toc516792468)

[2.3.1. Motivação como elemento central 28](#_Toc516792469)

[2.3.2. Framework M.D.A para desenvolvimento de soluções gamificadas 29](#_Toc516792470)

[2.3.3. Fundamentos do game design 31](#_Toc516792471)

[3. MATERIAIS E MÉTODOS 35](#_Toc516792472)

[3.1. METODOLOGIA DE PESQUISA 35](#_Toc516792473)

[3.1.1. Técnica de pesquisa 36](#_Toc516792474)

[3.2. MATERIAIS 37](#_Toc516792475)

[3.2.1. Tecnologias utilizadas 37](#_Toc516792476)

[3.2.2. Tecnologias Web 37](#_Toc516792477)

[3.2.3. JavaScript 38](#_Toc516792478)

[3.2.4. Phaser Framework 39](#_Toc516792479)

[3.3. DESENVOLVIMENTO DO VIRTUAL TABLE 39](#_Toc516792480)

[3.3.1. Ferramentas de desenvolvimento utilizadas 39](#_Toc516792481)

[3.3.2. Fase de desenvolvimento gráfico 40](#_Toc516792482)

[3.3.3. Fase de codificação 41](#_Toc516792483)

[4. RESULTADOS E DISCUSSÃO 43](#_Toc516792484)

[4.1. APRESENTAÇÃO DO VIRTUAL TABLE 43](#_Toc516792485)

[4.1.1. Funcionamento e mecânicas básicas do Virtual Table 45](#_Toc516792486)

[4.1.2. Interatividade como elemento central 47](#_Toc516792487)

[4.2. GAMIFICATION NO VIRTUAL TABLE 48](#_Toc516792488)

[4.2.1. Direcionando ações através do Gamification 48](#_Toc516792489)

[4.2.2. Trabalhando a motivação intrínseca e extrínseca 49](#_Toc516792490)

[4.3. DESENVOLVIMENTO GUIADO PELO GAME DESIGN 50](#_Toc516792491)

[4.3.1. Aplicação do M.D.A para a criação do Virtual Table 50](#_Toc516792492)

[5. CONSIDERAÇÕES FINAIS 52](#_Toc516792493)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 54](#_Toc516792494)

# INTRODUÇÃO

O ensino universitário apresenta diversos desafios para os alunos e professores, tendo cada curso suas particularidades. Segundo Takahashi (2009), os cursos relacionados à tecnologia, computação e matemática tem uma evasão média de 28%, superior aos 22% da média nacional. Segundo a mesma reportagem, o principal motivo para o abandono destes cursos é o pesado conteúdo de cálculo e lógica.

Contemplando o conteúdo de lógica, a disciplina de algoritmos e lógica de programação presente no plano curricular da maioria dos cursos citados anteriormente, costuma oferecer dificuldades aos iniciantes, o que contribui para o problema da evasão. A natureza desta disciplina exige por parte do aluno uma alta carga cognitiva e disposição para a resolução de problemas, pois são conceitos desconhecidos do grande público. Os ingressantes precisam entender estes conceitos, como estes operam e como utilizá-los para alcançar a resolução de um problema.

Essa natureza prática e pouco teórica da disciplina é também um desafio aos educadores, o que levou ao desenvolvimento de pesquisas na tentativa de elucidar os problemas, e desenvolver ferramentas e metodologias que auxiliassem os professores. Buscando entender as dificuldades encontradas pelos alunos, Ferreira (2005) desenvolveu uma pesquisa com turmas dos cursos de Informática e Sistemas de Informação da Universidade do Planalto Catarinense (UNIPLAC). Dentre os obstáculos relatados, destacam-se a falta de tempo para a realização de exercícios práticos, metodologias de ensino pouco didáticas. Segundo Lemos, Barros e Lopes (2013), o ato de programar exige capacidades cognitivas distintas, como a habilidade de compreender problemas e aplicar a lógica algorítmica para essas situações.

Um dos problemas relatados na literatura, refere-se à ausência de recursos computacionais onde o aprendiz recebe um feedback instantâneo de suas ações, já que testes de mesa tradicionais no papel exigem uma alta capacidade de abstração. Outro problema comum em turmas iniciais de algoritmos é a compreensão incompleta ou até mesmo equivocada de como funcionam todas as estruturas algorítmicas, impossibilitando sua aplicação para a resolução dos desafios propostos. Utilizar uma abordagem de ensino que comece diretamente com codificação, sem identificar as variáveis e estruturas que podem ser utilizadas, é um erro muito comum.

Baseando-se no referencial teórico existente e em estudos desenvolvidos anteriormente, este trabalho busca amenizar os obstáculos existentes entre docentes e discentes no ensino de lógica algorítmica, através de uma ferramenta computacional. As razões que levaram a essa escolha e as justificativas das abordagens adotada serão dissertadas ao longo deste trabalho.

## JUSTIFICATIVA

Um algoritmo é uma sequência de instruções necessárias para realizar de terminada tarefa. Em computação, um algoritmo é um conjunto de instruções que descrevem explicitamente as ações que devem ser realizadas por um software. Quando se desenvolve um software, essas instruções são escritas de forma estruturada e se utilizando de uma linguagem pré-definida, geralmente em baseada no inglês. As instruções são posteriormente traduzidas para linguagem de computacional e podem ser executadas por um computador (BATTISTELLI, 2018).

Sendo a base de toda linguagem de programação, a disciplina de algoritmos é ministrada antes das disciplinas de programação. Para o ensino de lógica algorítmica, é utilizada uma linguagem de sintaxe simples, pois um mesmo algoritmo pode ser escrito de diferentes formas dependendo da linguagem de programação utilizada, sendo as sintaxes relativamente complexas, o que pode tornar o aprendizado confuso inicialmente. O Virtual Table utiliza a linguagem portugol, pelo fato das instruções serem escritas em português e possuir uma sintaxe de fácil compreensão, reduzindo os potencias entraves para o aprendizado.

A natureza abstrata da disciplina de lógica de programação pode ser um obstáculo para os alunos que estão tendo o primeiro contato, habituados a disciplinas com uma pesada carga teórica, sem a prática de exercícios para fixação. O estudo de lógica exige esforço e prática, além de capacidades cognitivas de abstração e raciocínio lógico-matemático. Existem grandes diferenças entre os níveis de aptidão dos alunos de uma mesma turma, o que em teoria exigiria um nível de atenção individualizado por parte dos docentes, ou ao menos uma divisão da turma por grupos com níveis de aptidão semelhante. Considerando a quantidade de alunos por turma em cursos universitários, essas estratégias se mostram inviáveis, sendo adotado um sistema de ensino heterogêneo, que pode ser demasiado rápido para alguns e entediante para outros.

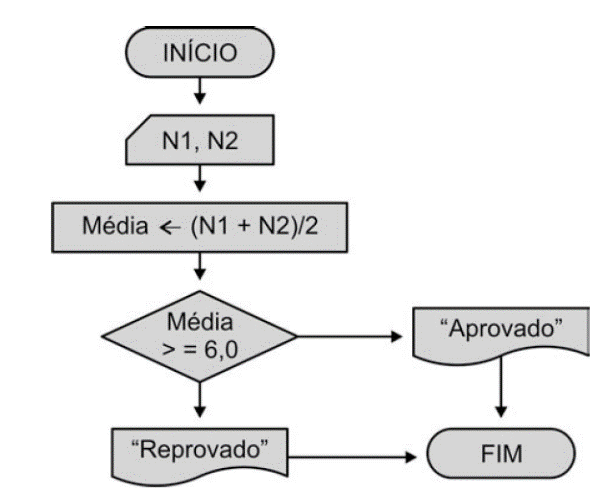
A disciplina de lógica de programação está presente na grade curricular de praticamente todos os cursos da área de T.I, e até mesmo em alguns cursos sem relação direta com esse campo, como engenharia, matemática e física. Sendo uma disciplina de base de todos esses cursos, é indispensável ao aluno compreender o funcionamento de um algoritmo, pois será necessário para a realização de futuras disciplinas, e potencialmente um diferencial no mercado de trabalho.

## CONTEXTUALIZAÇÃO

Em termos técnicos, um algoritmo é uma sequência de instruções logicamente estruturada e finita, que leva a resolução de um problema ou execução de uma tarefa. Etimologicamente, é uma versão latinizada do nome de Al-Khowarizmi, matemático árabe do século 9 (NICOLIELO, 2010). Em computação, um algoritmo é um conjunto de instruções que descrevem explicitamente as ações que devem ser realizadas por um software. Essas instruções podem indicar que diferentes ações devem ser tomadas em reação à uma ou mais informações anteriores. Essas informações são chamadas de entradas, e o resultado das ações baseadas nela são chamadas de saídas.

Um algoritmo pode ser representado visualmente através de fluxogramas, que são diagramas que empregam formas geométricas e traços direcionais para ilustrar os passos do algoritmo. A figura 1 mostra um algoritmo simples, que recebe 2 variáveis, calcula a média desses valores, e retorna uma saída de acordo com a média obtida.

**Figura 1**: Exemplo de fluxograma.



**Fonte**: Battistelli (2018).

Outra forma de representação visual de um algoritmo é utilizando pseudocódigos, que são um conjunto de instruções escritas que indicam explicitamente as ações que devem ser tomadas. Pseudocódigos possuem uma sintaxe mais simples quando comparadas com linguagens de programação reais como Java ou C, mas ainda mantém muitas similaridades. Essa simplificação da sintaxe é feita de maneira a facilitar o primeiro contato dos estudantes com a lógica algorítmica. O Portugol, ou português estruturado, é uma forma de pseudocódigo que utiliza comandos em língua portuguesa para representar os passos de um algoritmo. O algoritmo ilustrado na figura 1 seria escrito dessa maneira em Portugol:

Algotimo média;

Variável

N1, n2, média: inteiro;

Inicio

Média <- (n1 + n2)/2;

Se (média >= 6) escreva “Aprovado”;

Senão escreva “Reprovado”;

Fim;

## OBJETIVOS

### Objetivos gerais

O objetivo deste trabalho é a análise da utilização da gamificação para o ensino de algoritmos e lógica de programação, e utilizar essa análise para a proposição e desenvolvimento do protótipo de ferramenta digital de ensino, chamada Virtual Table. A análise tem como objetivo compreender de que maneira o *gamification* pode ser aplicado no ensino para facilitar o aprendizado da lógica algortítmica, e baseado nesta pesquisa, projetar e construir o protótipo do Virtual Table. Considerando a importância deste conteúdo para as futuras disciplinas e para o mercado de trabalho, esse protótipo tem como característica a utilização do *gamification* como diferencial para facilitar a compreensão do funcionamento de um algoritmo, utilizando para isso elementos de jogos e buscando tornar o aprendizado uma experiência gratificante.

### Objetivos específicos

Para atingir seu objetivo com excelência, este projeto tem como objetivos secundários o esclarecimento de questões indispensáveis para o desenvolvimento do VirtualTable, sendo elas:

* Identificar de que maneira o conhecimento é construído, utilizando para isso o referencial teórico existente;
* Compreender o que é gamification, como o mesmo funciona, e como pode ser aplicado para o desenvolvimento de uma solução de aprendizagem;
* Propor e desenvolver um protótipo de uma ferramenta digital chamada Virtual Tabel que utilize da gamificação diferencial para o ensino de algorítmos.

No primeiro capítulo é apresentado um contexto histórico nacional do ensino de lógica de programação nas universidades brasileiras, abordando a maneira pela qual este é conteúdo é apresentado atualmente, quais as metodologias e métricas de ensino nas universidades que contém disciplinas de algoritmos e lógica de programação na sua grade curricular. Na mesma seção serão discorridos também sobre as principais dificuldades encontradas no ensino e aprendizado desta disciplina, e procurando entender suas possíveis causas e potenciais consequências, utilizando como base a bibliografia disponível e trabalhos e pesquisas desenvolvidas anteriormente.

No segundo capítulo é apresentada a revisão da bibliografia utilizada durante o desenvolvimento deste trabalho, sendo apresentado neste capítulo os principais teóricos e suas ideias e hipóteses a respeito do ensino de lógica de programação. Buscando entender de forma mais aprofundada como funciona o aprendizado e como se dá o processo de assimilação de novas informações pelo cérebro humano. Serão apresentados também neste capítulo os principais conceitos de psicologia da educação, e uma breve contextualização das teorias desenvolvidas até o momento. Por fim, ainda no segundo capítulo, será apresentado o conceito de gamificação, e como esta abordagem pode ser útil para o aprendizado de lógica de programação.

O terceiro capítulo é dedicado a apresentar a metodologia de pesquisa utilizada na revisão bibliográfica, e as tecnologias utilizadas no desenvolvimento do Virtual Table. Ainda no terceiro capítulo é discorrido sobre o processo de desenvolvimento e programação. No quarto capítulo, é demonstrado de que maneira o Virtual Table se utiliza das bases conceituais adquiridas na pesquisa bibliográfica para facilitar o aprendizado de lógica de programação. O último capítulo deste trabalho é dedicado às últimas considerações à respeito do desenvolvimento do mesmo, as principais descobertas e desafios encontrados nas fases de pesquisa e desenvolvimento e o quanto trabalhar em um projeto como esse agregou do ponto de vista pessoal e profissional.

# REVISÃO BIBLIOGRÁFICA

## ASPECTOS TEÓRICOS DA APRENDIZAGEM

O objetivo do Virtual Table é ajudar o aluno a internalizar os conceitos de lógica de programação, sendo assim é necessário compreender de que maneira ocorre o aprendizado, e como a ferramenta pode ajudar nesse processo. Mas antes de compreender como a mesma funciona, é necessário definir o que é aprendizagem. Do ponto de vista biológico, o aprendizado pode ser explicado de forma simplificada como a criação e reforço de estruturas sinápticas[[1]](#footnote-1). Sinapses se tornam mais duradouras conforme são reforçadas, transferindo as informações da memória de curto prazo para longo prazo. Esse reforço pode ser resultado tanto do esforço quanto do envolvimento emocional do indivíduo no momento da ação (BRUNA, 2012).

Segundo o dicionário de língua portuguesa Aurélio (2018), a aprendizagem pode ser definida genericamente como o ato de adquirir conhecimento, estudar e compreender. De acordo com essa definição, aprender um novo idioma e a sequência de botões de uma máquina de café podem ser ambos considerados como aprendizado. Conceituar o aprendizado é uma tarefa mais complexa do que aparenta inicialmente. Este trabalho irá considerar como aprendizado uma mudança relativamente permanente no comportamento, resultante da experiência:

“Processo por meio do qual conhecimento, valores, habilidades e competências são adquiridos ou modificados como resultado de estudo, experiencia, formação, raciocínio e observação.” (ALVES, 2015).

A curiosidade de entender como funciona o aprendizado existe desde a antiguidade clássica, e ao longo do tempo diversas pesquisas e estudos foram desenvolvidos nesse campo. O ramo da psicologia chamado psicologia da educação tem como objetivo a compreensão do aprendizado para o desenvolvimento de técnicas pedagógicas para serem usadas em sala de aula. Nesse cenário, a corrente de estudo conhecida como behaviorismo analisa o processo de aprendizagem sob o ponto de vista externo, enquanto a corrente conhecida como cognitivismo analisa os estados mentais, de um ponto de vista interno (ANDRADE, 2009). Ambas correntes serão utilizadas para nortear o desenvolvimento do Virtual Table.

### Aprendizagem segundo a teoria behaviorista

Também chamada de comportamentalista, a corrente conhecida como behaviorismo tem sua origem na palavra inglesa *behavior*, que pode ser traduzida diretamente como comportamento. O estudo do behaviorismo se restringe ao ambiente como modelador do comportamento, desconsiderando a análise de características internas como estados mentais. O pressuposto de que todo e qualquer indivíduo pode ter seu comportamento modelado através de influências externas começou com o médico russo Ivan Pavlov. Utilizando experiências com cães, Pavlov criou a teoria dos reflexos condicionados, utilizada até hoje para o adestramento de animais. Mas o fundador do behaviorismo como escola foi o psicólogo américa John B. Watson, que teve como seguidor B. F. Skinner, figura mais proeminente e maior divulgador do behaviorismo (RODRIGUEZ, 2014).

Burthus Frederick Skinner foi um psicólogo norte-americano que focou sua pesquisa na aplicação do Behaviorismo na educação e aprendizagem. Skinner acreditava na premissa que todo ser humano é passível de controle através do padrão de estímulo-resposta: ao observar a resposta obtida com determinado estímulo, é possível moldar o comportamento ao reforçar os estímulos para respostas desejadas, e punir ao obter uma resposta indesejada. Esse método cientificamente rígido permitiu que Skinner realize experimentos apenas com animais (ratos e pombos), criando um dispositivo conhecido como caixa de Skinner, representado na Figura 2: a cobaia ficava em uma caixa completamente fechada, apenas com uma alavanca e um fornecedor de alimento. Quando o rato aciona a alavanca, um pouco de comida é despejado dentro da caixa. Ao repetir esse processo, a cobaia “aprende” que deve acionar a alavanca quando precisar de comida (RODRIGUEZ, 2014; ANDRADE, 2009).

**Figura 2:** Caixa de Skinner.



**Fonte**: Andrade (2009).

Através dos estudos de Skinner, o behaviorismo define que o mesmo processo pode ser aplicado no ensino, estimulando ações desejadas dos alunos com recompensas e reprimindo ações indesejadas. Skinner defendia principalmente a importância do reforço positivo para aprendizagem, criticando a maneira como os sistemas de ensino focam tem tedências contrárias, focando em punições. Observações posteriores de Skinner concluíram que os alunos possuem ritmos de aprendizados próprios, e que os mesmos estão mais propensos a aprenderem quando estimulados a aprender através da descoberta, ao invés de serem expostos ao conteúdo. Partindo dessa premissa, o behaviorismo define que o conteúdo a ser ensinado deve ser fragmento em partes menores, e ensinado ao aluno de maneira progressiva e com estímulos e reforços apropriados (RODRIGUEZ, 2014).

As principais críticas ao behaviorismo vêm do fato que os teóricos dessa corrente trabalham exclusivamente com estímulos externos, desconsiderando aspectos como consciência, inteligência, memória ou emoção. Esses são aspectos centrais na pesquisa do cognitivismo, que tem como premissa que o aprendizado não é moldado, mas construído através da interação entre indivíduo e o ambiente. O cognitivismo não descarta o uso de estímulos positivos, mas considera os estados mentais do aluno influenciam o aprendizado na mesma proporção que o ambiente.

### Aprendizagem segundo a teoria cognitivista

O biólogo norte-americano Jean Piaget é o maior expoente do cognitivismo. Piaget dedicou sua vida ao estudo da formação da inteligência humana, e como biólogo, conclui que o comportamento dos seres vivos não é apenas inato nem resultado de condicionamento, mas constantemente construído e reconstruído na interação dos seres vivos com o meio. Segundo Argento (2018), “quanto mais complexa for esta interação, mais “inteligente” será o indivíduo”.

Para decifrar o conhecimento, a teoria cognitivista trabalha com o conceito de **esquemas** como estrutura básica do conhecimento. Os esquemas podem ser compreendidos como estruturas cognitivas que existem para identificar e processar estímulos. Uma criança possui um esquema de “cachorro”, associando a ele características físicas (quadrúpede, focinho, pelagem), maneira que se move, latido. Qualquer animal que apresente características semelhantes será entendido pela criança como um “cachorro”. Nesse cenário o esquema é utilizado para processar os estímulos visuais e auditivos e identificar o que a criança vê. O conceito de esquema vai de encontro à premissa behaviorista, pois os mesmos não podem ser observados nem mensurados, possuindo uma natureza hipotético-dedutiva (ARGENTO, 2003; ANDRADE, 2014).

Considerando o esquema como a base do conhecimento, a teoria cognitivismo estabelece que o aprendizado acontece através da modificação de esquemas anteriores. Não é possível a criação de um “novo” esquema, pois a aquisição de novos conhecimentos necessita dos esquemas já existentes. Segundo Piaget (1982, apud ARGENTO 2003), “não existe estrutura sem gênese, nem gênese sem estrutura”. Os esquemas de uma criança são em sua maioria de natureza reflexa, que são adaptados para criação de novos esquemas mais complexos conforme o crescimento e desenvolvimento intelectual do indivíduo.

Os processos de modificação dos esquemas existentes são chamados na teoria cognitivista de **assimilação** ou **acomodação**. Quando uma criança passa por novas experiências, essas informações deverão ser integradas à esquemas prévios. Utilizando o exemplo do cachorro, isso ocorre quando a criança conhece outras raças de cães distintas entre si, mas que ainda mantém características básicas em comum. Esse é o processo de assimilação, quando a criança aprende ao tornar os esquemas prévios mais complexos ao integrar novas informações. A acomodação acontece quando a criança é apresentada a um conceito completamente novo e diferente de tudo o que já vivenciou. Considere-se a situação hipotética de que uma criança veja um cavalo pela primeira vez: nessa situação, o animal será inicialmente associado à um cachorro devido as semelhanças físicas, mas ao ser corrigida, a criança vai “acomodar” o conceito do cavalo através da modificação do conceito de “cachorro” (ARGENTO, 2003; ANDRADE, 2014).

Segundo o cognitivismo, o aprendizado é resultado das constantes assimilações e acomodações. Aprender um novo conteúdo envolve acomodar estruturas cognitivas existentes para assimilar o que está sendo aprendido. Novas informações causam um desequilíbrio dos esquemas existentes, e o conhecimento é formado na busca pelo equilíbrio entre acomodação e assimilação dos novos esquemas, que posteriormente serão a base para o aprendizado de novos conhecimentos. Por essa razão, a teoria cognitivista enfatiza o fato de dentro do ambiente pedagógico, o conteúdo deve ser sistematicamente organizado para que seja apresentado ao aprendiz de forma progressiva, facilitando dessa forma o processo de equilibração. Pelo fato de considerar a carga cognitiva do aluno como fundamental e ponto de partida da teoria cognitivista, o aluno é enxergado como principal responsável pelo aprendizado, ressaltando o valor do ensino através da descoberta e exploração.

As principais críticas ao cognitivismo se devem ao fato de que o mesmo não fornece um método de ensino propriamente dito (ARGENTO, 2003). Diferente do behaviorismo, que é considerada uma corrente epistemológica[[2]](#footnote-2), o cognitivismo tem como objetivo compreender a gênese do conhecimento, e dessa forma facilitar o processo de ensino-aprendizagem. Já no behaviorismo, a ênfase é na pesquisa é a criação de métodos de ensino baseada em uma abordagem científica, por isso ambas são tão divergentes na maneira que enxergam o aluno. Mas apesar das diferenças, ambas concordam no fato de que o aluno deve ser livre para buscar as repostas. O professor assume o papel de educador, de facilitador do conhecimento. Por essa razão, o educador é responsável por organizar o conhecimento, para que seja exposto ao aprendiz de acordo com sua capacidade intelectual.

Ambas as teorias também convergem no ponto em que o aprendizado deve ocorrer de forma progressiva, e o enfoque nas ferramentas de ensino como reforçadores positivos ao invés de priorizar métodos de punição. Nesse cenário, o Virtual Table pode ser considerado mais uma das ferramentas de aprendizagem à disposição do aluno, pois em conjunto com o Gamification, fornece o ambiente e os estímulos necessários para encorajar o aprendizado, onde o aluno interage com o objeto de estudo cuidadosamente organizado.

## GAMIFICATION

De forma simplificada, *gamification* ou gamificação pode ser entendido como a ação de atribuir características de jogos em atividades que não possuem nenhuma relação com estes. É a tentativa de tornar uma atividade sem apelo em algo tão interessante e divertido quanto um jogo. O objetivo desta seção é buscar compreender a essência do *gamification*, entender como este funciona, e quais são suas características fundamentais. A linha que separa um jogo propriamente dito de uma atividade gamificada é tênue e nem sempre esses limites são muito claros. Será necessária uma definição clara destas diferenças antes de destrinchar o funcionamento do *gamification*.

### Origem e definição do termo *gamification*

Apesar de ser atualmente uma palavra relativamente popular e aparentemente nova, o termo *gamification*, ou gamificação em sua versão aportuguesada, foi cunhada pelo consultor britânico Nick Pelling no ano de 2002 para descrever os serviços oferecidos pela sua consultoria chamada *Conundra*. Pelling oferecia consultoria para tornar interfaces de dispositivos eletrônicos parecidas com jogos. Como descrito no site da própria Conundra: “todo dispositivo se tornará um jogo”. Apesar da consultoria não obter sucesso, a palavra se popularizou e o termo foi alçado ao *Google Trends[[3]](#footnote-3) 8* anos depois (PELLING, 2018; BURKE, 2015).

O sentido inicial de Pelling era exclusivo para os dispositivos eletrônicos, mas ao longo dos anos o termo adquiriu a definição aceita atualmente, mais conceitual e relacionada à natureza dos jogos. A gamificação é oficialmente definida pelo dicionário de Oxford como:

“Aplicação de elementos típicos de jogos (pontuação, competição e regras de jogo) para outras áreas de atividade, tipicamente como uma técnica de marketing online para encorajar engajamento à determinado produto ou serviço” (OXFORD, 2018).

Como mencionado na definição, as aplicações mais comuns *gamification* são como estratégias de marketing. Os programas de milhas aéreas ou qualquer estabelecimento comercial que utilize programas de acúmulo de pontos para serem trocados por vantagens pelos clientes está utilizando *gamification*. Mas a utilização de pontos e recompensas são apenas algumas das características dos jogos, e a aplicação do *gamification* não se resumo à programas de vantagens.

Assim como o conceito de aprendizado, definir o que é exatamente um jogo é uma tarefa mais complexa do aparenta inicialmente. Segundo Huinzinga (2000), a maioria dos idiomas possui apenas palavras genéricas para se referir a qualquer tipo de atividade lúdica, como os verbos **divertir** em português e ***fun*** em inglês. Apesar do sentido distinto, as palavras “**jogar**” e “**brincar**” são comumente utilizadas com o mesmo significado em diferentes contextos. Compreender o que é um jogo e quais são suas características básicas é necessário para entender de que maneira funciona o *gamification*.

### Definição de jogo, de acordo com Johan Huizinga

Huizinga foi um historiador holandês que buscou em sua pesquisa compreender a relação do homem com o lúdico, discutindo qual é o propósito de atividades lúdicas como jogos e brincadeiras: seriam uma função biológica herdada pela evolução ou um traço cultural desenvolvido em conjunto com a civilização? Utilizando como referencial pesquisas e teorias de outros autores, o autor encontra evidências para ambos: apesar da própria definição de do jogo presumir a necessidade de contexto social, em alguns grupos de animais, especialmente os sociáveis, é possível observar atividades semelhantes a brincadeiras, mas não é possível afirmar que as mesmas são puramente biológicas, pois não há algum propósito biológico aparente para estas atividades.

Esta natureza dúbia e a aparente ausência de um objetivo específico torna complexa e atualmente impossível a tarefa de definir a função exata de um jogo. Huizinga (2000) conclui que ambas correntes teóricas sobre a origem do jogo (biológica e cultural) podem ser aceitas como corretas. O autor concentra seus esforços em tentar compreender o que define um jogo: porque um jogo é tão fascinante a ponto de manter os participantes tão imersos em uma atividade? O questionamento do autor é o exato objetivo do *gamification*, que busca entender o que torna um jogo divertido para aplicar em atividades que não estão relacionadas a nenhum contexto lúdico. Huizinga não chegou à uma definição objetiva, mas observou algumas características em comum que tornam determinada atividade passível de ser considerada um jogo.

### Características fundamentais dos jogos

A primeira – e fundamental – característica de um jogo, é a **liberdade**: um jogo é primariamente uma atividade voluntária, geralmente considerada pelos participantes como algo dispensável, praticado nas “horas vagas”. Retomando a questão biológica, os jogos estão distantes de qualquer necessidade fisiológicas. Jogar jamais é uma tarefa, o que leva diretamente à segunda característica: **o jogo não reflete a realidade**. Um jogo pode absorver traços e até mesmo a essência da realidade, mas não busca imitar a mesma. O jogo cria uma realidade própria e autônoma, delineada por regras e limites livremente aceitos pelos participantes. É essa distância da “vida real” que garante o aspecto lúdico e mágico dos jogos. A terceira característica distinta de um jogo é o **isolamento e limitação**: essa limitação não se resume apenas as delimitações de tempo e espaço, mas também o que é permitido ou não (regras), de que maneira deve ser jogado e quais são os seus objetivos. Os participantes devem ter conhecimento destas limitações antes de imergirem na realidade do jogo (HUIZINGA, 2000).

Huizinga destaca que estes limites distinguem os jogos de outras atividades lúdicas como brincadeiras ou rituais. Apesar de brincadeiras proporcionarem a mesma liberdade e “afastamento” da realidade, as mesmas não possuem os limites de um jogo (regras, demarcações e objetivos), e quando possuem, são subjetivos e flexíveis. E por mais que rituais aconteçam em um lugar e tempo definidos, possuindo as próprias regras, os mesmos geralmente possuem um propósito além da diversão pura, podendo ser um rito de passagem ou uma decisão política.

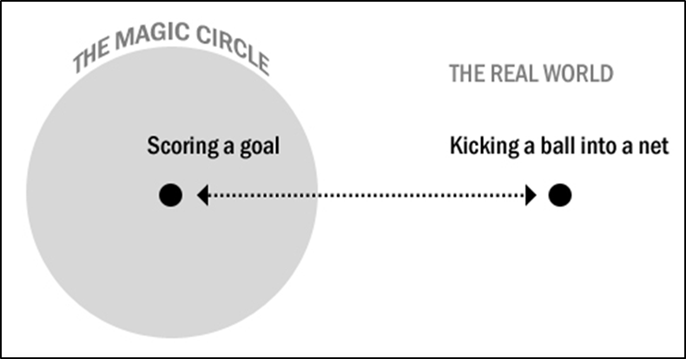
Considerando as características descritas nos parágrafos anteriores, pode-se concluir que o jogo é uma atividade autônoma, ou seja, tem finalidade nela mesma e, portanto, não existem razões lógicas que expliquem a razão dos jogadores que participam do mesmo. Jogamos determinado jogo por que gostamos e porque o mesmo nos oferece algum significado. A relação de cada jogador com um jogo é pessoal, assim como os motivos que levam o mesmo a jogar.

### O círculo mágico

Para ilustrar as características descritas na seção anterior, Huizinga criou o conceito do círculo mágico, ilustrado na figura 3. O perímetro do círculo pode ser interpretado como o limite que separa o jogo da realidade, onde as ações cotidianas se tornam algo mais significativo. Conforme exemplificado na figura 3, dentro de uma partida de futebol, o ato simples de chutar uma bola dentro de uma rede se torna algo mais significativo, um “gol”. Mas esse significado só existe dentro do círculo mágico, o “gol” não tem valor nenhum fora da realidade do jogo.

Os jogos evoluíram em conjunto com a humanidade e com o contínuo progresso tecnológico foram criadas novas formas de se jogar, começando com simples traços na areia para complexos jogos de tabuleiro com possibilidades quase infinitas. A popularização dos computadores e a ascensão da era digital possibilitou o surgimento dos videogames, que apesar de suas características distintas, em essência, ainda são jogos e se encaixam no conceito do círculo mágico. O conceito do círculo mágico é fundamental para compreensão dos jogos, e por consequência, para a criação de soluções gamificadas.

**Figura 3**: O círculo mágico.



**Fonte**: Huinzinga (2000, apud VADER, 2012).

A utilização de pontos, rankings e medalhas são geralmente os primeiros termos associados à gamificação. Embora realmente façam parte da “caixa de ferramentas” da gamificação, os mesmos sozinhos não garantem uma solução gamificada, se não acompanhados das características fundamentais dos jogos. Uma solução gamificada deve oferecer algo que seja significativo aos seus participantes, um sentido que os motive a participarem do “jogo”. Segundo Burke (2015), o uso indiscriminado de pontos e recompensas leva à um efeito contrário, como definido pelo autor, uma “fadiga de placar”.

### Diferenças entre *gamification,* jogos e jogos sérios

O que difere uma solução que utiliza *gamification,* de um jogo prorpriamente dito e um jogo educativo? Baseado nas definições utilizadas até agora, pode-se concluir que todas partilham do mesmo princípio: manter alguém tão interessado em uma atividade como se esta fosse um jogo. Como definido pela autora Flora Alves:

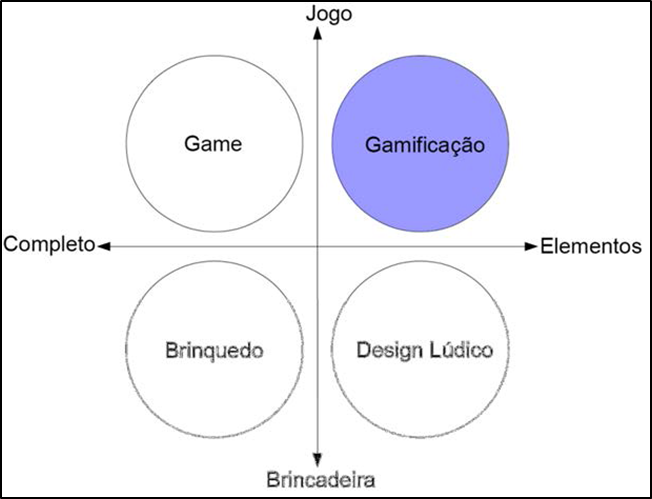
“Estar baseado em games implica na construção de um sistema na qual aprendizes, jogadores ou consumidores se engajarão num desafio abstrato, definido por regras claras, interagindo e aceitando feedback com o alcance dos resultados quantificáveis e com a presença de reações emocionais (ALVES, 2015) ”.

O desenvolvimento de soluções gamificadas tem como meta atingir o engajamento voluntário e espontâneo de seus participantes utilizando para isso características de jogos, como o aspecto lúdico, a liberdade e as limitações. Em outras palavras, a gamificação busca transportar os participantes para o círculo mágico de Huizinga (ALVES, 2015). Se ambos (*gamification* e *games*) se utilizam do conceito do círculo mágico, o que indica quando uma solução gamificada se torna de fato um jogo? Esse limite não é muito claro, e muitas vezes subjetivo. Um jogo ou game tem como característica primordial ser desinteressado, uma ação que tem fim nela mesma e não traz lucros materiais imediatos aos participantes. Os jogadores jogam por pura e simples diversão. Soluções gamificadas podem até ser divertidas, mas utilizam da diversão para atingir uma meta, como conquistar um cliente, ensinar um procedimento ou influenciar um comportamento (BURKE, 2015).

Para ilustrar como esses limites podem se tornar confusos, Burke (2015) cita como exemplo o aplicativo social de turismo *Foursquare*: a gameficação foi inicialmente utilizada como uma forma tornar a aplicação mais interessante, já que o aplicativo dependia da base de usuários para alcançar o sucesso, e possuía poucos até aquele momento. Um dos principais elementos gamificados utilizados pelo aplicativo foi oferecer diferentes títulos de acordo com as conquistas acumuladas. Essas conquistas eram alcançadas de acordo com os *check-ins[[4]](#footnote-4)* do usuário, garantindo títulos como “prefeito”, ou “baladeiro”. Posteriormente, os elementos de games se tornaram tão infundidos na plataforma que definir se o mesmo era um jogo de realidade aumentada ou um aplicativo de turismo se tornou critério dos usuários. Desde de 2014 o Foursquare deixou de utilizar a gamificação e transferiu as funções de check-in para outro aplicativo, chamado Swarm (BRAGA, 2014).

Uma forma objetiva de situar o gamification e o isolar de outros conceitos semelhantes à primeira vista é utilizar o quadrante criado por Alves (2018 apud DETERDING, 2018), ilustrado na figura 4. No quadrante de Deterding, o eixo vertical representa jogos e brincadeiras como opostos, enquanto o eixo vertical representa a utilização total ou apenas partes dos mesmos. Em um dos extremos do quadrante se encontram as brincadeiras, que são a diversão em sua forma mais pura. No mesmo eixo podemos situar o que são conhecidos como jogos sérios, também chamados de jogos educativos. Estes são realmente jogos, mas geralmente utilizados em contextos educacionais ou de treinamento. Na outra metade inferior dos quadrantes pode-se utilizar como exemplo dinâmicas de grupo, que possuem elementos de brincadeira, mas não são atividades gamificadas propriamente ditas. E por último situa-se o gamification, que não é um jogo, mas assimila características de um (ALVES, 2015; apud DETERDING, 2018).

**Figura 4**: Quadrante de Deterding.



**Fonte**: ALVES, 2015, apud DETERDING, 2018.

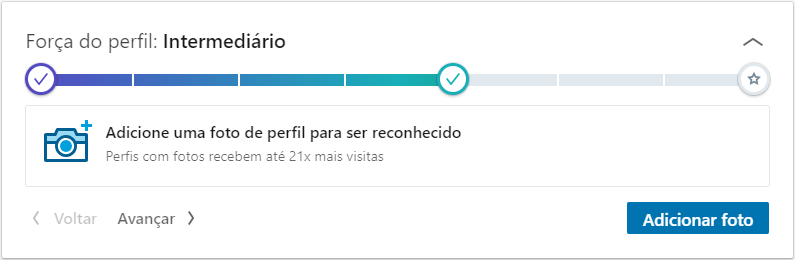
Baseado no quadrante de Deterding, o Virtual Table pode ser enquadrado na categoria de gamification, pois se propõe a ensinar lógica de programação em um processo divertido, mas não é um jogo, e consequentemente não é um jogo educativo. O Virtual Table é uma ferramenta gamificada, pois os elementos de jogos estão presentes, mas o objetivo da ferramenta é facilitar o aprendizado, ao invés de entreter.

### Aplicabilidade do *gamification*

A possibilidade de utilização do *gamification* é bastante ampla. A semelhança com jogos o torna atraente para o público em geral, apesar das aplicações mais comuns serem relacionadas às áreas de marketing, vendas e fidelização do consumidor. O *gamification* pode também ser utilizado em um ambiente interno, como em programas de treinamento e capacitação.

Como parte de uma iniciativa chamada “*Make Agents Great*” a Microsoft buscava aumentar o comprometimento de seus agentes de *BPO[[5]](#footnote-5)*, e a gamificação servia aos propósitos definidos. Segundo Rimon (2018), programas de recompensas de acordo com metas já existiam, mas apenas à nível mensal e semestral, se restringindo a uma pequena parcela dos envolvidos. Foi então desenvolvido um sistema de pontuação atrelado às informações quantitativas das chamadas (número de ligações, vendas), que também possuía metas diárias. As metas diárias eram mais fáceis de se alcançar do que metas mensais, tornando os objetivos mais tangíveis, e a pontuação oferecia um senso de progresso e recompensa. A satisfação por atingir os objetivos melhorou o envolvimento dos agentes, resultando num aumento de 10% da produtividade e redução de 12% do absenteísmo (RIMON, 2018).

**Figura 5**: Barra de progresso utilizada pelo Linkedin.



**Fonte**: Linkedin (2018).

O *gamification* pode ser também utilizado de mais maneira mais pontual. O linkedin encontrou na gamificação uma maneira de incentivar seus usuários a terminar de concluir o cadastro das informações de perfil. Implementando uma barra de progresso para indicar o percentual que já havia sido preenchido, garantiu-se um aumento de 20% no índice de cadastros preenchidos. Essa barra indica de maneira visual “o quão forte é seu perfil” de acordo com as informações preenchidas, como ilustrado na figura 5 (ALVES, 2015).

Porém é de extrema importância que se considere o fato de que o *gamification* por si só não é garantia de sucesso. De acordo com Burke (2015), se o produto não possui vantagens intrínsecas em relação à concorrência, não há nada que a gameficação possa ajudar. Deve-se avaliar se o gamification realmente agregará valor ao produto ou serviço em questão. A gameficação não pode magicamente tornar divertido um processo essencialmente burocrático (BURKE, 2015).

### *Gamification* na educação

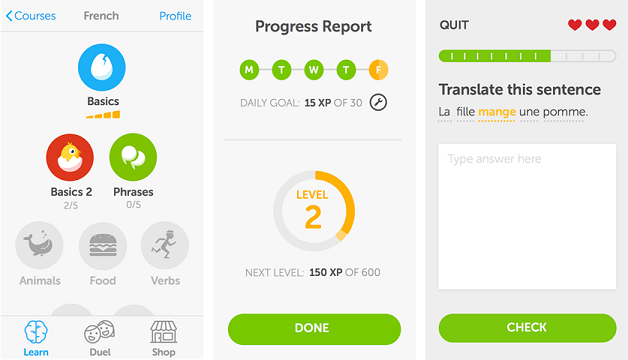
Uma das aplicações que o gamification tem grande potencial para agregar valor é a educação, que é também o objetivo deste trabalho. Com a popularização das mídias digitais, atualmente os profissionais do ensino disputam a atenção do aluno com videogames, mídias sociais e aplicativos de celulares. A gamificação tem sido vista como uma alternativa ao modelo de ensino vigente, buscando um melhor diálogo com as novas gerações.

Entre as tentativas de se aplicar o gamification na educação, pode-se citar a escola Q2L, ou *Quest to Learn*. A Q2L é uma escola pública experimental localizada em Nova York para estudantes do ensino fundamental, com o currículo inteiramente baseado em jogos. A escola busca diferenciar da abordagem quase behaviorista do modelo de ensino vigente atualmente, utilizando uma abordagem auto definida como “*Game-based Learning*”, que pode ser traduzida literalmente como “aprendizagem baseada em jogos”. Como descrito na página da escola, todos os conteúdos são aprendidos através de jogos, eletrônicos ou não. A abordagem da escola se diferencia do modelo vigente não apenas na maneira como o conteúdo é ministrado, mas também em sua organização. Não existe distinção entre disciplinas, e os conteúdos são aprendidos de forma livre e interdisciplinar, remetendo aos princípios cognitivistas de aprendizagem (Q2L, 2018).

Os diferenciais desse modelo de ensino, como definido pela própria escola, são o desafio e feedback constante, que garantem ao estudante um senso claro do próprio desenvolvimento. As falhas também deixam de ser vistas como fracasso e passam a fazer parte do processo de aprendizado, tornando o ensino mais sólido. A escola afirma na própria página que seus estudantes alcançaram em 2015 um índice de proficiência 54% maior do que 30% da cidade de Nova York no exame ELA, que mensura o nível de aprendizagem comum dos estudantes nas principais disciplinas (Q2L, 2018).

Apesar de não descrever desta maneira, o aplicativo **Duolingo**, utiliza de elementos gamificados como diferencial para o ensino de idiomas. No aplicativo, os conteúdos são organizados em forma de “fases”, relacionadas à alguma estrutura do idioma que está sendo aprendido (verbos, substantivos, artigos). Quando o usuário erra uma questão, perde uma “vida”, e quando acerta ou conclui fases, é recompensado com “moedas”. A dificuldade aumenta gradativamente, e como o conhecimento é cumulativo, as estruturas mais simples são aprendidas inicialmente e constantemente reutilizadas ao longo das fases posteriores. A interface do aplicativo com seus elementos gamificados pode ser vista na figura 6.

**Figura 6**: Interface do Duolingo.



**Fonte**: Duolingo (2018).

O Duolingo utiliza em sua concepção princípios behavioristas e cognitivistas além do *gamification,* podendo ser classificada como uma solução gamificada, de acordo com o quadrante de Deterding. O aplicativo é uma grande inspiração e referencial para a criação do Virtual Table.

## DESENVOLVIMENTO DE SOLUÇÕES GAMIFICADAS

Segundo Alves (2015), a construção de soluções gamificadas exige que se pense em jogos pela perspectiva do *designer*, ao invés da perspectiva de jogador com a qual a maioria está acostumada. É necessário estudar os jogos de maneira analítica, entender o que faz dos mesmos especiais e buscar aplicar essas características na solução que está sendo desenvolvida. A recomendação da autora para encontrar a peculiaridade dos jogos é: Jogar! Nas palavras da autora: “analise o que estimula você, o que o motiva a continuar, as sensações que você tem durante as mudanças de fases ou quando você precisa enfrentar alguém”.

Conforme já mencionado nas seções anteriores, a gamificação sozinha não garante que os resultados desejados serão alcançados se a mesma não oferecer vantagens claras em relação à concorrência, e isso também pode ser considerado para o desenvolvimento de soluções de aprendizagem gamificadas. Segundo Alves (2015), o *gamification* nunca deve ser pensado como a principal estratégia instrucional, mas como um complemento da estratégia principal. A autora enfatiza ainda o fato de que deve ser escolhida a forma de gamificação que melhor se adapta ao propósito da solução.

Segundo Alves (2015 apud KAPP, 2014), soluções de aprendizagem gamificadas podem ser classificadas em 2 tipos básicos: ***gamification* estrutural** e ***gamification* de conteúdo**. O primeiro tipo se refere à aplicação de gamificação sem grandes alterações no conteúdo que está sendo ensinado. Adaptar o conteúdo ensinado para um sistema de *Quiz[[6]](#footnote-6)*, pode ser considerado gamificação estrutural. O Virtual Table se encaixa no segundo tipo, em que todo o conteúdo ensinado deve ser moldado para funcionar de forma análoga a um jogo.

### Motivação como elemento central

Na seção dedicada a descrever as características dos jogos, foi definido que um jogo é uma atividade descompromissada e voluntária. A liberdade é uma característica fundamental dos jogos, e por consequência da gamificação.

“As pessoas jogam, envolvem-se e dedicam seu tempo a esta atividade em busca de emoções positivas e diversão, desta forma a motivação é um ponto de extrema importância quando o assunto é o Gamification e aprendizagem (ALVES, 2015). ”

O jogador deve estar motivado a jogar, da mesma forma que o aprendiz deve estar motivado a aprender. O aprendiz pode ser interessado o suficiente no conteúdo ensinado à ponto de se auto motivar, da mesma forma que essa motivação pode ser resultado de influência externa. Segundo Alves (2015), existem dois tipos básicos de motivação: **intrínseca** e **extrínseca**.

A motivação **extrínseca** acontece quando agente externo influencia diretamente alguma decisão. Essas influências geralmente acontecem em alguma forma de recompensa, material (dinheiro, objetos, troféus) ou abstrata (serviços, reconhecimento, *status*). A pessoa é motivada a fazer algo em troca de alguma forma de pagamento, ou para evitar perder dinheiro. A motivação extrínseca tem uma abordagem muito próxima do behaviorismo, que desconsidera qualquer tipo de motivação **intrínseca**, pois a mesma possui uma natureza subjetiva e intuitiva. Quando motivado intrinsecamente, o indivíduo é motivado a fazer algo porque aquela ação oferece algum significado para ele. Citando o exemplo de Alves (2015), o altruísmo no ato de doar sangue é um exemplo de motivação intrínseca, que se extingue quando existe algum tipo de compensação, monetária ou não.

Segundo Alves (2015), um designer de soluções gamificadas deve utilizar os elementos de jogos disponíveis para encontrar um equilíbrio de motivadores externos e internos. Embora seja utilizado primariamente para guiar o desenvolvimento de jogos, a metodologia *M.D.A* pode também ser utilizada na criação de soluções gamificadas. Essa metodologia classifica os elementos de jogos em três categorias distintas e interdependentes (mecânicas, dinâmicas e estética), com o objetivo de facilitar a organização dos elementos de jogos para a criação do mesmo, e que também pode ser aplicado para o *Gamification*.

### *Framework* M.D.A para desenvolvimento de soluções gamificadas

O ***M.D.A*** (*mechanics, dynamics and aesthetics*), é um modelo formal de design de jogos que busca facilitar a comunicação entre programadores, artistas, designers de jogo e todos os profissionais envolvidos na criação de um jogo. Este modelo fragmenta todos os jogos em 3 partes essenciais: mecânicas, dinâmicas e estética. Uma vez que todos os envolvidos na criação conhecem as bases do jogo, o restante do processo é facilitado (ALVES, 2015).

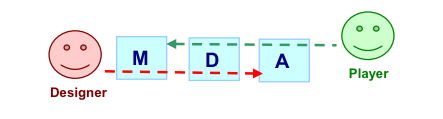
As **mecânicas** de um jogo podem ser compreendidas como a parte “invisível”, mas indispensável para a existência de um jogo: regras do jogo, limitações, algoritmos e todos sistemas que estabelecem o funcionamento daquele mundo fictício. Em jogos não-digitais, todas as estruturas de regras dos jogos devem ser compreendidas de forma clara pelos participantes, sendo às vezes necessário designar um participante para garantir que estão sendo cumpridas (os “juízes”). Em jogos digitais as mecânicas definem por exemplo a altura dos saltos de personagens em jogos de plataforma, e a maneira como os demais interagem com o jogador e o ambiente, além do que é permitido ou não fazer dentro do jogo. A vantagem dos jogos digitais é que toda a estrutura de regras já está programada, e essa programação que estabelecem as mecânicas do jogo (ALVES, 2015; MIGNANO, 2018).

Aliado ao poder computacional das plataformas de videogame atuais, as mecânicas possíveis de serem programadas dentro de um jogo são virtualmente infinitas. A maneira como o jogador interage com as mecânicas do ambiente, como utiliza os elementos disponíveis dentro das limitações impostas pelo jogo, e as estratégias que emergem disso para atingir os objetivos do jogo definem as **dinâmicas** do mesmo. O xadrez por exemplo, possui um conjunto de regras relativamente simples, mas milhões de dinâmicas possíveis, pois o movimento do adversário é imprevisível. A dinâmica pode ser resumida como o resultado da interação do jogador com as mecânicas e elementos não-controláveis de um jogo. A estética por sua vez, é desenvolvida à partir das dinâmicas do jogo (ALVES, 2015; MIGNANO, 2018).

A **estética** de um jogo é definida como a combinação de todos os elementos do mesmo que estabelecem interface com o jogador. Embora seja comumente associada ao aspecto visual, a estética deve ser entendida como toda a experiencia promovida por um jogo. Em termos mais precisos, a estética descreve as respostas emocionais que se deseja obter dos jogadores, como fantasia, desafio, exploração ou expressão. Tudo o que o jogador vê, ouve, sente e experimenta é estética. Geralmente jogos não focam em uma estética específica, mas numa combinação harmônica de várias (ALVES, 2015; MIGNANO, 2018).

A compreensão incorreta do que é a estética algumas vezes leva à classificação errônea de jogos. Devido ao senso comum de estética como visual, jogos de gêneros distintos são classificados na mesma categoria por exibirem semelhanças visuais como o estilo de câmera utilizada, o visual dos cenários, itens disponíveis e até mesmo mecânicas. Mas os mesmos elementos citados podem ser utilizados para criar um frenético jogo de ação ou um jogo de solução de puzzles, com ritmo mais calmo. Como mencionado por James (2012), é necessário considerar toda a estética de um jogo para uma classificação mais fiel.

**Figura 7**: Metodologia MDA aplicada.



**Fonte**: Marco (2018).

O desenvolvimento de jogos através do modelo *MDA* considera que os aspectos descritos anteriormente devem ser construídos de forma progressiva: a estética emerge das dinâmicas, que por sua vez emerge das mecânicas. O modelo tem como pressuposto básico que as mecânicas de um jogo devem ser pensadas incialmente, e a dinâmica proveniente dessas mecânicas definirá a estética final do jogo. Compreendendo estes conceitos, os envolvidos na produção de um jogo terão um senso de como cada mudança impactará no resultado final e na experiência do jogador. No *MDA*, o ponto de vista dos desenvolvedores é diretamente oposto ao dos jogadores, que experimentam as dinâmicas de um jogo através da estética e posteriormente deduzem as dinâmicas e mecânicas, ilustrado na figura 7. De acordo com Marco (2018), apesar da ordem definida pelo *MDA*, ambos os pontos de vista devem ser considerados na criação de um jogo.

### Fundamentos do *game design*

Videogames revolucionaram a maneira como enxergamos os jogos, oferecendo possibilidades nunca imaginadas. Se antes a “realidade paralela” do círculo mágico era conceitual, os videogames tornaram possível a materialização desse conceito em mundos virtuais completamente autônomos. O desenvolvimento da internet também possibilitou que pessoas de qualquer parte do mundo participem da mesma partida, transportando os limites físicos do jogo para o mundo virtual. A principal diferença entre videogames e jogos tradicionais está na interação o jogador e o jogo. Uma vez que a maior parte da lógica, regras e estética do jogo está programada no software do jogo, os únicos elementos físicos são os dispositivos que fazem a interface com o jogador, como mouses, teclados, joysticks, televisores, sensores de movimento, e recentemente, os capacetes de realidade virtual.

O nascimento da indústria dos videogames pode ser considerado como o boom dos fliperamas[[7]](#footnote-7) na década de 70, com o sucesso da Atari. Como cita Ryan (2009), o mercado estava saturado de fliperamas, havendo máquinas instaladas até mesmo em supermercados, farmácias e padarias. Conceitos como game design e experiência de usuário não existiam ainda, mas a indústria havia descoberto o poder sedutor das moedas cintilantes numa tela de fósforo. Desde o início já foi possível notar como os jogos tem a capacidade de manter o interesse em um nível viciante, especialmente videogames.

O jogo *Pong* (Figura 8), é considerado oficialmente como o primeiro videogame. Nas últimas décadas, os videogames evoluíram paralelamente ao poder computacional das suas plataformas, passando pelas pesadas máquinas de fliperama até os modernos consoles de mesa. Esse quase meio século de desenvolvimento adaptou os elementos de jogos para as novas tecnologias e amadureceu as técnicas de desenvolvimento de videogames. A recente área de estudos chamada de game design combina as possibilidades da computação com a pesquisa já existente sobre jogos o objetivo de oferecer a melhor de experiência para os jogadores (BELLO, 2017).

**Figura 8**: Pong, o primeiro videogame.



**Fonte**: Knott (2017).

Com o progresso da tecnologia, o poder computacional disponível para os criadores de jogos aumentou exponencialmente, permitindo mecânicas mais complexas e visuais mais realistas. Consequentemente, foi necessário entender de maneira detalhada a relação dos jogadores com os videogames, e como fazer jogos melhores. Ao longo dos anos, com pesquisas e experiência de desenvolvedores, conceitos próprios do desenvolvimento de jogos como *level design* e *pacing* amadureceram dentro da comunidade de desenvolvedores e deram origem ao game design (BELLO, 2017).

**Figura 9:** Donkey Kong, 1980.



**Fonte**: Ryan (2012).

O primeiro jogo a utilizar os modernos conceitos de game design, talvez de forma não intencional, foi o primeiro *Donkey Kong*, criado e distribuído pela Nintendo ainda no auge da era dos fliperamas. Nele o personagem principal deve salvar a princesa de um macaco gigante, que atira barris contra o personagem principal do topo da tela, onde mantém cativa a princesa. O personagem deve chegar até a princesa, desviando ou destruindo os barris utilizando itens espalhados estrategicamente pelo cenário. A direção única (sempre para o topo da tela) que o personagem se movia, dava ao jogador um senso claro de onde ele deveria chegar. Ações desejadas como desviar ou destruir barris eram recompensadas com “moedas” (uma das formas de pontuação do jogo). A história simples oferecia um contexto lúdico para as ações do personagem, potencializado pela trilha sonora dinâmica, que era calma na maior parte do jogo, mas adotava um tom mais frenético quando o personagem estava habilitado a destruir barris (RYAN, 2009).

Apesar de serem elementos comuns aos jogos atualmente, nenhum jogo na época possuía algo semelhante. O primeiro *Donkey Kong* foi um marco na indústria, estabelecendo as bases do game design para os jogos posteriores e consequentemente para o *gamification*. A figura 8 exibe a tela de uma das fases do jogo. Uma dessas bases, utilizados até hoje no desenvolvimento de jogos, é o conceito behaviorista de **recompensas fixas** e **recompensas variáveis**. A mecânica de intervalos fixos garante que o jogador sempre encontrará uma recompensa já prevista ao realizar determinada ação (receber uma quantidade fixa da moeda fictícia do jogo ao eliminar um monstro).

Já em relação às recompensas variáveis, o jogo não deixa claro quais ações levarão à determinado tipo de recompensa, nem a quantidade ou valor desta. As recompensas variáveis tem maior potencial de engajamento devido à incerteza gerada: o jogador continua jogando, pois sempre existe a chance de ganhar mais do que a última vez, como em um caça-níquel. Mas os *designers* devem ser cuidadosos na escolha destas mecânicas: se houverem recompensas fixas em excesso, o jogo se tornará previsível, mas recompensas variáveis em excesso tornará o jogo demasiado incerto e independente da habilidade do jogador, afastando os jogadores iniciais. O correto balanceamento do sistema de recompensas deve ser considerado também na criação de soluções gamificadas (ALVES, 2015).

Outro elemento básico de game design é a clareza dos objetivos. O jogador precisa saber de maneira intuitiva quais são as metas do jogo, e o que deve ser feito para alcançá-las. No jogo *Donkey Kong*, a princesa era sempre posicionada no topo da tela, e o jogador começava o nível na parte inferior da tela. Isso fornecia aos jogadores apenas um caminho a ser seguido: para cima. O movimento unidirecional do personagem garantia ainda outro elemento básico do *game design*: o senso de progresso. A sensação de que o jogador está “avançando”, é essencial em qualquer jogo, da mesma maneira em soluções gamificadas.

# MATERIAIS E MÉTODOS

Este capítulo tem como objetivo apresentar os métodos de pesquisa e desenvolvimento utilizados para o desenvolvimento da aplicação que é o foco deste trabalho. Devido à natureza do trabalho em questão, não foram utilizadas técnicas estatísticas para aferição de resultados, sendo a pesquisa focada integralmente no referencial teórico e técnico disponível. Sendo um trabalho de desenvolvimento, o mesmo foi construído baseado em tecnologias já existentes. Inicialmente serão suscintamente apresentadas as tecnologias utilizadas, como estas funcionam e a razão da utilização destas tecnologias. Posteriormente será apresentado o processo de criação da ferramenta e a aplicação dos conceitos apresentados na fundamentação teórica, justificando as escolhas tomadas ao longo do processo de desenvolvimento.

## METODOLOGIA DE PESQUISA

O objeto de estudo deste trabalho foi a análise e desenvolvimento de um protótipo de uma ferramenta de aprendizagem aplicando o *gamification*. Para entender de forma aprofundada o objeto de estudo, foi necessário buscar o material disponível tanto na literatura quanto de outras fontes não-acadêmicas. Considerando essa metodologia, esta pesquisa pode ser considerada de natureza indutiva, pois segundo Lakatos e Marconi:

"Indução é um processo mental por intermédio do qual, partindo de dados particulares, suficientemente constatados, infere-se uma verdade geral ou universal, não contidas nas partes examinadas. Portanto, o objetivo dos argumentos indutivos é levar a conclusões cujo conteúdo é muito mais amplo do que o das premissas nas quais se basearam. (LAKATOS e MARCONI, 2003)"

O método indutivo tem como pressuposto de que o caminho de passagem vai do especial ao mais geral, e desta forma ocorreu a pesquisa bibliográfica deste trabalho. Conhecia-se inicialmente o potencial do *gamification* na educação, porém não de forma detalhada. Ao aprofundar a pesquisa sobre o gamification foi possível alcançar objetos de estudo mais detalhados que não estavam previstos, mas que eram úteis para o desnvolvimento do trabalho como um todo. A gamificação tem uma natureza próxima dos jogos, e compreender o conceito do círculo mágico foi de grande valia para a compreensão do próprio gamification. Por ser uma ferramenta digital, a relação com vídeo-games era inevitável, o que consequentemente encaminhou para o estudo das técnicas de desenvolvimento de jogos digitais (LAKATOS e MARCONI, 2003).

A pesquisa seguiu a premissa do método indutivo, em que primeiro são observados os fenômenos, para que seja averiguado se os mesmos estão de alguma forma relacionados e posteriormente seja elaborada uma generalização (LAKATOS e MARCONI, 2003). Inicialmente, através de observações e principalmente da vivência, foi possível constatar uma relação entre o ato de aprender e jogar. Aprofundando a pesquisa destes dois objetos de estudos distintos, posteriormente foi possível constatar a relação entre os princípios cognitivistas e behavioristas com as características fundamentais dos jogos. À partir dessa relação foi possível generalizar os conceitos necessários para o desenvolvimento do Virtual Table e guiar o restante da pesquisa e desenvolvimento deste trabalho.

### Técnica de pesquisa

Por ser objeto de estudo desde os primórdios da humanidade, o processo de aprendizagem já vem sendo estudado à bastante tempo, sendo as teorias da aprendizagem conceitos relativamente bem estabelecidos. Por essa razão existe uma grande disponibilidade de literatura a respeito deste tema, e consequentemente uma grande disponibilidade de pesquisas e trabalhos desenvolvidos com foco nestas teorias.

As teorias da aprendizagem não eram o foco de estudo deste trabalho, mas era necessário compreender a maneira como cada uma delas entendia o aprendizado, para buscar a existência de alguma relação com jogos e consequentemente com o *gamification*. Para estabelecer estas relações foram utilizadas pesquisas e trabalhos desenvolvidos por outros autores, reduzindo as chances de retrabalho na coleta de informações, além da descoberta de informações que talvez passariam despercebidos caso fosse o material fosse analisado na fonte. Outro fator relevante para utilizar outras pesquisas ao invés da literatura fonte, é o fato de que as mesmas possuem uma pesada carga teórica, sendo alguns conteúdos irrevelantes para este trabalho.

Como mencionado no segundo capítulo, o conceito de *gamification* possui menos de duas décadas de existência, razão pela qual não existe muita literatura nem pesquisas desenvolvidas a respeito do mesmo. Por essa razão, foram os utilizadas as mais variadas fontes de conteúdo disponível à respeito do mesmo: livros, artigos acadêmicos, trabalhos de graduação, artigos disponíveis na internet, vídeo-aulas e palestras. Ao contrário das teorias da aprendizagem, no caso do gamification foi necessário consultar o material fonte algumas vezes, pois o conteúdo utilizado se apresentava incompleto ou insuficiente.

Considerando as técnicas de pesquisa utilizadas e de Acordo com Lakatos e Marconi (2003), a pesquisa do referencial teórico pode ser definida como uma pesquisa bibliográfica, pois toda a coleta de dados foi feita através de fontes secundárias. Não houve uma análise direta de documentos em busca de informações, e as informações utilizadas como base tiveram origem em outras pesquisas.

## MATERIAIS

### Tecnologias utilizadas

Por ser baseada em tecnologias web, a ferramenta foi desenvolvida para ser utilizada diretamente de qualquer navegador, sem a necessidade de downloads e instalações. As razões para a escolha desta tecnologia como plataforma da aplicação foram principalmente técnicas, mas também por questões de facilidade de uso para o usuário final. O amadurecimento das tecnologias web como plataforma de desenvolvimento de aplicações interativas e a disponibilidade de *frameworks* específicos para o desenvolvimento de games pouparam muito tempo de desenvolvimento e foram as principais razões para escolha desta tecnologia. Qual framework foi utilizado e os conceitos mencionados anteriormente serão melhor explanados posteriormente. Considera-se também o fato de que a maioria dos usuários comuns, que são o foco desta ferramenta já estão habituados à utilização de navegadores web e a ausência de procedimentos mais complexos de instalação e download facilitam o caminho para a utilização da ferramenta e tornam o processo mais amigável ao usuário final.

### Tecnologias Web

Todo o conteúdo web é executado e exibido através de browsers. Também chamados de navegadores em português, é um jargão técnico para designar aplicações computacionais utilizados para execução de conteúdo para a internet. A base para o desenvolvimento é o HTML, sigla para *HyperText Markup Language*, que em português, significa linguagem para marcação de hipertexto. “Hipertexto” é relativo à capacidade dos textos na internet de se associarem a outros textos na web através dos links, sendo as imagens, sons e vídeos considerados como hipermídia (SILVA, 2015; FLATSCHART, 2011).

Chamado simplesmente de HTML quando criado em 1992, passou por grandes modificações ao longo dos anos, sendo a versão mais atual a 5.2. No início, o conteúdo era formado basicamente por texto, sendo posteriormente incorporadas imagens e estilos de formatação. A adição dessas novas características expandiu as possibilidades do HTML ao mesmo tempo em que tornavam o processo de produção dos conteúdos mais trabalhosos, pois não havia distinção entre o que era realmente elemento de conteúdo, formatação ou interatividade da página. Com o objetivo de criar acessíveis e operáveis em qualquer navegador, foi criado em 1994 o W3C (*World Wide Web Consortium*), pelo mesmo idealizador do HTML, Tim Berners-Lee (EIS, 2012; FLATSCHART, 2011).

Os padrões definidos pela W3C separam o conteúdo desenvolvido para em 3 camadas distintas e independentes, mas que funcionam plenamente apenas em conjunto. A primeira camada é a de informação: todo o conteúdo que é consumido pelo usuário deve ser identificado através das marcações do HTML (parágrafos, títulos, textos, imagens). A segunda camada é a de formatação, responsável por definir o visual da informação e como essa vai ser apresentada de acordo com o dispositivo utilizado. Essa camada é controlada pelo CSS[[8]](#footnote-8), outra linguagem de marcação. A última camada é a de comportamento e interatividade, e que torna possível a existência do VirtualTable. Essa camada é controlada atualmente pelo JavaScript, uma linguagem de script que possibilita a manipulação de elementos HTML e estilos CSS (EIS, 2012).

### JavaScript

Diferente das linguagens de marcação citadas anteriormente, o *Javascript* possui uma estrutura sintática semelhante às linguagens de programação mais conhecidas, como C e Java, que utilizam variáveis, condicionais e loops. Apesar do nome, não existe relação com a última, sendo uma convenção adotada pelos desenvolvedores devido às semelhanças sintáticas. É oficialmente nomeada de *ECMAScript*, e atualmente se encontra na versão 5.0 (FLANAGAN, 2006).

Como mencionado anteriormente, a principal característica do Javascript é tornar dinâmico o conteúdo primariamente estático do HTML e CSS. Utilizando o mesmo é possível alterar a forma e conteúdo dos elementos de uma página web e realizar funções inexistentes no HTML puro, como cálculos e processamento de informações. Além destas características, o que tornou possível o desenvolvimento do VirtualTable é a capacidade do *Javascript* de adicionar interatividade ao conteúdo, programando comportamentos e ações que respondem à ações específicas do usuário (FLANAGAN, 2006).

Por ter a estrutura sintática semelhante à de linguagens de programação, o Javascript permite a reutilização de funções já programadas. Aplicações com propósitos semelhantes geralmente utilizam as mesmas funções, e consequentemente, acabam utilizando scripts semelhantes. Visando reduzir o tempo de desenvolvimento, existem os *frameworks*, que são basicamente coleções de scripts para ações mais comuns em um determinado tipo de aplicação. Para o desenvolvimento deste trabalho foi utilizado o Phaser, um framework livre que reúne funções que visam facilitar o desenvolvimento de *games* para *browsers*.

### Phaser Framework

*Phaser* é um *framework* *open source* *Javascript* voltado para o desenvolvimento de jogos que funcionem em navegadores. O framework foi desenvolvido pela empresa *Foton Storm*, que antes da criação do Phaser já havia trabalhado com desenvolvido de jogos e colaborou na criação de outro framework para games, o *Flix*. A escassez de frameworks para criação de jogos nos anos iniciais e as experiencias anteriores influenciaram na criação do Phaser, que foi concebido com o objetivo de ser uma ferramenta de auxílio para a criação de outros jogos, automatizando rotinas de códigos comumente utilizadas e permitindo que a equipe de desenvolvimento foque no design de interação (DAVEY, 2018).

Por ser uma biblioteca *Javascript*, é possível a execução de aplicações baseadas no mesmo em qualquer navegador com suporte à HTML5, incluindo navegadores mobile. A criação de aplicações nativas para desktop também é possível através da utilização de ferramentas como *CocoonJs* e *Ejecta*.

Com o amadurecimento do HTML5 como plataforma de desenvolvimento, foram criados outros *frameworks* para games em paralelo ao *Phaser*, como o *babylon.js* e *Three.js*. Os principais recursos que destacam o Phaser entre os outros frameworks disponíveis, segundo seus criadores, é o fato de ser uma biblioteca Javascript pura e funções para o carregamento dos *assets[[9]](#footnote-9)*. Podem ser citados também como particularidades do mesmo o suporte do mesmo para arquivos de áudio e a existência de funções para animações e interação física entre objetos de um jogo, que poupa muito trabalho dos desenvolvedores.

A biblioteca foi disponibilizada à público em 2013, havendo atualizações constantes durante os últimos 5 anos. O mesmo possui uma comunidade bastante ativa, sendo esta a principal razão para a escolha deste *framework*. É necessário para o desenvolvimento deste projeto ferramentas robustas, e o *know-how* adquirido na utilização da mesma é de grande valia. As características do *Phaser* citadas anteriormente e a grande quantidade de material para estudo também influenciaram na escolha do mesmo (DAVEY, 2018).

## DESENVOLVIMENTO DO VIRTUAL TABLE

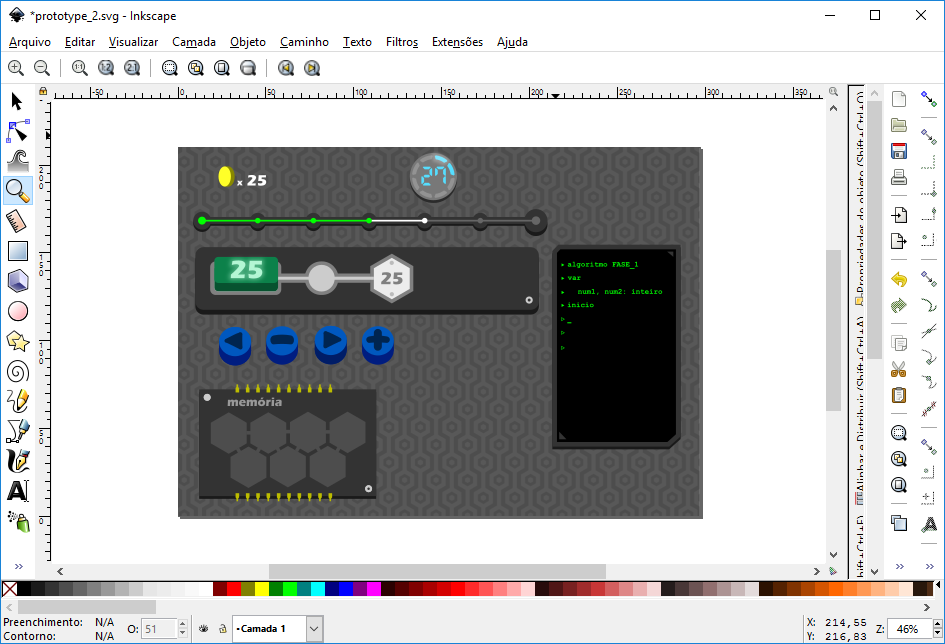
### Ferramentas de desenvolvimento utilizadas

Como ferramentas de desenvolvimento foram utilizados o IDE *Atom* em conjunto com o software XAMMP para a criação de um servidor web local onde eram realizados a depuração dos códigos escritos e os testes de usabilidade. Para a criação dos elementos gráficos da ferramenta foi utilizado o software *Inkscape*, sendo outros elementos retirados de repositórios disponíveis na internet. Deve-se destacar o fato que todos as ferramentas utilizadas são Open Source, sendo uma das principais razões para a escolha das mesmas. Entre outras razões pode-se a facilidade de configuração do XAMMP e a familiaridade de uso do *Inkscape* e *Atom*.

### Fase de desenvolvimento gráfico

Todos os elementos gráficos do Virtual Table foram desenvolvidos com o software livre *Inkscape*, com exceção dos elementos que foram retirados de repositórios de arquivos públicos. A figura 10 ilustra a interface gráfica do *Inkscape* durante o processo de desenvolvimento do protótipo.

**Figura 10**: Interface do Inkscape.



**Fonte**: Autoria própria (2018).

O *Phaser* utiliza a técnica de *sprites* para elementos que possuam qualquer tipo de animação. A utilização de *Sprites* é uma técnica em que todos os *frames* de uma animação estão no mesmo arquivo de imagem, de forma sequencial. Quando este arquivo é visualizado fora de ambiente de execução, se assemelha à um rolo de filme, como na figura 11. Nesta figura é possível ver o *Sprite* de uma das animações utilizadas no Virtual Table.

**Figura 11**: Exemplo de Sprite.



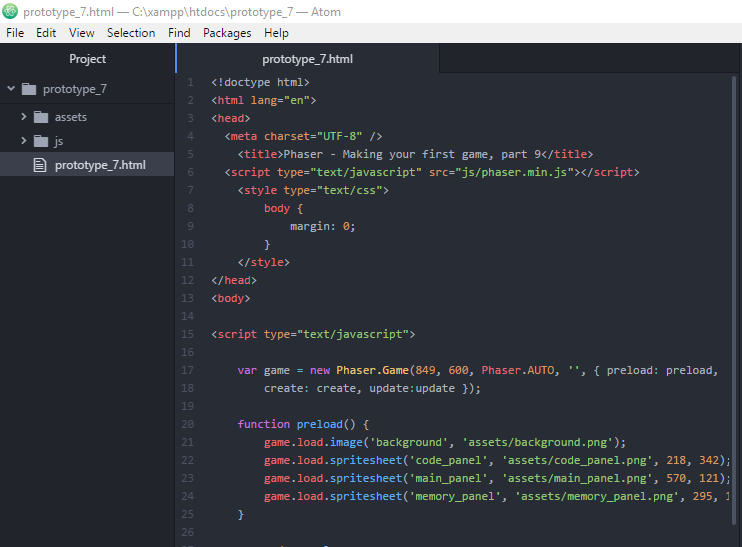
**Fonte**: Autoria própria (2018).

Como praticamente todos os elementos visuais dentro do protótipo possuem algum tipo de interação, foi necessário criar os *Sprites* das animações de todos eles. Utilizando esta técnica foi possível criar animações personalizadas de acordo com interação que ocorreu, além de ser também possível agrupar em um único arquivo todos os elementos que são muito semelhantes, como os botões que serão melhor exemplificados no terceiro capítulo.

### Fase de codificação

Para a codificação do protótipo foi utilizado a IDE livre Atom. Para execução e teste do protótipo foi utilizado o navegador web Google Chrome, e o ambiente para a criação de um servidor web local foi feita através do também software XAMMP.

**Figura 12**: Interface do Atom

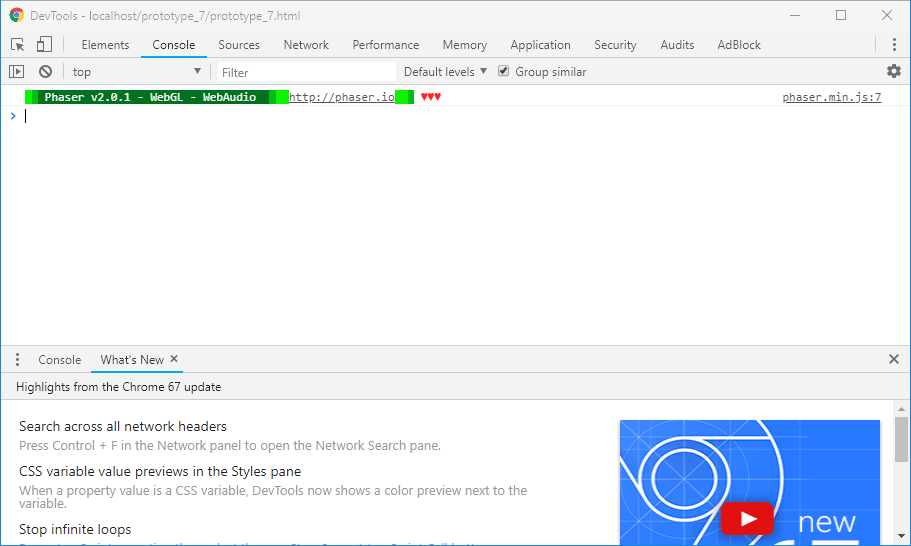


**Fonte**: Autoria prórpia (2018)

O protótipo é inteiramente executado em ambiente web, através do Java Script. Os arquivos de extensão .js, contém os códigos relativos à cada parte do projeto, e são chamados pelo script principal, contido dentro de um arquivo .html. Na figura 12 é possível ver a interface do software Atom durante o processo de codificação do Virtual Table.

Sendo o Phaser um framework específico para a criação de games, o mesmo possui diversas funções próprias para este tipo de aplicação. Grande parte da coficação do protótipo envolveu instânciar e informar os parâmetros para as funções Phaser, como para a criação de *sprites,* habilitar entrada de dados do usuário e tratar, entre outras funções. Quando se fez necessário movimentar um animar um objeto, é preciso informar os parâmetros corretos para a função correta do framework.

**Figura 13**: Interface da ferramenta de desenvolvimento do chrome



**Fonte**: Autoria própria (2018).

Como este protótipo ainda não comtempla gravação de dados, não foram criadas tabelas para esse fim. Quase toda a codificação foi baseada em reutilização de código do Phaser, e para testes foi utilizada a função avançada de desenvolvendo o Google Chrome, inpecionando as mensagens de erro retornados pelo própro Feamework. A figura 13 ilustra a interface dessa ferramenta utilizada na devolução, no caso em específicos não houveram erros.

# RESULTADOS E DISCUSSÃO

A ferramenta foi desenvolvida baseada nos princípios de gamificação estabelecidas no segundo capítulo deste trabalho. Por ser uma ferramenta digital, a gamificação foi aplicada em conjunto com os fundamentos de game design também descritos no mesmo capítulo. Esta seção tratará incialmente na apresentação da ferramenta, e como a pesquisa foi aplicada para o desenvolvimento da mesma. Posteriormente será explicada a maneira como o framework MDA foi utilizado para a para guiar o desenvolvimento da ferramenta. Os aspectos técnicos do desenvolvimento, como o funcionamento do framework Phaser, esclarecimento de conceitos que são específicos de desenvolvimento de jogos como *sprites* e já foram abordados no terceiro capítulo. Será tratado neste capítulo também os aspectos gamificados da ferramenta, e como os conceitos anteriormente estudados foram aplicados com o objetivo melhorar o aprendizado de lógica de programação.

## APRESENTAÇÃO DO VIRTUAL TABLE

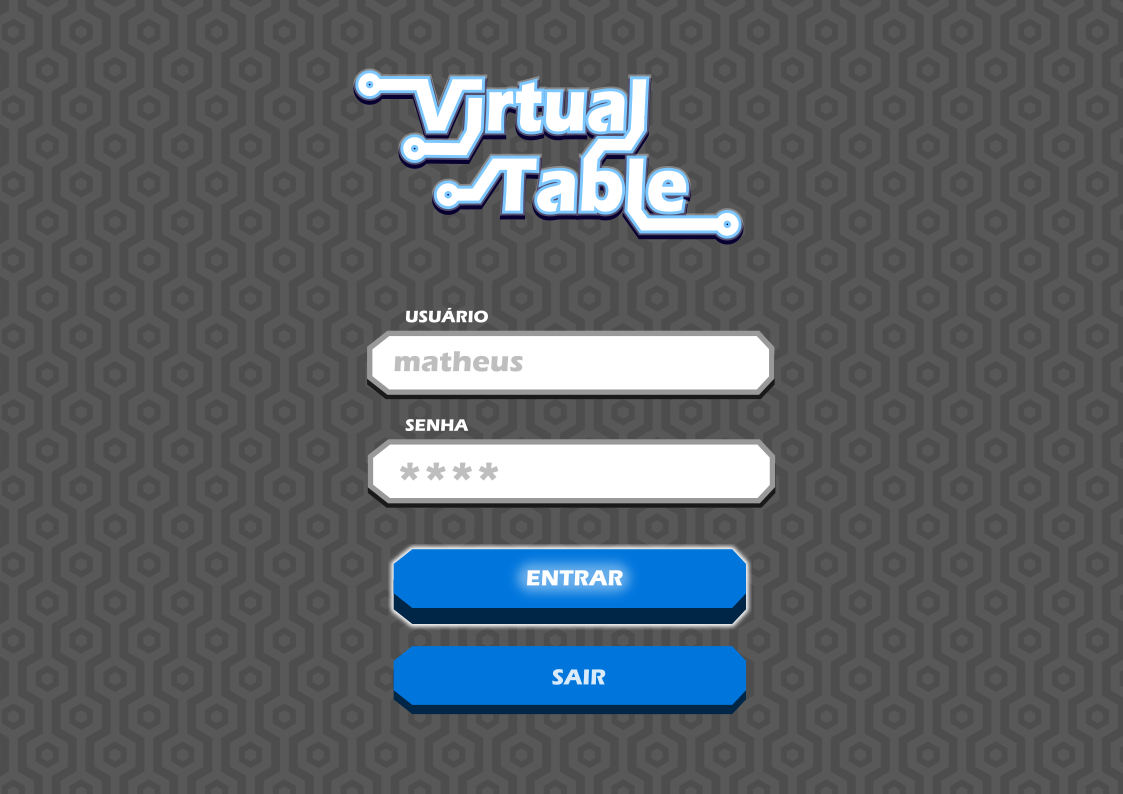
A primeira tela do Virtual Table apresentada ao usuário possui apenas as opções para entrar com um usuário já cadastrado, cadastrar um novo usuário ou exibir informações sobre a própria ferramenta, mostrado na figura 14. Ao escolher cadastrar um novo usuário, é necessário apenas que seja definido uma identificação, um nome e uma senha, como na figura 15. Após cadastrado, a ferramenta já pode ser utilizada.

**Figura 14:** Tela inicial do Virtual Table.



**Fonte**: Autoria própria (2018).

**Figura 15:** Tela de cadastro do Virtual Table.



**Fonte**: Autoria própria (2018).

Seguindo a maneira informal como os níveis de jogos em geral são chamadas, os níveis individuais serão chamados de “fases”, e as seções internas de cada fase serão chamadas de “*Quests*”. Como mencionado anteriormente, os conceitos de lógica são tratados de forma progressiva semelhante aos níveis de um jogo. Variáveis são os primeiros conceitos à serem aprendidos, por serem a base da programação e dos outros conceitos como condicionais e laços de repetição. Na figura 16 está ilustrada a tela em que são mostradas as “fases” correspondentes aos conceitos de lógica de programação (variáveis, condicionais, laços e funções).

**Figura 16:** Tela de seleção de níveis do Virtual Table.



**Fonte**: Autoria propria (2018).

A mecânica de gamificada de barra de progresso já é utilizada na tela de seleção das “fases”: há uma barra indicando o progresso do usuário naquele determinado conceito. Seguindo a corrente cognitivista de aprendizado progressivo, as “fases” posteriores serão habilitadas apenas quando a anterior for concluída. É necessário que o conceito de variável como um “contâiner” de informação esteja bem estabelecido antes de tentar compreender o conceito de condicionais. O aluno precisa “acomodar” o esquema Piagetiano de variável antes de todos os outros.

**Figura 17**: Seleção de sub-níveis do Virtual Table.



**Fonte**: Autoria própria (2018).

Na figura 17 está ilustrada a tela de seleção das “*Quests”* de cada uma das “fases” correspondentes aos conceitos estudados. As “*Quests”* também são organizadas de forma progressiva, onde o nível de dificuldade e desafio aumenta gradativamente.

### Funcionamento e mecânicas básicas do Virtual Table

Nesta primeira fase, o objetivo é fazer com que o usuário assimile o funcionamento de uma variável – um espaço na memória que armazena um valor – com a maneira que este deve ser escrito em portugol. A primeira “Quest”, é um simples algoritmo que cria duas variáveis e atribui valores às mesmas: um valor é atribuído à primeira variável, que depois é passado desta para a segunda variável. No final da fase, o algoritmo escrito em portugol será esse:

Algoritmo fase1;

Variável

num1, num2: inteiro;

Inicio

25 -> num1;

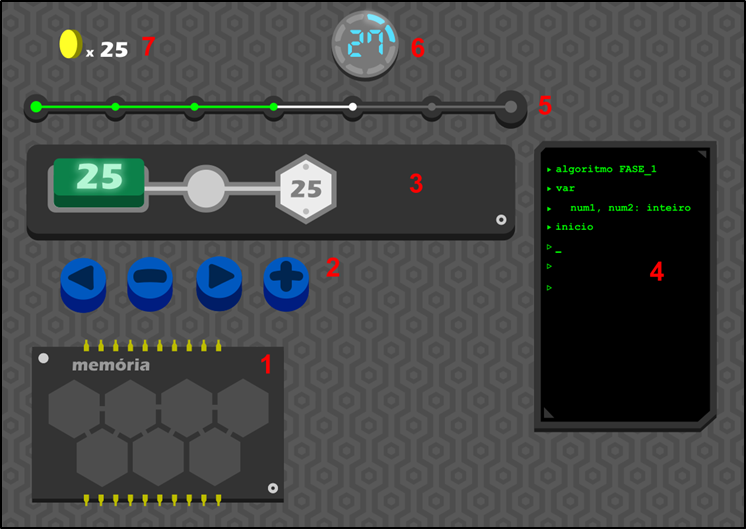
num1 -> num2;

Fim

Quando o usuário clica para executar a primeira *Quest* da primeira fase, a tela exibida é ilustrada na figura 9. Na figura estão indicadas as seções disponíveis: na primeira está a representação visual da memória, onde são disponibilizadas as variáveis criadas. Na segunda estão os objetos referentes aos operadores. Esses são elementos interativos que devem ser “encaixados” e “montados” nos espaços disponíveis na seção 3. As ações de organizar os objetos dos operadores são instantaneamente refletidas no quadro da seção 4, que exibe o algoritmo em portugol que está sendo escrito, que também reflete na barra na seção 5, exibindo o progresso da “fase”. Por fim, a seção 6 exibe um “timer”, que delimita o tempo para concluir a “fase”, e a seção 7 exibe as recompensas obtidas durante o decorrer da “fase”.

Apesar de mudar a estrutura conforme os níveis se tornam mais complexos, cada “fase” é basicamente um algoritmo, que aumenta de complexidade e dificuldade progressivamente. Cada linha de código do algoritmo é um “passo” individual da “fase”. Dessa maneira, os objetos disponíveis para manipulação serão os valores, operadores e posteriormente as variáveis. À cada “passo” concluído são adicionados novos objetos para utilização. Os passos também possuem seus objetivos individuais, e na figura 10 é indicado sutilmente no número ‘25’, dentro do espaço dedicado à variável.

**Figura 18**: Tela inicial do Virtual Table.



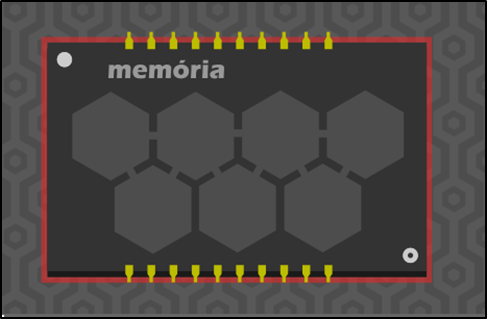
**Fonte**: Autoria própria (2018).

Neste primeiro passo o usuário deve mover e “encaixar” os objetos disponíveis na seção de algoritmo. Considerando a liberdade como aspecto primordial dos jogos e consequentemente do *gamification*, e também da boa prática de *Game Design*, são disponibilizados 4 objetos de diferentes para escolha do usuário. Apenas um deles cumpre o objetivo, ficando à critério do jogador explorar as possibilidades. Nesta primeira “fase” também não há indicadores verbais do que deve ser feito, ficando à cargo do usuário intuir sobre as ações que devem e podem ser realizadas através da tentativa e erro. Os elementos à disposição e a maneira como sua manipulação reflete no código escrito em tempo real serão alterados de acordo com a necessidade e o conceito de programação que está sendo utilizado, mas a mecânica central é a interatividade com os objetos disponíveis.

### Interatividade como elemento central

O Virtual Table tem como premissa básica que todos os elementos disponíveis em tela devem possuir algum nível de interatividade. Utilizando como exemplo a primeira “fase”, ao movimentar o cursor do mouse sobre os elementos dispostos (tela de código, barra de progresso, operadores, memória), estes são contornados ou com a cor vermelha, indicando que não podem ser movidos (Figura 19), ou com a cor verde, indicando que podem ser movidos (Figura 20).

**Figura 19:** Contorno indicando que um objeto não pode ser movido.



**Fonte**: Autoria própria (2018).

**Figura 20**: Contorno indicando que um objeto pode ser movido.



**Fonte**: Autoria própria (2018).

A disposição dos elementos que representam os operadores no painel trabalha de forma a mimetizar o funcionamento de um circuito elétrico. No caso da primeira “fase”, quando o operador é selecionado movido para o painel, ele se torna um botão acionável quando posicionado. O acionamento desse botão fecha o “circuito elétrico” fictício, “energizando” a variável. A primeira mecânica aprendida pelo usuário do Virtual Table é que um “passo” da fase só é concluído quando todos os elementos do painel estão “energizados” (representado pelo brilho exibido pelos elementos energizados). Quando um passo um concluído, os elementos anteriores são removidos e são adicionados novos, assim como o painel de organização que exibe outro padrão de preenchimento. O painel de código passa então a escrever a próxima linha, e o algoritmo vai sendo construído.

## *GAMIFICATION* NO VIRTUAL TABLE

### Direcionando ações através do *Gamification*

Como mencionado no segundo capítulo, a gamificação pode ser utilizada para guiar o usuário a realizar uma ação desejada, no caso do Virtual Table, compreender um conceito de lógica de programação e aplicá-lo para chegar a uma solução. Uma das mecânicas para obter engajamento utilizado pela gamificação, muito utilizado pelos jogos, é recompensar uma ação desejada e punir uma ação não desejada. Esta técnica tem o objetivo de obter um engajamento rápido de maneira superficial, reforçando o aspecto behaviorista desta mecânica.

Este conceito é aplicado logo no primeiro contato do usuário com a ferramenta. Na primeira fase, caso o usuário utilize o operador errado, assim que acionar o botão para energizar o circuito, o mesmo se “quebra”, e não poderá mais ser utilizado no jogo. No entanto, quando utilizar o operador correto, ao acionar o circuito, receberá um feedback visual positivo (os elementos energizados se iluminam), além da pontuação. Dessa forma, a ferramenta direciona o usuário para a escolha do usuário através das recompensas e punições. Mesmo que erre inicialmente, os outros operadores ainda estarão disponíveis para serem testados à critério do usuário.

Assim como no game design e em soluções gamificadas, as recompensas e punições devem ser corretamente balanceadas para se manter o usuário engajado. Nas primeiras “fases”, as punições não oferecem grandes perdas ao progresso do jogo, ajudando na verdade a guiar para a escolha correta. As recompensas por sua vez são propositalmente bastante visíveis, mas com pouca variação. As “Quests” iniciais de cada fase tem por padrão o objetivo de ensinar o usuário sobre o conceito que está sendo trabalhado de forma não expositiva, utilizando as próprias mecânicas disponíveis, sendo esse o motivo das punições não oferecerem grandes perdas.

No entanto, nas “*Quests*” subsequentes o foco é motivar o usuário através do desafio, considerando que os principais conceitos e mecânicas já foram trabalhados nas “*Quests*” anteriores. As estruturas a serem montadas se tornam mais complexas e são disponibilizados uma maior quantidade de objetos a serem utilizados. Devido a isso, as consequências da utilização incorreta de operadores se tornam mais imprevisíveis, forçando o usuário a pensar de forma estratégica. As recompensas também se tornam mais imprevisíveis, sendo maiores quando utilizadas combinações mais otimizadas e em menor tempo. Tudo isso tem como intenção contribuir para a construção do aprendizado, pois motiva o usuário a buscar a melhor solução dentro das limitações, consolidando conhecimentos adquiridos anteriormente ao utilizá-los para atingir outro objetivo.

Reiterando que o objetivo do Virtual Table é ensinar lógica de programação, as escolhas do usuário são exibidas de forma dinâmica em sua forma de código portugol. Apesar de indisponível nas “*Quests*” iniciais, outra mecânica disponibilizada é a possibilidade de se escrever código no painel de código e ver a ação ser refletida na disponibilização de novos objetos. Dessa maneira o conhecimento assimilado na organização dos objetos é relacionado com o Portugol, que poderá ser utilizado fora da ferramenta.

### Trabalhando a motivação intrínseca e extrínseca

Na seção anterior foi explicada a maneira como as recompensas e punições são utilizadas dentro do Virtual Table. Como descrito no segundo capítulo, a natureza condicionante de recompensas externas como pontos, moedas, insígnias e placares possuem um engajamento rápido, porém superficial. Por esse motivo tanto jogos quanto soluções gamificadas não podem se sustentar apenas motivando o usuário com recompensas, é necessário também motivar os mesmos de formas intrínseca, oferecer algo que se de alguma forma conecte se conecte com o participante, recorrendo mais uma vez à analogia do círculo mágico. É necessário um balanceamento adequado entre motivação intrínseca e extrínseca.

A motivação extrínseca dos usuários do Virtual Table é trabalhada principalmente nas recompensas e punições descritas nas seções anteriores. À cada fase é necessário o balanceamento correto entre o nível das punições e recompensas oferecidas, de modo a incentivar o usuário a continuar tentando. A motivação intrínseca por sua vez é trabalhada principalmente através dos desafios. Nos mapas de cada fase, os desafios de cada “Quest” são exibidos dentro de um contexto, de formas semelhantes a charadas, contribuindo para o aspecto lúdico. A complexidade progressiva de cada fase também é uma maneira de desafiar e incentivar os usuários. Essa mesma complexidade torna as consequências das ações mais imprevisíveis que aliado às recompensas variáveis e as limitações de tempo, criam a tensão adequada para manter o usuário interessado.

A motivação é o aspecto central da gamificação, por isso os elementos de jogos cumprem um papel fundamental na criação de soluções gamificadas. Utilizando os conceitos estudados no segundo capítulo deste trabalho, foram utilizados elementos de games, principalmente de games digitais, para engajar o participante em uma atividade e auxiliar o mesmo a atingir um objetivo, que não é a diversão em si, mas a assimilação de um conceito abstrato. A diversão não é o objetivo do Virtual Table, embora possa fazer parte do processo. Dentro desta definição, o Virtual Table pode ser considerado como uma solução gamificado de acordo com o quadrante de Deterding.

## DESENVOLVIMENTO GUIADO PELO *GAME DESIGN*

A ideia inicial do Virtual Table era ser uma ferramenta que representasse as estruturas de um algoritmo de maneira visual, devido ao fato de que informações visuais são melhor assimiladas que informações verbais. Durante o desenvolvimento da pesquisa bibliográfica foram analisadas algumas ferramentas de natureza e objetivo semelhante. O Gamification passou a ser então considerado como diferencial do Virtual Table, pois além de concretizar conceitos abstratos de programação ainda haveria o impulso motivacional.

A relação com videogames se estabeleceu naturalmente, pois soluções gamificadas se utilizam dos elementos de jogos, e por ser uma ferramenta digital, o Virtual Table poderia se beneficiar dos elementos utilizados em vídeo-games. Dessa maneira, os princípios de game design já tão bem estabelecidos seriam perfeitos para guiar o desenvolvimento da ferramenta. A metodologia para desenvolvimento de jogos MDA, já descrita no segundo capítulo, direcionou os passos para a criação da ferramenta.

### Aplicação do M.D.A para a criação do Virtual Table

A metodologia define que as mecânicas do jogo devem ser os primeiros aspectos considerados, e partindo das mecânicas que serão utilizados devem ser planejadas as dinâmicas do jogo, que serão então decisivas na escolha da estética do jogo. Partindo da ideia inicial de interatividade dos elementos visuais, foi escolhida como mecânica principal do Virtual Table a organização desses elementos, utilizando o mouse para “arrastar” e “soltar”. Além de passíveis de manipulação, os elementos que representam operadores de cálculo e atribuição de valores ganham uma segunda camada de interatividade quando organizados, se tornando botões que podem ser ativados com o click do mouse. Cada linha do algoritmo é tratada individualmente, criando um algoritmo completo ao final. Dessa maneira o usuário consegue analisar o código por partes e visualizando os resultados de forma dinâmica, como os valores das variáveis naquele momento.

Das mecânicas definidas naturalmente surgiram as dinâmicas, sendo a principal delas a liberdade de interação com o ambiente da ferramenta, a ausência de instruções explícitas e as diferentes estratégias para se alcançar os objetivos utilizando os elementos disponibilizados. Para a estética do jogo foi escolhido algo que fosse integrado de forma coesa com as mecânicas e dinâmicas, e a idéia de basear o funcionamento da ferramenta em circuitos elétricos serviu perfeitamente a esse propósito. Visualmente, os objetos manipuláveis possuem sutis trações tridimensionais, garantindo volume e aos mesmos e reforçando a representação de um objeto sólido. A aparência do módulo de organização dos objetos foi planejada para lembrar as trilhas de um circuito elétrico, assim como a estética visual geral da ferramenta foi baseada em placas de circuitos eletrônicos.

# CONSIDERAÇÕES FINAIS

A pesquisa bibliográfica possibilitou uma concepção diferente a respeito da construção do conhecimento, através da visão do behaviorismo e cognitivismo. Entender como acontece o aprendizado é básico para qualquer solução que se proponha a facilitar o ensino. Apesar dos pontos divergentes, foi possível constatar que as duas teorias têm muito em comum na maneira com que ambas entendem o processo de aprendizado, principalmente na importância do reforço positivo e progressivo do aprendizado. Foi possível compreender o gamification de maneira detalhada, desde sua gênese até o significado do mesmo nos dias de hoje. Como está inferido no próprio nome, a gamificação possui uma relação muito próxima com os jogos, embora possuam significados e propósitos distintos. Compreender estas diferenças, bem como o que separa o *gamification* de outras técnicas de ensino que também utilizam elementos de jogos, é essencial para a construção de soluções gamificadas.

Devido à relação com os jogos, foi possível constatar durante a pesquisa que as técnicas de desenvolvimento de jogos podem ser utilizadas para o desenvolvimento de soluções gamificação, considerando as devidas adaptações. Foi possível constatar durante a pesquisa que as técnicas modernas de *game design* já utilizam muitos conceitos das teorias de ensino, em especial o behaviorismo. Os games também precisam ensinar os primeiros jogadores sobre as suas regras e mecânicas, e já fazem isso de maneira “gamificada”. É interessante notar como os games de certa forma já vem utilizando o *gamification* para ensinar há quase meio século, e fazem isso de maneira intuitiva e invisível para os jogadores. Essa pesquisa forneceu não apenas a base teórica para o desenvolvimento do Virtual Table, mas também um guia para execução do projeto, que foram de grande valia posteriormente.

Durante a concepção das primeiras idéias, foram analisadas algumas ferramentas que possuem propostas semelhantes ao Virtual Table, tanto em busca de alguma forma de inspiração quanto a procura por alguma necessidade que poderia ser preenchida. Todas estas ferramentas utilizavam muito bem o *gamification,* e posteriormente, durante a pesquisa foi possível constatar que estas ferramentas utilizavam a forma de *gamification* de conteúdo, sem grandes alterações em sua estrutura. Os “exercícios” ainda eram muito semelhantes aos aplicados em sala de aula, apesar dos aspectos gamificados como recompensas, rankings e *feedback.*

A idéia inicial do diferencial do Virtual Table era a aplicação *gamification* de forma “embutida”*,* em que todos os aspectos da ferramenta seriam gamificados. Em termos técnicos, o diferencial do Virtual Table seria a utilização do *gamification* em toda a sua estrutura. Antes da concepção deste trabalho não havia o conhecimento dessa maneira “intuitiva” dos jogos de ensinar, e esse foi uma das grandes descobertas da pesquisa bibliográfica. Havia o conhecimento de que pelo fato de compartilharem algumas características, a educação era uma área de fácil adaptação para o *gamification*. Antes da concepção deste trabalho não havia o conhecimento dessa maneira “intuitiva” dos jogos de ensinar, e esse foi uma das grandes descobertas da pesquisa bibliográfica. Havia o conhecimento de que pelo fato de compartilharem algumas características, a educação era uma área de fácil adaptação para o *gamification*.

A forma como o aprendizado acontece nos vídeogames foi a grande inspiração para a maneira como o *gamification* poderia ser aplicado de forma estrutural. A pesquisa sobre os fundamentos do *game design* levou até o M.D.A, que facilitou o “pensamento de games” no projeto e desenvolvimento do protótipo.

As tecnologias que seriam utilizadas para o desenvolvimento do protótipo já estavam definidas antes da pesquisa, principalmente por questões de familiaridade com as ferramentas e o fato de que todas softwares livres de licença. A fase de codificação foi um desafio, devido ao fato de que a programação do protótipo se assemelhava muito à programação de um jogo. A programação de jogos é relativamente diferente da codificação de outros tipos de aplicações, pois exige verificar as ações do usuário a todo instante, os chamados *game loops.* Foi necessário um tempo de adaptação e familiarização com esse “modo de pensar”, aparentemente complexo e impossível à primeira vista, mas muito simples depois de compreendido.

Paralelamente às adaptações de ordem abstrata e cognitiva, estavam às adaptações de ordem técnica. Por ser um framework específicos para o desenvolvimento de jogos, as funções disponibilizadas pelo mesmo são rotineiramente comuns em ambientes de desenvolvimento de jogos, mas desconhecidas para programadores de outro campo. Isso tornou o aprendizado incial um pouco difícil, mas se tornou mais simples em conjunto com o entendimento mais aprofundado sobre o desenvolvimento de jogos.

Por fim, apesar de aplicar todos os conceitos coletados na pesquisa bibliográfica, infelizmente não foi possível testar a eficácia da aplicação. A aplicação do Gamification foi realmente bem-sucedida em atingir seus objetivos na perspectiva do *Designer.* Devido à falta de tempo e disponibilidade não foi possível realizar uma pesquisa quantiva a respeito da eficácia dete protótipo.

# REFERÊNCIAS BIBLIOGRÁFICAS

ALVES, Flora. **GAMIFICATION: Como criar experiencias de aprendizagem engajadoras**. Um guia completo: do conceito à prática. 2 Ed. DVS Editora, São Paulo, 2015.

ANDRADE, Eliziário Souza. **Psicologia da educação**. Rede de ensino FTC, 2009.

ARGENTO, Heloisa. **As Teorias da Aprendizagem**. Apresentação TCC, UFJF, Juiz de Fora- MG ,2003.

AURELIO. **Significado de Aprendizagem**. Disponível em: <https://dicionariodoaurelio.com/aprendizagem>. Acessado em 01 Mai. 2018.

BATTISTELLI, Juliana. **O que é algoritmo: a base do pensamento computacional explicada.** Disponível em: <https://blog.mastertech.tech/tecnologia/o-que-e-algoritmo-base-do-pensamento-computacional-explicada/>. Acessado em 23 Mai. 2018.

BELLO, Robson Scarassati. **História e videogames: como os jogos eletrônicos podem ser pensados por historiadores**. Disponível em: <https://www.cafehistoria.com.br/historia-e-videogames/> em 13 nov. 2017. Acessado em: 22 abr. 2018.

BRAGA, Lucas. **Conheça o Swarm, o novo aplicativo de check-in do Foursquare**. Disponível em <https://tecnoblog.net/156690/conheca-o-swarm-o-novo-aplicativo-de-check-foursquare/> em abr. 2014. Acessado em 30 Abr. 2014.

BRUNA, Maria Helena Varella. **Memória**. Disponível em: <https://drauziovarella.uol.com.br/corpo-humano/memoria/> em 12 Mar. 2018. Acessado 30 Abr. 2018.

BURKE, Brian. Gamificar: Como a gamificação motiva as pessoas a fazerem coisas extraordinárias. DVS Editora, São Paulo, 2015.

CAMBRIDGE, dictionary. **Asset**. Disponível em: <https://dictionary.cambridge.org/pt/dicionario/ingles/asset>. Acessado em: 23 Mai. 2018.

DAVEY, Richard. **Aprenda a Utilizar a Game Engine Phaser Para Desenvolvimento de Jogos em HTML5**. Disponível em: <https://gamedevelopment.tutsplus.com/pt/articles/how-to-learn-the-Phaser-html5-game-engine--gamedev-13643>. Acessado em: 16 abr. 2018.

DETERDING, Sebastian; DIXON, Dan; KHALED, Rilla; NECKE, Lennart. **From Game Design Elements to Gamefulness: Defining Gamification**.

DUOLINGO. Duolingo: **Aprenda idiomas de graça. Para sempre**. Disponível em: <https://pt.duolingo.com/>. Acessado em 06 Abr. 2018.

EIS, Diego. Uma breve história; As 3 camadas de desenvolvimento. In. HTML e CSS3 com farinha e pimenta. São Paulo, 2012. Cap. 1-2.

FLANAGAN, David. **Introduction to JavaScript**. In. **Java Script: The Definitive Guide**. United States of America: O’Reilly Media, Inc, 2006.

FLATSCHART, Fabio. **O Hipertexto e as linguagens de marcação; A evolução do HTML**. In. **HTML5 Embarque imediato**. Rio de Janeiro: Brasport Livros e Multimídia Ltda, 2011. Cap 1-2.

GOOGLE. **Google Trends:** **Veja o que o mundo está pesquisando**. Disponível em: <https://trends.google.com.br/trends/?geo=BR>. Acessado em 02 Jun. 2018.

HUIZINGA, Johan. **Homo Ludens**. Tradução de João Paulo Monteiro. 4º edição, editora Perspectiva S.A, São Paulo, 2000.

KNOTT, Rebeka. **Atari’s pong: The pioneer of video-games!** Disponível em: <http://groovyhistory.com/ataris-pong-the-pioneer-of-video-games> Nov. 2017. Acessado em 30 Abr. 2018.

LINKEDIN. **Linkedin Web**. Disponível em: < https://www.linkedin.com/>. Acessado em 23 Mar. 2018.

MARCONI, Maria de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica.** 5º Ed. São Paulo, Editora Atlas S.A, 2003.

MIGNANO, Marco. **Mechanics Dynamics Aesthetics (MDA): The Game Design Theory Behind Games**. Disponível em: <http://gamedevelopertips.com/mechanics-dynamics-aesthetics-game-design-theory-behind-games/>. Acessado em 23 Mai. 2018.

NICOLIELO, Bruna. **O que é algoritmo?** Disponível em: <https://novaescola.org.br/conteudo/2675/o-que-e-algoritmo>. Acessado em 24 Mai. 2018.

OXFORD. **Definition of Gamification**. Disponível em: <https://en.oxforddictionaries.com/definition/gamification>. Acessado em 25 Mai. 2018.

PELLING, Nick. **Consultoria** **CONUNDRA**. Disponível em: <http://www.nanodome.com/conundra.co.uk/>. Acessado em: 23 abr. 2018.

PORTNOW, James. **Aesthetics of Play - Redefining Genres in Gaming**. Disponível em: <https://www.youtube.com/watch?v=uepAJ-rqJKA&t=1s> em Nov. 2012. Acessado em 23 Mai. 2018.

PRIBERAM, dicionário de língua portuguesa. **Siginificado de Quiz**. Disponível em: <https://www.priberam.pt/dlpo/quiz>. Acessado em 23 Mai. 2018.

Q2L. **About Quest to Learn**. Disponível em <http://www.q2l.org/about/>. Acessado em 30 Abr. 2018.

RIMON, Gal. Microsoft Case Study: **Using Gamified Performance and Learning to Drive Call Center Agents**. Disponível em: <https://www.gameffective.com/using-gamified-performance-and-learning-to-drive-call-center-agents/> em Jan. 2018. Acessado em: 30 Abr. 2018.

RODRIGUEZ, José Paz. **O modelo didático do ensino programado, segundo B. F. Skinner**. Disponível em: <http://pgl.gal/o-modelo-didatico-do-ensino-programado-segundo-b-f-skinner/> em 20 de Ago. 2014. Acessado em 27 de Mai. 2018.

RYAN, Jeff. **Nos Bastidores da Nintendo**. Editora Saraiva; 1ª Edição, 19 de dezembro de 2012.

SILVA, Mauricio Sammy. **Histórico, ferramentas e terminologia**. In. **Fundamento de HTML5 e CSS3**. São Paulo: Novatec Editora Ltda, 2015. Cap. 1.

TAKAHASHI, F. **Matemática e ciências da computação têm alta taxa de abandono**. Disponível em: <http://www1.folha.uol.com.br/educacao/2009/04/546576-matematica-e-ciencias-da-computacao-tem-alta-taxa-de-abandono.shtml> Acessado em 17 Fev. 2018.

VADER, Vince. **Jesper Juul**. Disponível em: <http://gamingconceptz.blogspot.com.br/2011/10/jesper-juul.html>. Acessado em: 17 abr. 2018.

1. Estrutura sináptica: Regiões entre neurônios que agem como mediadores, transmitindo os impulsos elétricos entre dois neurônios. Basicamente, toda a memória e processamento de informações do cérebro é regido pelas sinapses (BRUNA, 2018). [↑](#footnote-ref-1)
2. Corrente epistemológica: Do grego episteme, conhecimento, e *logia*, ciência. É ciência que estuda toda forma de conhecimento, sua natureza e limitações (ANDRADE, 2014). [↑](#footnote-ref-2)
3. Google Trends: É uma ferramenta do Google que mostra os mais populares termos buscados em um passado recente (GOOGLE, 2006). [↑](#footnote-ref-3)
4. *Check-in*: Palavra de origem inglesa também utilizada em português para se referir ao processo de registros de dados pessoais em hotéis e aeroportos. No contexto de redes sócias, check-in é o ato de publicar que você está ou esteve em determinado estabelecimento, geralmente locais de socialização como bares e eventos. Autoria própria (2018). [↑](#footnote-ref-4)
5. *BPO****:***Sigla em inglês para *Business Process Outsourcer,* pode ser entendido como terceirização de processos de negócios não relacionados diretamente à principal estratégia de negócios de uma empresa (RIMON, 2018). [↑](#footnote-ref-5)
6. *Quiz:* Jogo de perguntas e respostas que tem como objetivo testar os conhecimentos do participante em relação à determinado tema (PRIBERAM, 2018). [↑](#footnote-ref-6)
7. Fliperama: Versão aportuguesada da palavra inglês *flipper,* que se refere aos botões deslizantes das máquinas de *pinball*. Popularmente utilizada para se referir às máquinas de vídeo-games que eram jogadas com a inserção de fichas, bastante populares na década de 80. [↑](#footnote-ref-7)
8. CSS: Sigla para *Cascade Style Sheet,* pode ser traduzido como folha de estilos em cascata. O CSS é uma linguagem de marcação que define o estilo (fonte, tamanho, cor) dos elementos HTML. O termo cascata vem do fato de que os elementos herdam o estilo do elemento superior direto na hierarquia (título, subtítulo, corpo do texto). Eis (2012). [↑](#footnote-ref-8)
9. *Asset*: Palavra inglesa utilizada para se referir a algo de valor que pertence à uma pessoa ou organização. No contexto de desenvolvimento de jogos e aplicações, a palavra se refere a todos os arquivos que serão utilizados: imagens, sons, animações, códigos (CAMBRIDGE, 2018). [↑](#footnote-ref-9)