



UNIVERSIDADE FEDERAL DO TOCANTINS
CAMPUS UNIVERSITÁRIO DE PALMAS
PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM COMPUTACIONAL DE
SISTEMAS

JOSÉ ROBSON MARIANO ALVES

JOGOS SÉRIOS E GAMIFICAÇÃO PARA ENGAJAR ALUNOS NA
APRENDIZAGEM DE LÓGICA DE PROGRAMAÇÃO

PALMAS
2019

JOSÉ ROBSON MARIANO ALVES

JOGOS SÉRIOS E GAMIFICAÇÃO PARA ENGAJAR ALUNOS NA APRENDIZAGEM
DE LÓGICA DE PROGRAMAÇÃO

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional de Sistemas como requisito parcial à obtenção do grau de Mestre em Modelagem Computacional de Sistemas.

Orientador: Dr. Patrick Letouze

PALMAS

2019

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da Universidade Federal do Tocantins

A474j Alves, José Robson Mariano.
JOGOS SÉRIOS E GAMIFICAÇÃO PARA ENGAJAR ALUNOS NA
APRENDIZAGEM DE LÓGICA DE PROGRAMAÇÃO. / José Robson
Mariano Alves. – Palmas, TO, 2019.

106 f.

Dissertação (Mestrado Acadêmico) - Universidade Federal do Tocantins
– Câmpus Universitário de Palmas - Curso de Pós-Graduação (Mestrado) em
Modelagem Computacional de Sistemas, 2019.

Orientador: Patrick Letouze

1. Jogos sérios. 2. Gamificação. 3. Ambiente virtual de aprendizagem. 4.
Educação. I. Título

CDD 4

TODOS OS DIREITOS RESERVADOS – A reprodução total ou parcial, de qualquer
forma ou por qualquer meio deste documento é autorizado desde que citada a fonte.
A violação dos direitos do autor (Lei nº 9.610/98) é crime estabelecido pelo artigo 184
do Código Penal.

**Elaborado pelo sistema de geração automática de ficha catalográfica da UFT com os
dados fornecidos pelo(a) autor(a).**



SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DO TOCANTINS
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
COORDENAÇÃO DO PROGRAMA DE PÓS-GRADUAÇÃO
MODELAGEM COMPUTACIONAL DE SISTEMAS

Palmas, 21 de março de 2019.

Aos 21 (cinco) dias do mês de Março de 2019, realizou-se a defesa do Projeto de Pesquisa do aluno José Robson Mariano Alves, do Curso de Mestrado em Modelagem Computacional de Sistemas, da Universidade Federal do Tocantins (UFT), intitulado: **“JOGOS SÉRIOS E GAMIFICAÇÃO PARA ENGAJAR ALUNOS NA APRENDIZAGEM DE LÓGICA DE PROGRAMAÇÃO”**, realizado sob a Orientação do Professor Dr. Patrick Letouze Moreira, tendo como banca avaliadora, os professores abaixo relacionados.


Atribuíram a Nota Final A (_____) pelo trabalho, tendo sido considerado APROVADO. Nada mais tendo a constar, assinam esta Ata os professores componentes da banca.

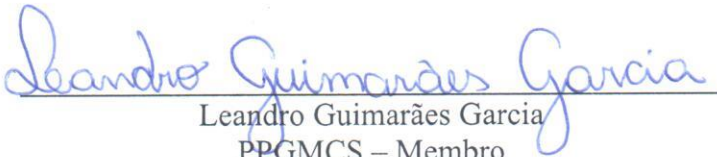
Observações: _____

O coordenador do Curso substituiu o orientador por motivo de licença médica

David Nadler Prata
Coord. Programa de Pós-Graduação
Modelagem Computacional de Sistemas

Patrick Letouze Moreira
PPGMCS – Orientador


Gentil Veloso Barbosa
PPGMCS – Membro


Leandro Guimarães Garcia
PPGMCS – Membro


Mirelle da Silva Freitas
IFTO – Membro Externo

*Dedico este trabalho a minha esposa que
sempre me suporta.*

RESUMO

Este trabalho tem como objetivo demonstrar como a gamificação e jogos sérios para aumentar o engajamento de alunos em disciplinas de lógica de programação, para isto, optou-se pela gamificação de uma disciplina em um ambiente virtual de aprendizagem e pela adaptação de um jogo sério para utilização na sala de aula, dessa forma enquanto o jogo motiva o aluno dentro da sala de aula, o ambiente virtual de aprendizagem motiva o mesmo fora da sala de aula, complementando-se. Serão apresentados os componentes básicos utilizados para a construção de soluções gamificadas, as teorias motivacionais que suportam a gamificação, além de seus benefícios e malefícios. O ambiente virtual de aprendizagem escolhido foi o Moodle, sendo realizado um levantamento dos *plug-ins* disponíveis e quais componentes os mesmos implementam, bem como estratégias para gamificação de disciplinas, as quais também podem ser parcialmente aplicadas em outros ambientes que não o Moodle. O jogo sério escolhido foi o Colobot, o mesmo foi completamente traduzido para o idioma português para facilitar sua utilização em salas de aula brasileiras. Foi realizada uma integração curricular do jogo com uma disciplina introdutória de lógica de programação resultando na criação de missões personalizadas que implementam os itens do currículo, por fim, as missões foram disponibilizadas em um repositório on-line e os resultados obtidos foram as estratégias para gamificação de disciplinas em um ambiente virtual de aprendizagem e a produção de um objeto educacional para ser utilizado em sala de aula.

Palavras-Chave: Jogos sérios, Gamificação, Educação, Ambiente virtual de aprendizagem.

ABSTRACT

This work aims to use gamification and serious games to increase the engagement of students in disciplines of programming logic, for this, we chose to gamificate a discipline in a virtual learning environment and by adapting a serious game to use in the classroom, in this way while the game motivates the student inside the classroom, the virtual learning environment motivates the same outside the classroom, complementing each other. It will be presented the basic components used for the construction of gamification solutions, the motivational theories that support gamification, besides its benefits and evils. The virtual learning environment chosen was Moodle, with a survey of the available plug-ins and which components they implement, as well as strategies for gamification of disciplines, which can also be partially applied in environments other than Moodle. The serious game chosen was Colobot, it was completely translated into Portuguese language to facilitate its use in Brazilian classrooms. A curricular integration of the game was carried out with an introductory discipline of programming logic resulting in the creation of customized missions that implement the curriculum items. Finally, the missions were made available in an online repository and the results obtained were strategies for gamification of disciplines in a virtual learning environment and the production of an educational object to be used in classroom.

Keywords: Serious games, Gamification, Education, Virtual learning environment.

LISTA DE ILUSTRAÇÕES

Figura 1 - Tipos de jogadores	15
Figura 2 - Hierarquia de elementos de jogos	16
Figura 3 - Aplicativo Esquadrão da Dor.....	20
Figura 4 – Layout do Moodle	24
Figura 5 - Acompanhamento de conclusão e restrições de acesso	24
Figura 6 - Emblemas	25
Figura 7 - <i>Level Up!</i> e <i>Level Availability</i>	25
Figura 8 - <i>Stash</i> e <i>Stash Availability</i>	26
Figura 9 - <i>NED Tabs Course Format</i> e <i>NED My Progress</i>	26
Figura 10 – Como gamificar uma disciplina	27
Figura 11 - Ambiente de programação do Colobot	33
Figura 12 - Gestão Interdisciplinar de Projetos de Pesquisa	34
Figura 13 - Aplicação do IRPM neste trabalho	35
Figura 14 - Modelo IPO	36
Quadro 1 - Categorias de ações para robôs no jogo Colobot	37
Quadro 2 - Missão “Coleta de informações”	37
Figura 15 - Código de processamento para completar a missão “Coleta de informações”	38
Figura 16 - Missão “Coleta de informações”	38
Quadro 3 - Traduzibilidade de um jogo.....	39
Figura 17 – <i>Pull Request</i> da tradução do Colobot para o idioma português	40
Figura 18 - Exemplo de tradução	41
Quadro 4 - Objetivos das missões integradas no currículo	42
Figura 19 - Repositório das missões no GitHub.....	43
Figura 20 - Status do repositório das missões	43
Figura 21 - Missões em português no Colobot.....	44
Figura 22 - Manual do Colobot em português.....	45

LISTA DE TABELAS

Tabela 1 - Componentes curriculares e seus capítulos	41
--	----

LISTA DE SIGLAS

AVA	Ambiente Virtual de Aprendizagem.
IPO	Input-Process-Output.
IRPM	Interdisciplinary Research Project Management.
MOODLE	Modular Object-Oriented Dynamic Learning Environment.
MUD	Multi-user Dungeon.

SUMÁRIO

1 INTRODUÇÃO.....	11
2 GAMIFICAÇÃO: A SOLUÇÃO PARA TODOS OS PROBLEMAS?	13
2.1 Tudo se resume a motivação	13
2.2 Elementos de jogo	15
2.3 Um caso de sucesso	19
2.4 Gamificação na educação	20
3 GAMIFICAÇÃO NO AMBIENTE VIRTUAL DE APRENDIZAGEM MOODLE	23
3.1 Recursos e <i>plug-ins</i>	23
3.2 Gamificando uma disciplina.....	27
4 INTEGRAÇÃO DO CURRÍCULO DE UM CURSO INTRODUTÓRIO DE LÓGICA DE PROGRAMAÇÃO COM O JOGO SÉRIO - COLOBOT	32
4.1 O jogo sério Colobot	32
4.2 Gestão Interdisciplinar de Projetos de Pesquisa.....	33
4.3 Integração Curricular	35
4.4 As missões	36
4.5 A tradução	39
4.6 Resultados e discussão	41
5 CONSIDERAÇÕES FINAIS	46
REFERÊNCIAS	48
APÊNDICES	51
APÊNDICE A – <i>THE CURRICULUM INTEGRATION OF A COURSE OF 'INTRODUCTION TO PROGRAMMING LOGIC' WITH A SERIOUS GAME - COLOBOT</i>	52
APÊNDICE B – GAMIFICATION NO AMBIENTE VIRTUAL DE APRENDIZAGEM MOODLE: ELEMENTOS, RECURSOS E ESTRATÉGIAS	59
APÊNDICE C – GAMIFICAÇÃO: A SOLUÇÃO PARA TODOS OS PROBLEMAS?.....	71

APÊNDICE D – <i>AN IMPLEMENTATION METHODOLOGY OF BIOLOGICAL COLLECTIONS TO SPECIFY BASED ON IRPM</i>	82
APÊNDICE E – IMPLEMENTAÇÃO DE ALGORÍTMO NSGA-III COM OPERADOR DE ESTIMATIVA DE DISTRIBUIÇÃO	90
APÊNDICE F – DESEMPENHO PRODUTIVO E TEOR DE AMIDO DE VARIEDADES DE MANDIOCA EM PLINTOSSOLO NA REGIÃO SUDOESTE DO TOCANTINS.....	99

1 INTRODUÇÃO

Dentre as disciplinas da área da tecnologia de informação, a lógica de programação se destaca, não pela sua complexidade, mas por sua importância. É nela que os estudantes são apresentados ao mundo do desenvolvimento de software, os quais são tão importantes para nossa sociedade moderna, compondo um dos pilares da atual era da informação. É quase impossível sobreviver no mundo atual sem ter contato com algum tipo de software, isso sem considerar os softwares embutidos nos equipamentos eletrônicos que usamos no dia a dia.

Nas salas de aula de lógica de programação, como na maioria das salas de aula, a motivação é um problema recorrente. Para combatê-lo novas abordagens foram utilizadas ao longo do tempo, por exemplo, a utilização de jogos ou técnicas baseadas em jogos. É cada vez mais comum a utilização de jogos na educação, pois as novas gerações são cativadas mais facilmente por abordagens dessa natureza. Os jogos já marcam presença há um bom tempo, porém, o novo "jogador" é a gamificação que na última década só vem crescendo, ora obtendo resultados positivos, ora causando efeitos negativos.

Uma revisão sistemática realizada por Silva, Fernandes e Santos (2018) sobre a utilização de jogos digitais no ensino de programação no nível superior chega a conclusão que o uso de jogos ou métodos lúdicos tem uma alta aceitação entre os alunos e promove seu engajamento, também destaca que a maioria dos estudos utilizou como metodologia a teoria pedagógica do construtivismo e a taxonomia de Bloom. Enquanto o construtivismo pode ser utilizado para estimular a produção de conhecimento, a taxonomia de Bloom pode ser utilizada para mensurar efetivamente a aprendizagem dos alunos.

Tendo em mente o poder motivacional dos jogos e o ensino de lógica de programação, este trabalho tem como objetivo utilizar a gamificação e jogos sérios para aumentar o engajamento de alunos em disciplinas de lógica de programação, para isto, optou-se pela gamificação de uma disciplina em um ambiente virtual de aprendizagem e pela adaptação de um jogo sério para utilização na sala de aula. As disciplinas de lógica de programação são as mais problemáticas em relação a absorção de conteúdo, dessa forma, melhorar o engajamento pode resultar em aumento da taxa de aprendizagem. Enquanto o jogo motiva o aluno dentro da sala de aula, o ambiente virtual de aprendizagem motiva o mesmo fora da sala de aula, complementando-se. Este trabalho diferencia-se dos demais pela elaboração de estratégias para gamificação de uma disciplina em um ambiente virtual de aprendizagem e pela produção de um objeto educacional para ser utilizado em sala de aula.

Em relação ao tipo de pesquisa, quanto a abordagem, é uma pesquisa qualitativa, visto que não há coleta de dados quantificáveis. Quanto a natureza, é uma pesquisa aplicada, pois há aplicação prática para resolução de problemas específicos. Quanto aos objetivos, é uma pesquisa exploratória devido ao fato de que não descreve ou explica o problema, foca-se na exploração e análise do mesmo. Quanto aos procedimentos, é uma pesquisa bibliográfica.

O presente trabalho apresenta-se dividido em três capítulos, o primeiro tem como objetivo conceituar gamificação, apresentar os elementos de jogos que a compõem, apresentar casos de uso e explicar como a mesma funciona e porque, quando mal utilizada, pode causar efeitos indesejados.

O segundo capítulo aborda a gamificação de uma disciplina em um ambiente virtual de aprendizagem. O ambiente escolhido é o Moodle. Dentre os motivos para sua escolha, pode-se citar que ele é gratuito, bastante popular e extremamente personalizável, o que facilita muito sua gamificação, além de existirem plug-ins próprios para gamificação em seu repositório online. O capítulo levanta o estado da arte da gamificação no Moodle, bem como quais *plug-ins* estão disponíveis, quais componentes eles implementam e as estratégias básicas para a construção de um ambiente gamificado, as quais podem ser utilizadas parcialmente em outros ambientes que não o Moodle.

O terceiro capítulo descreve a adaptação de um jogo sério para utilização na sala de aula. O jogo escolhido é o Colobot. Ele é um dos poucos jogos que permitem o uso de algoritmos para resolver problemas, ou seja, é um jogo especialmente feito para os jogadores utilizarem suas habilidades de lógica de programação. A adaptação se resume a integração curricular da disciplina com o jogo, ou seja, a criação de um conjunto de missões personalizadas que implementem os itens do currículo de uma disciplina de introdução a lógica de programação, bem como apresenta a tradução do jogo que foi realizada durante o processo de desenvolvimento das missões.

2 GAMIFICAÇÃO: A SOLUÇÃO PARA TODOS OS PROBLEMAS?

Os jogos estão presentes na sociedade humana desde a antiguidade, são utilizados principalmente para fins de entretenimento pois possuem grande capacidade de engajamento, ou seja, evocam dedicação nos jogadores, resultando em uma participação com afinco e vontade. A popularidade dos jogos está relacionada à diversão que eles proporcionam, mas qual a fórmula do seu sucesso? Pode-se usar essa fórmula em outras áreas? A resposta para essas perguntas é a Gamificação. Com isto em mente, este capítulo se propõe a conceituar a gamificação, levantar os componentes que a compõe e teorias motivacionais que a suportam, bem como apresentar sua utilização na educação.

O termo Gamificação vem do inglês *gamification*, foi usado pela primeira vez em 2002, porém só passou a ser difundido no segundo semestre de 2010 quando se popularizou. O termo gerou muita controvérsia por causa de múltiplas interpretações. O trabalho de Deterding et al. (2011) foi o primeiro a apresentar uma definição formal para a gamificação: uso de elementos de design de jogos em contextos não relacionados a jogos. No ano seguinte Kapp (2012) define gamificação como a utilização de mecânicas, aparência e pensamentos baseados em jogos para motivar pessoas, promover a aprendizagem e resolver problemas em contextos que não são jogos.

Perceba que na definição de Kapp a motivação entra em cena, pois um dos propósitos da gamificação é utilizar a fórmula dos jogos (mecânicas, aparência e pensamentos) para engajar os envolvidos. É comum encontrar professores no ensino fundamental que dão estrelinhas para alunos exemplares, esse é um exemplo básico de gamificação na sala de aula - contexto não relacionado a jogos – onde as estrelinhas são medalhas, uma mecânica encontrada em jogos para promover engajamento.

Deve-se frisar que jogos sérios, jogos educacionais ou ambientes que simplesmente tenham a aparência de jogos, porém não tem como foco motivar seus usuários a cumprir um objetivo, não são exemplos de gamificação. A gamificação vai além de simplesmente inserir um sistema de pontos ou um avatar, ela deve motivar seus usuários, seu objetivo final não é a diversão e a mesma não é um jogo, ou seja, seu propósito não é ser jogável.

2.1 Tudo se resume a motivação

Segundo Skinner (2003)¹ em sua teoria do condicionamento operante, é possível modelar o comportamento de um indivíduo a partir de estímulos. Ele fez experimentos com ratos e desenvolveu uma caixa conhecida como “Caixa de Skinner”, onde um rato é colocado no interior da mesma junto a uma alavanca que quando puxada pelo rato o fornecia alimento, desta forma o rato aprendia a utilizar a alavanca quando necessitasse alimento, caso o alimento parasse de ser fornecido quando o rato puxasse a alavanca, o mesmo cessaria esse comportamento, mas se o fornecimento fosse mantido com intervalos regulares, por exemplo, puxar a alavanca 10 vezes para obter alimento, o comportamento era preservado. Veja o exemplo clássico do caça-níqueis, a máquina sempre recompensará o jogador em um intervalo de tempo aleatório, se ela nunca o recompensasse, ele não jogaria.

Esse conceito de condicionamento operante é muito utilizado em jogos e gamificação, por exemplo, alguns aplicativos de compras dão parte de um cupom (estímulo) para os clientes que logam todo dia (comportamento), e um cupom completo para cada 7 dias (reforço com intervalo fixo), com isso induzem o cliente a entrar no aplicativo diariamente.

A recompensa de Skinner gera uma motivação denominada extrínseca, pois é resultante de um estímulo externo, caso o estímulo seja removido, a motivação deixa de existir. Motivações resultantes de estímulos internos são denominadas intrínsecas. Tome como exemplo a doação de sangue, se a pessoa doar sangue por altruísmo, a motivação é intrínseca, caso a doação seja motivada por uma recompensa, será extrínseca.

Pink (2010) nos diz que motivações extrínsecas duram pouco tempo e podem causar o efeito contrário, pois o usuário foca-se apenas na recompensa, diferente das motivações intrínsecas que promovem um maior envolvimento, pois são motivadas por satisfação própria. A gamificação promove principalmente motivações intrínsecas, o que a diferencia de outras abordagens.

Ainda falando sobre motivação, cada pessoa possui particularidades que influenciam sua motivação para jogar, Bartle (1996) dividiu os jogadores de MUDs (Multi-user Dungeon) – um tipo de jogo para múltiplos jogadores - em 4 tipos de acordo com sua motivação: Assassinos, Exploradores, Socializadores e Conquistadores.

Os assassinos são os mais competitivos, estão interessados em demonstrar sua superioridade para outros humanos, são orgulhosos de suas habilidades e acham mais divertido usá-las em uma pessoa real ao invés de uma entidade computadorizada sem emoção.

1 Original de 1953

Os exploradores são os mais curiosos, querem descobrir o máximo que puderem, gostam de interagir com o jogo, deixar que o jogo os surpreenda, explorar o mundo, descobrir novas áreas, são orgulhosos de seu conhecimento.

Os socializadores são os mais sociáveis, estão interessados em interagir com outras pessoas, gostam de conversar e conhecer pessoas, o jogo é apenas um fundo, são os personagens que o tornam interessante, são orgulhosos de suas amizades, contatos e influência.

Os conquistadores são os mais ambiciosos, estão interessados em fazer coisas no jogo, em conquistar os objetivos, subir no ranking, conseguir pontos e medalhas, são orgulhosos de seu status na hierarquia do jogo.

Figura 1 - Tipos de jogadores



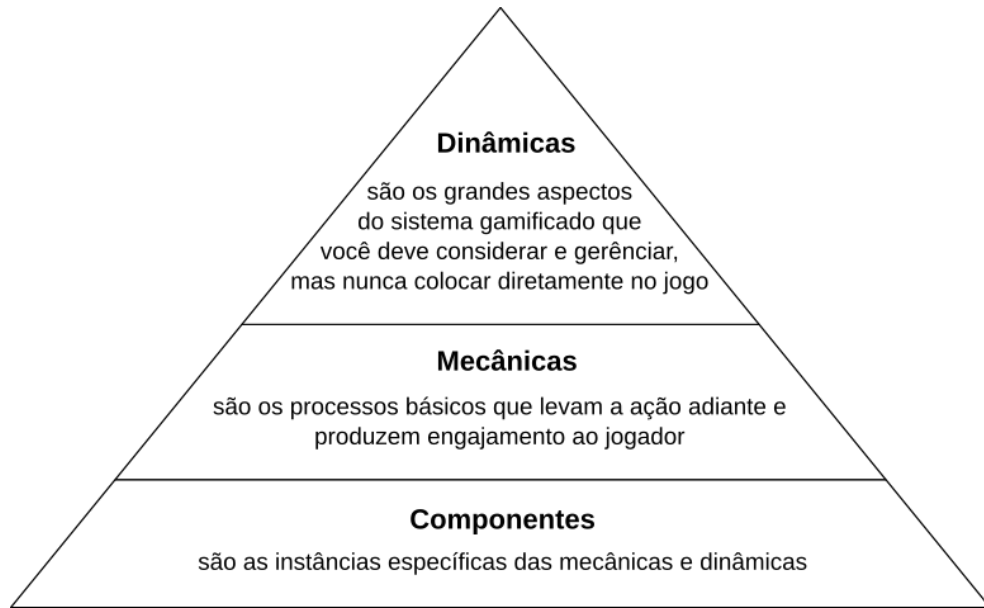
Fonte: Adaptado de Bartle (1996).

É importante entender o que motiva os jogadores a jogar, pois você pode (e deve) explorar essas fontes de motivação. O estudo de Bartle não é o único, existem estudos posteriores mais abrangentes, em especial o de MARCZEWSKI (2015) que além de diversificar o assunto, fornece um framework para identificação de tipos de jogadores.

2.2 Elementos de jogo

Para gamificar algo é necessário utilizar elementos de jogos, neste caso, é preciso conhecê-los, Werbach e Hunter (2015) constroem uma hierarquia e dividem os elementos em 3 tipos: Dinâmicas, Mecânicas e Componentes.

Figura 2 - Hierarquia de elementos de jogos



Fonte: Adaptado de Werbach e Hunter (2015).

Werbach e Hunter (2015) frisam que eles não abordam todos os componentes existentes, mas os mais importantes estão na lista. As dinâmicas são os elementos do topo da hierarquia, são complexas para projetar e úteis para apontar a direção correta para onde você deve ir. Existem 5 dinâmicas:

Restrições – restringem as ações do jogador, forçando que o mesmo escolha somente as opções disponíveis. Os jogos são compostos por regras e limitações, restringir as ações do usuário aumentam o engajamento, por exemplo, uma limitação de tempo gera tensão que pode aumentar o foco do usuário na tarefa.

Emoções – como chave da motivação intrínseca, as emoções são fundamentais para motivar os jogadores. Curiosidade, competitividade, felicidade, orgulho, até emoções negativas como frustração podem ser utilizadas para gerar motivação.

Narrativa – é responsável por dar sentido ao jogo, pode ser explícita – uma história está sendo contada – ou implícita, estabelecendo lógica entre os componentes do jogo. Uma narrativa mal construída pode passar a sensação de que os componentes e personagens do jogo não se encaixam, enquanto uma boa passa a sensação de que cada ação está conectada a um evento maior.

Progressão – jogos não podem ser estáticos, é necessário que haja alguma dinâmica de progressão pelo qual o jogador perceba que ele está avançando, esse feedback engaja o jogador a seguir em frente.

Relacionamentos – jogos são sociais, apesar de existirem jogos para um só jogador, a grande maioria implementa interações entre jogadores, desde competições a jogos cooperativos. Com o advento dos jogos online essa dinâmica é cada vez mais utilizada, basta olhar para os jogos de maior sucesso.

As mecânicas são responsáveis por levar o jogo adiante e criar o engajamento, normalmente são meios para implementar uma ou mais dinâmicas. As mecânicas são:

Desafios – são tarefas que requerem mais esforço para resolver, dão uma sensação de superação ao serem completadas, quanto maior a dificuldade, maior essa sensação, respeitando-se o limite do impossível.

Sorte – significa que deve haver um pouco de aleatoriedade, pode ser utilizado para diminuir a previsibilidade ou como fator de equilíbrio. Pode ser frustrante se mal gerenciada.

Competição – um jogador ou grupo ganha enquanto o outro perde, apesar de haver jogos não competitivos, a grande maioria implementa algum tipo de mecânica de competição.

Cooperação – pode acontecer em conjunto com a competição, dois ou mais jogadores se juntam para cumprir um objetivo comum o qual não conseguiriam individualmente.

Feedback – é a resposta dada ao jogador informando-o sobre como ele está indo, pode ser uma mensagem ou um sistema completo que o informa em tempo real.

Aquisição de recursos – é o ato dos jogadores poderem coletar itens que podem ser úteis no jogo, utilizados para troca entre outros jogadores, ou simplesmente porque é divertido.

Recompensas – é um benefício dado ao jogador por suas ações ou conquistas. Podem ter valor no jogo como itens ou pontos.

Transações – é a troca de itens que pode acontecer entre jogadores, lojas do jogo ou com o próprio sistema. Alguns jogos possuem até casas de leilão, criando assim uma economia.

Turnos – muito utilizados em jogos de cartas e tabuleiros, exigem que cada jogador faça todas as suas jogadas antes de passar a vez para o próximo jogador.

Estados de vitória – são os objetivos que definem a vitória do jogador, por exemplo, destruir todos os inimigos ou acumular 100 pontos.

Por último temos os componentes que são instâncias das mecânicas que por sua vez são instâncias das dinâmicas, estes são realmente vistos pelos jogadores, diferente das mecânicas e dinâmicas. Os componentes mais utilizados são pontos, medalhas e tabelas de classificação.

Conquistas – são objetivos definidos, podem oferecer recompensas e normalmente são utilizados para premiar feitos memoráveis, por exemplo, matar 500 inimigos sem ser morto ou explorar todo o mapa do mundo.

Avatares – são as representações visuais dos jogadores, variam desde uma foto a um modelo 3D. Devem representar o jogador de maneira única, garantindo-o uma identidade.

Medalhas – são representações visuais das conquistas, normalmente podem ser exibidas a outros jogadores de maneira a demonstrar os méritos do jogador, basta lembrar das medalhas dadas em competições esportivas.

Lutas contra chefes – em jogos um “chefe” é um monstro muito forte em comparação aos demais e normalmente é encontrado ao fim de um nível, em que a sua derrota permite que o jogador avance até o próximo nível, nesse caso, representa um desafio maior que os demais, pode oferecer recompensas a altura para recompensar o esforço.

Coleções – são conjuntos de itens, equipamentos ou outros recursos do jogo, algumas conquistas podem exigir, por exemplo, que o jogador reúna todos os itens de uma coleção.

Combate – uma forma de competição, pode acontecer entre jogadores ou entre o jogador e um monstro do jogo, por exemplo.

Desbloqueio de conteúdo – quando o jogo libera novos conteúdos caso o jogador atinja objetivos pré-definidos, estes conteúdos podem ser novas áreas do jogo, novas missões ou até novos modos de jogar.

Presentear – o jogo permite que o jogador dê itens ou recursos para outro jogador de forma voluntária e altruísta.

Tabelas de classificação – também conhecidos como rankings, são tabelas que mostram a classificação do jogador em relação a outros jogadores seguindo critérios específicos.

Níveis – passos na progressão do jogador, ajudam o mesmo a perceber como ele está em relação aos outros e facilita a implementação de restrições e recompensas.

Pontos – representações numéricas da progressão do jogo, normalmente são utilizados como recompensas e sua acumulação possibilita subir de nível.

Missões – desafios que normalmente possuem uma história e dão recompensas quando completadas, Missões podem ser encadeadas de forma a unir seus enredos compondo uma narrativa maior.

Gráfico social – exibe as conexões sociais do jogador. Muito utilizados em redes sociais nas quais os jogadores podem compartilhar conquistas com os amigos, verificar o status dos competidores e até solicitar recursos.

Times – grupos de jogadores que trabalham juntos em prol de um objetivo comum.

Bens virtuais – recursos do jogo que são obtidos com dinheiro virtual, ou até mesmo com dinheiro real, podem ser desde itens simples como uma arma até uma propriedade, como uma casa.

2.3 Um caso de sucesso

Conhecendo os elementos de jogos e tendo em mente que gamificação não é somente pegar qualquer elemento e sair utilizando, mas sim organizá-los de maneira lógica e estruturada, com uma boa narrativa de forma a motivar intrinsecamente os usuários tendo em mente que eles se diferem entre si, então você conseguiu compreender o que é gamificação. Deve-se frisar que o objetivo deste capítulo é conceituar gamificação, ainda há muito para se falar no quesito de projetar sistemas gamificados.

Para solidificar sua compreensão, Burke (2015) em seu livro “Gamificar: como a gamificação motiva as pessoas a fazerem coisas extraordinárias” cita vários casos de sucesso que demonstram a aplicabilidade dos conceitos vistos até agora, pegarei como exemplo o caso do Hospital for Sick Children (Hospital para Crianças Doentes) em Toronto no Canadá que trata de crianças que lutam contra o câncer.

O hospital está sempre tentando achar as melhores terapias enquanto minimiza a dor dos pacientes, para isto ele precisa receber relatórios diários das crianças sobre o nível de dor que estão sentindo, porém, o tratamento é doloroso e as crianças acabam na maioria das vezes não preenchendo corretamente os relatórios, gerando informações incongruentes.

Os pesquisadores decidiram tentar outra abordagem, visualizaram o problema do **ponto de vista das crianças** e junto a uma agência de comunicação chamada Cundari desenvolveram um aplicativo para celular chamado Esquadrão da Dor, cujo objetivo era coletar diariamente os níveis de dor das crianças. Como narrativa o aplicativo lista todos os pacientes como membros de um esquadrão especial da polícia cuja missão é capturar e destruir a dor. O aplicativo notifica as crianças para que elas reportem os níveis de dor duas vezes ao dia, porém é necessário motivar intrinsecamente as crianças para que a proposta tenha sucesso.

Antes do aplicativo as crianças recebiam alguma recompensa física, porém motivações extrínsecas não ajudariam neste caso, a solução encontrada foi criar um sistema de patentes (níveis) em que caso a criança preenchesse o relatório 3 dias seguidos passaria de cabo para sargento e assim por diante. Desta forma, lembrando um jogo, as crianças conseguiam ver seu progresso, ver as condecorações que já ganharam e quando teriam que preencher seu próximo

relatório. E equipe também pegou atores de seriados policiais da TV e criaram vídeos de encorajamento que são exibidos juntos as missões.

Figura 3 - Aplicativo Esquadrão da Dor



Fonte: www.cundari.com.

Este é um exemplo de sucesso da aplicação de gamificação para motivar seus usuários, as crianças sentem que são parte de algo maior e isso as motiva, a taxa de participação no preenchimento dos relatórios aumentou, bem como a obtenção de valores mais precisos.

2.4 Gamificação na educação

A gamificação já vem sendo utilizada na educação a algum tempo, com vários casos de sucesso quando bem empregada, por exemplo, Freitas et al. (2016) gamificaram uma disciplina de um curso de graduação de forma que os alunos tivessem que produzir conteúdo em casa e levar para a sala de aula, com isto obtiveram resultados satisfatórios como: maior empenho dos alunos, um aumento do tempo gasto em casa preparando-se para a aula, maior uso da bibliográfica da disciplina e um aumento na taxa de aprendizagem. A motivação obtida foi tamanha que em alguns casos foi necessário “desmotivar” o jogador pois as outras disciplinas estavam sendo ignoradas. Almeida et al. (2016) também obtém resultados satisfatórios com sua avaliação do processo de gamificação de uma disciplina, reforçando ainda mais a ideia de que a gamificação pode ajudar a melhorar o engajamento dos alunos, consequentemente aumentando a taxa de aprendizagem.

O uso da gamificação não implica no uso de ferramentas como um computador, um tablet ou um smartphone, porém a maioria das implementações utiliza alguma forma de tecnologia devido a facilidade de quantificação de dados e feedback imediato que a mesma possibilita. Na educação a distância é comum o uso de AVAs (Ambiente Virtual de

Aprendizagem) para gerenciamento das disciplinas, conteúdos e atividades, porém esses ambientes também podem ser utilizados no ensino presencial, não só pelos benefícios que os mesmos proporcionam, mas também pelo seu potencial para gamificação. Albino et al. (2018) fala sobre a gamificação de AVAs e defende o seu uso para gamificação, independente se o curso é a distância ou não e cita que pode ser um bom “ponto de partida” para professores que desejam gamificar suas disciplinas.

A gamificação é utilizada para motivar os alunos e acaba alterando seu comportamento. Linehan, Kirman e Rochee (2015) utilizam psicologia comportamental para explicar os efeitos da gamificação, concluem que uma melhor compreensão dos princípios de psicologia comportamental possibilitaria a construção de soluções gamificadas mais eficientes. Apesar da gamificação estar ligada ao behaviorismo, isto não significa que eles são dependentes, pelo contrário, a gamificação pode ser utilizada para motivar os estudantes em qualquer tipo de abordagem, porém, alguns elementos de jogos, como a competição, podem não ser ideais em todas as abordagens.

É importante frisar que a gamificação não veio para resolver todos os problemas da educação e que apesar de resultados positivos terem sido obtidos com seu uso na educação, ela também pode ser prejudicial. Toda, Valle e Isotani (2017) em seu estudo sobre o lado negro da gamificação identificam 4 resultados negativos que a gamificação pode causar: perda de desempenho, comportamento indesejado, indiferença e efeitos de declínio.

A perda de desempenho nos estudos que Toda, Valle e Isotani (2017) analisaram é decorrente do não entendimento das regras, dos jogadores não gostarem de serem penalizados, de jogadores mais ativos ganharem menos pontos do que outros menos ativos, de jogadores focarem mais nas mecânicas que no conteúdo ou da dificuldade das atividades gamificadas. O comportamento indesejado ocorreu porque a gamificação causou um efeito diferente do esperado, devido a mau planejamento ou a falta do mesmo, causando por exemplo, competição excessiva, frustração por não conseguir completar os objetivos e sistemas pouco atraentes e/ou com erros. A indiferença acontece quando a gamificação não consegue influenciar os jogadores, sua implementação não engajou os alunos, nem os afastou, chegando ao ponto de que os alunos prefeririam o sistema tradicional. Por fim, os efeitos de declínio referem-se a perda gradual da motivação por causa da implantação da gamificação, que levam a perda de desempenho e de motivação com o tempo.

Toda, Valle e Isotani (2017) concluem que “esses problemas ocorreram devido à falta de métodos apropriados e/ou estruturas para planejar e implantar gamificação em um contexto de aprendizado”. Eles também relatam a falta de teorias instrucionais para suportar a

implantação da gamificação, essas teorias são utilizadas para produção de estratégias de gamificação otimizadas para cada situação.

Com isto observamos que uma estratégia de gamificação mal planejada pode ser desastrosa e que sua utilização deve ser tratada com cuidado, afinal cada pessoa tem um perfil motivacional que deve ser levado em conta, bem como cada elemento de jogo tem sua situação ideal para aplicação. Espero que o leitor entenda que é necessário estudar a gamificação a fundo e construir um bom plano antes de sair aplicando-a em suas salas de aula.

3 GAMIFICAÇÃO NO AMBIENTE VIRTUAL DE APRENDIZAGEM MOODLE

O sistema educativo sofre alterações constantes ao longo do tempo, atualmente é comum ver escolas utilizando ambientes virtuais de aprendizagem, sistemas que proporcionam uma integração online entre professor e estudante. Apesar dos benefícios, continua sendo um desafio motivar os estudantes, especialmente em ambientes como esses, onde existe uma distância física.

Kapp (2012) nos diz que no quesito “motivar pessoas” existem áreas bem-sucedidas, por exemplo: jogos. Pare um pouco e olhe ao seu redor: seus amigos estão jogando no celular, seu filho está jogando no *tablet*, seu parceiro está jogando no *facebook*, seus avós estão jogando na mesa. Os jogos estão presentes no cotidiano das pessoas, sejam crianças ou adultos, até mesmo idosos, todos estão jogando, e o mais incrível nisso tudo, é que nenhum deles está sendo obrigado a jogar, todos estão jogando porque querem. Com isto em mente, profissionais estão tirando as características dos jogos que os fazem tão interessantes e implementando-as em outras áreas, principalmente na educação.

O uso de gamificação não é uma abordagem nova, são utilizadas em diversas áreas, bem como existem diversos exemplos de uso no ensino, com altos índices de aprovação pelos usuários, inclusive em AVAs. É importante ressaltar que a má utilização de gamificação pode provocar um efeito contrário, por exemplo, estudantes que se interessam somente pelos prêmios e não pelo conteúdo. (BUSARELLO, 2016)

3.1 Recursos e *plug-ins*

O *Modular Object-Oriented Dynamic Learning Environment* (Moodle) é um AVA livre e altamente personalizável. A maioria das funcionalidades disponíveis são *plug-ins* individuais ou coleções dos mesmos. É possível estender as funcionalidades através da instalação de *plug-ins* criados pela comunidade que podem ser obtidos no diretório de *plug-ins* do Moodle. (HENRICK; HOLLAND, 2015)

Diversos *plug-ins* são disponibilizados em forma de blocos. Blocos são pequenos componentes personalizáveis adicionados ao *layout* (Figura 4), posicionam-se a esquerda e/ou a direita do conteúdo e fornecem diversas funcionalidades, como um calendário, um relatório ou até mesmo uma lista de usuários *on-line*.

Figura 4 – Layout do Moodle



Fonte: Autor.

Atividades são como o nome diz, interações que permitem avaliar a participação do estudante através da atribuição de notas. Segundo DENMEADE (2015), dentre os recursos nativos do Moodle, deve-se dar destaque a dois que podem ser aplicados a atividades e são extremamente importantes ao se trabalhar com gamificação:

1. Acompanhamento de conclusão – adiciona uma caixa de seleção ao lado da atividade que será marcada automaticamente quando o estudante satisfizer as condições definidas pelo professor. Caso não se use condições, o estudante pode marcar manualmente;
2. Restrições de acesso – permite-se adicionar restrições que impedem o acesso a determinada atividade ou conteúdo, as restrições disponíveis incluem data, nota, conclusão de atividade e perfil de usuário. *Plug-ins* de restrições de disponibilidade adicionam novos tipos de restrições.

Através dos recursos de acompanhamento de conclusão e restrições de acesso obtém-se o componente citado anteriormente: **desbloqueio de conteúdo** (Figura 5).

Figura 5 - Acompanhamento de conclusão e restrições de acesso



Fonte: Autor.

O Moodle possui suporte nativo a emblemas (Figura 6), ou seja, não é necessário instalar. Esse recurso possibilita dar emblemas a estudantes de forma manual ou automática,

desde que satisfaçam os requisitos estabelecidos. O uso de emblemas no Moodle caracteriza o componente *badges* (distintivos), que representa um tipo de **realização**, outro componente.

Figura 6 - Emblemas



Fonte: Autor.

Level Up! e *Level Availability* (Figura 7) são dois *plug-ins* criados pela comunidade que implementam os componentes **pontos**, **níveis**, **placar** e **desbloqueio de conteúdo**. Eles podem ser obtidos através do diretório de *plug-ins* do Moodle. O primeiro possibilita ao estudante acumular pontos e subir de nível quando completa atividades e interage com o AVA, bem como disponibiliza um *ranking* que lista os estudantes de acordo com a quantidade de pontos. O segundo permite criar restrições por nível nas atividades e conteúdos. A forma como o estudante ganha os pontos e a quantidade de pontos por nível pode ser personalizada pelo professor.

Figura 7 - *Level Up!* e *Level Availability*

Missão secreta

Restrição

Disponível se: Você está no nível 2 ou maior.

Posição	Nome completo	Nível	Pontos de experiência	Progresso
1	José Robson Mariano Alves	2	150	30/156
2	Dannetha de Oliveira Rocha	1	50	50/120

Ranking

MEU NÍVEL

1

30/120

Participe do curso para ganhar pontos de experiência e passar de nível!

Informação - Visualizar o ranking

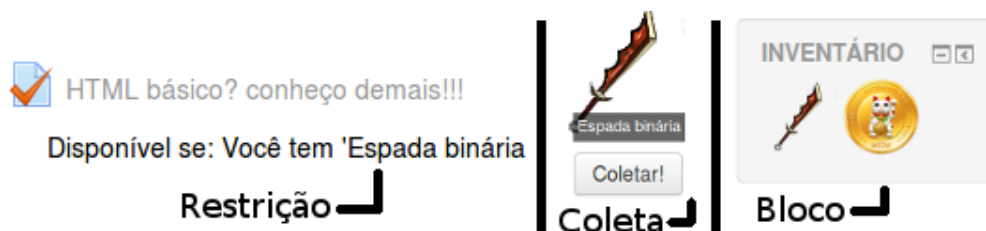
Bloco

Fonte: Autor.

Os componentes **exploração**, **coleções** e **desbloqueio de conteúdos** podem ser implementados através dos *plug-ins* *Stash* e *Stash Availability* (Figura 8), também disponíveis

no diretório de *plug-ins* do Moodle. O primeiro permite ao professor criar “objetos” que podem ser colocados em qualquer lugar do curso, quando o estudante passar por ali ele poderá coletar o item e armazenar em sua coleção. O segundo permite restringir acesso ao conteúdo de acordo com objetos da coleção do usuário.

Figura 8 - *Stash e Stash Availability*

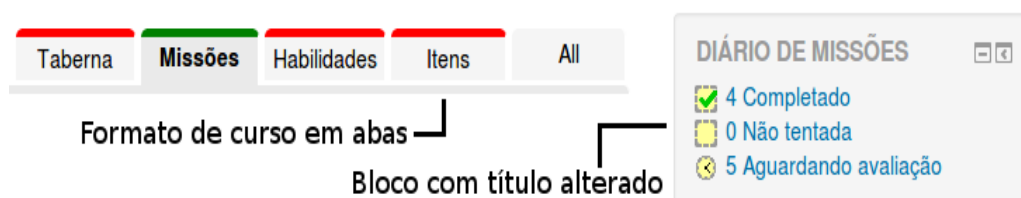


Fonte: Autor.

O *Active Quiz* é um *plug-in* do tipo atividade e pode ser utilizado para implementar o componente **combate** (competição) entre os estudantes. Ele permite criar questionários e sessões com os mesmos, os estudantes se juntam à sessão e começam a responder as perguntas antes do tempo limite acabar. O professor tem acesso a um relatório de classificação após o fim da sessão.

Os *plug-ins* *NED Tabs Course Format* e *NED My Progress* (Figura 9) aumentam o *feedback* ao estudante, o primeiro organiza o curso em formato de abas, sendo que as abas vermelhas simbolizam a existência de atividades não realizadas. O segundo cria um bloco que fornece um relatório ao estudante sobre as atividades realizadas e não realizadas, o título desse bloco é personalizável.

Figura 9 - *NED Tabs Course Format e NED My Progress*



Fonte: Autor.

O sistema de mensagens do Moodle, a criação de grupos para realização de tarefas e atividades do tipo fórum, *wiki* e *chat* possibilitam interação e colaboração entre os estudantes. O *plug-in* **Game** também deve ser citado, pois adiciona vários tipos de atividades em forma de

jogos ao Moodle, como cruzadinhas, jogo da forca, entre outros.

3.2 Gamificando uma disciplina

Um AVA que utiliza técnicas apropriadas de gamificação consegue atrair mais a atenção dos estudantes, e conseqüentemente, aumenta a taxa de retenção de conteúdo, porém utilizar a gamificação no processo de aprendizagem é uma tarefa árdua. (BUSARELLO, 2016)

Para demonstrar o processo de gamificação de uma disciplina no Moodle optou-se por dividir as ações por tempo de execução (Figura 10), onde a seção “apenas uma vez” contém ações que serão executadas na criação na disciplina, enquanto as demais seções no decorrer das aulas.

Figura 10 – Como gamificar uma disciplina

APENAS UMA VEZ	ESPORADICAMENTE	SEMPRE
 Adicione o bloco Level Up!	 Apresente desafios	 Adicione uma pitada de humor e fantasia
 Adicione o bloco Stash	 Inicie campanhas	 Utilize alguma forma de narrativa
 Adicione o bloco My Progress	 Crie baús do tesouro	 Lembre-se da teoria do fluxo
 Mude para o layout de abas	 Realize combates	 Instigue uma competição saudável
 Defina uma nova nomenclatura	 Esconda itens	 Estimule a motivação intrínseca
 Use rótulos com imagens	 Recompense realizações com emblemas	 Utilize o desbloqueio de conteúdos

Fonte: Autor.

O layout do Moodle é composto por blocos e o professor tem o poder de adicioná-los ou remove-los conforme sua necessidade, desta forma os itens “**Adicione o bloco Level Up!**”, “**Adicione o bloco Stash**” e “**Adicione o bloco My Progress**” referem-se a adição dos respectivos blocos a disciplina, ação que ativa a utilização destes plug-ins. “**Mude para o layout de abas**” significa mudar o layout do curso para o layout de abas mencionado no capítulo anterior.

Neto (2015) diz que a temática ou a fantasia é um aspecto motivacional importante.

Com isso em mente, personalizar o ambiente para que o mesmo se torne diferente do mundo real ajuda na motivação do estudante.

Defina uma nova nomenclatura - criar uma nomenclatura única que irá contribuir para a fantasia, como por exemplo, alterar o nome dos blocos, chamar o estudante de guerreiro, de missões as atividades, de habilidades o conteúdo a ser aprendido, de itens as ferramentas a serem utilizadas etc. A figura 9 mostra um ambiente com nomenclatura personalizada. Utilizar um sistema de notas por conceitos também contribui para a fantasia, por exemplo, no lugar de notas de 0 a 10, pode-se utilizar conceitos como Bronze, Prata, Ouro, Platina, Diamante etc.

Use rótulos como imagens - os Rótulos são componentes adicionados ao curso que podem conter textos e imagens. Utilizá-los com sabedoria pode mudar completamente a aparência do curso e torná-lo mais atraente e convidativo. Utilize-os para construir o layout da disciplina, ou seja, o mundo por onde os alunos irão viajar.

Apresente desafios - desafios como *Boss Fights*, atividades com um nível de dificuldade maior em relação às outras, entretanto com recompensas maiores, desafiam o aluno e proporcionam uma sensação maior de superação ao completá-las. Como recompensas, por exemplo, pode-se oferecer um emblema que dá algum bônus à nota do estudante ou itens que desbloqueiam um “baú do tesouro”.

Inicie campanhas - ao começar uma narrativa em uma atividade e continuá-la em mais uma ou duas atividades, cria-se uma série ou campanha. Para que a narrativa seja seguida à risca, pode-se criar as atividades de forma a possuírem restrições de conclusão em relação as outras, por exemplo, o estudante não poderá realizar a segunda atividade sem completar a primeira e a narrativa da segunda atividade é uma continuação da narrativa da primeira.

Crie baús do tesouro - para incitar a curiosidade pode-se criar “baús do tesouro”, conteúdos que possuem múltiplas restrições, porém contém grandes recompensas. No Moodle o recurso **página** é uma boa escolha para criação de baús do tesouro, pois permite colocar conteúdo estático sem necessidade de avaliação, porque não é um tipo de atividade. Para limitar o acesso ao “baú” pode-se criar uma restrição de um objeto do *Stash*, como uma “chave”, obtida através de uma atividade. Caso queira aumentar o desafio pode-se utilizar múltiplas restrições, como múltiplas “chaves” para abrir um “baú”. Ao abrir esse “baú” o estudante encontraria itens ou o final de uma narrativa, por exemplo.

Realize combates - os combates podem ser realizados por meio do plug-in *Active Quiz* citado no capítulo anterior.

Segundo BUSARELLO (2016) as motivações extrínsecas originam-se do mundo que envolve o usuário, ou seja, são externas a ele e têm como ponto de partida o desejo de obter

uma recompensa como reconhecimento social ou prêmios. Os pontos, o *ranking*, os emblemas e as coleções são algumas formas de incentivar a motivação extrínseca.

Esconda itens - para motivar os estudantes a visitar os fóruns regularmente e conseqüentemente adquirir esse hábito, pode-se colocar objetos do *Stash* nos fóruns e regularmente postar atividades que necessitem desses objetos para desbloqueá-las.

Conforme o exemplo acima, a motivação extrínseca foi utilizada para instigar hábitos nos estudantes, porém é necessário frisar que um efeito negativo pode ser obtido, por exemplo, os estudantes visitam os fóruns somente para coletar os itens e nunca olham os tópicos. Aliar alguma ação do estudante à obtenção do prêmio pode ajudar a evitar esses problemas. O Moodle oferece fóruns do tipo **perguntas e respostas**, nos quais somente após o estudante responder, ele poderá ver as respostas dos outros, dessa forma é possível esconder o objeto e liberar o acesso somente quando o estudante contribuir para a discussão.

Criar atividades do tipo **lição** permite, além do conteúdo para estudo, a inserção de perguntas. Pode-se personalizar o *feedback* positivo dessas perguntas e colocar um item do *Stash* que desbloqueie uma atividade. Dessa forma, para que o estudante tenha acesso à atividade ele deverá visualizar a lição, ler o conteúdo e responder à pergunta corretamente para coletar o item.

A surpresa também é um elemento fundamental em ambientes que utilizam gamificação. O recurso **Restrições de acesso** permite obter esse comportamento no Moodle, com ele pode-se criar atividades ocultas que aparecem quando o estudante satisfaz condições pré-estabelecidas.

Recompense realizações com emblemas - para estimular o esforço e a pontualidade pode-se utilizar emblemas que somente são obtidos caso o estudante complete um desafio até uma data limite.

Adicione uma pitada de humor e fantasia - ao nomear atividades, recomenda-se utilizar títulos que representem o conteúdo das mesmas e se possível, com uma pitada de humor para atrair a atenção dos estudantes. A figura 5 exibe algumas atividades com títulos que representam seu conteúdo e possuem humor.

Utilize alguma forma de narrativa – segundo Neto (2015) a narrativa é muito importante em um sistema que utilize gamificação, pois sem uma boa narrativa os componentes de jogos podem aparentar terem sido colocados sem um propósito claro.

Ao criar atividades e nomeá-las como missões, cria-se a expectativa de que a missão ofereça uma recompensa, o conteúdo da missão pode descrever além dos propósitos da atividade, uma pequena história e as recompensas que o estudante obterá ao completar a missão,

como pontos ou itens, dessa forma, a narrativa dá um propósito ao sistema de pontos e itens, que podem ser utilizados para desbloquear outras missões.

Lembre-se da teoria do fluxo - segundo o psicólogo Csikszentmihalyi (1990, apud ALVES, 2015, p.171), quando totalmente focados em um jogo ou brincadeira estamos em um estado chamado “flow” (fluxo).

A dificuldade da tarefa também determina se ela é divertida. Para manter o estado de fluxo, deve-se regular a dificuldade da atividade de forma que não fique fácil demais, tornando-se tediosa, nem difícil demais, gerando ansiedade. É por esse motivo que os jogos vão aumentando a dificuldade gradativamente. (ALVES, 2015)

Ao criar atividades deve-se aumentar a dificuldade das mesmas ao longo do tempo de forma a manter o estado de fluxo. Restringir as atividades por níveis de experiência deixa claro ao estudante que atividades que requerem o nível 2 são mais difíceis do que as que requerem o nível 1.

Instigue uma competição saudável - sistemas de ranqueamento são muito utilizados em gamificação, eles geram competição entre os usuários que disputam o 1º lugar, porém nem todas as pessoas são competitivas, algumas podem até não gostar de competições.

O *ranking* do *plug-in Level Up!* leva em consideração os pontos obtidos pelos estudantes. A forma de obtenção de pontos varia desde a abertura do curso a completar uma tarefa, ou seja, é um ranqueamento por acessos, pois desde que o estudante abra o curso, ele ganhará pontos. Isso pode causar um comportamento conhecido como *cheat* (fraude), pois se o estudante abrir o curso diversas vezes ele ganhará pontos e criará uma distância entre ele e os demais, sem que o mesmo tenha realmente completado tarefas que mereçam o recebimento destes pontos.

Como a obtenção de pontos pode ser livremente configurada, caso o estudante só ganhe pontos quando o acompanhamento de conclusão for ativado, ou seja, quando completar uma atividade, cria-se um *ranking* “justo”, pois não avalia quem acessa mais o ambiente, mas sim quem completa mais atividades. Quando os estudantes completam todas as atividades, eles compartilham o 1º lugar. O *ranking* também pode ser utilizado pelo professor como *feedback* para acompanhar os alunos que estão com dificuldades ou que não estão participando ativamente do curso.

Estimule a motivação intrínseca - segundo BUSARELLO (2016) as motivações intrínsecas originam-se do próprio sujeito, quando o indivíduo interage com as coisas por vontade própria, pois elas despertam interesse, desafio, envolvimento e prazer, incitando sua

curiosidade e busca por novos conhecimentos. A exploração, os desafios e a narrativa são algumas das formas para incentivar a motivação intrínseca.

É importante estabelecer metas que os jogadores devam alcançar e fazer com que elas sejam adotadas por eles. (BURKE, 2015)

Não é interessante deixar claro ao estudante que ele é obrigado a fazer as atividades, pode-se criá-las de forma que o mesmo não tenha muita pressão e instigá-lo a realizar as atividades com uma meta que possa ser alcançada dessa forma, por exemplo, a meta definida pelo professor é “chegar ao nível 10”, o estudante adota a meta e realiza todas as atividades para chegar ao nível 10.

Utilize o desbloqueio de conteúdos - a meta anterior pode ser alcançada se não existirem datas para entrega em nenhuma atividade (sem pressão), mas existirem restrições que impeçam a realização das atividades mais recentes, caso ele não complete as antigas (obrigação). Pode-se fazer isso utilizando as restrições por **nível** do *plug-in Level Availability*. O estudante pensará que, por exemplo, porque ele não é nível 2, ele não pode realizar as atividades mais avançadas, logo ele tem de “jogar” mais para subir de nível e ter acesso às atividades. Veja a figura 7 que demonstra uma atividade com restrição de nível.

4 INTEGRAÇÃO DO CURRÍCULO DE UM CURSO INTRODUTÓRIO DE LÓGICA DE PROGRAMAÇÃO COM O JOGO SÉRIO - COLOBOT

Contreras e Siu (2015) verificaram a importância de ferramentas que facilitassem o ensino de lógica de programação. Eles argumentam que programação deveria ser ensinada para estudantes de todas as áreas, já que a maioria dos produtos e processos atualmente são tecnológicos em natureza, e estudantes deveriam saber não somente como usar a tecnologia, mas também como criar tecnologia. Rahmat et al. (2012) nos diz que o ensino introdutório de linguagens de programação muitas vezes encontra barreiras à aprendizagem do aluno, e o engajamento é uma delas. Como pode ser verificado no estudo de Annetta et al. (2009) uma boa alternativa para resolver este problema é o uso de jogos, Kazimoglu et al. (2012) confirma isto ao obter resultados satisfatórios quando utiliza jogos no ensino de lógica de programação.

Um exemplo de um jogo sério para aprender programação é o jogo Colobot: Edição de Ouro. Muratet et al. (2009) em seu estudo sobre jogos sérios para ensino de programação observa que o jogo Colobot não era gratuito e existiam algumas dificuldades na construção de novos exercícios por professores. Posteriormente, em 2014, seu código foi aberto, e vários membros da comunidade o tornaram disponível com novos modos de jogo, assim sendo, os problemas destacados por Muratet foram resolvidos e o Colobot se tornou uma melhor opção para o ensino de habilidades de programação para os estudantes.

O Colobot é um jogo que foca na colonização de planetas com a ajuda de robôs que o jogador pode programar para realizar várias tarefas, e para isso, ele inclui uma linguagem de programação denominada CBOT, muito similar a C++. Ele também permite que seus usuários criem e customizem missões divididas em capítulos e níveis.

O propósito principal deste capítulo é apresentar a integração do jogo sério Colobot com a disciplina “Introdução a lógica de programação” do curso Técnico em Informática ministrado no campus Avançado Lagoa da Confusão do Instituto Federal de Educação, Ciência e Tecnologia do Tocantins. O propósito secundário é relatar que o jogo Colobot foi traduzido para o idioma Português do Brasil, o que é um requisito para seu uso de salas de aula brasileiras.

4.1 O jogo sério Colobot

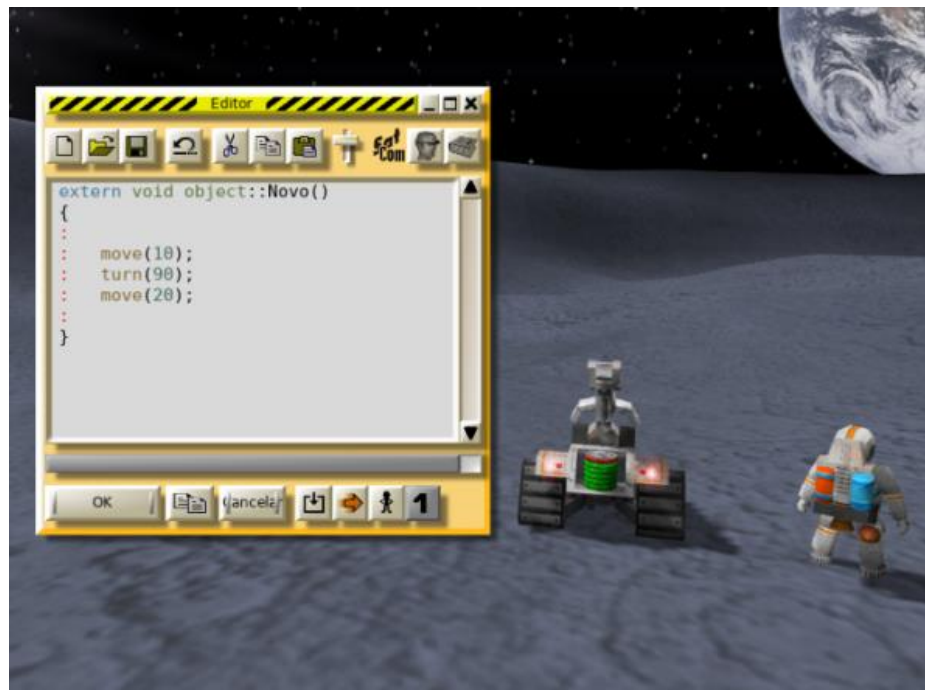
Segundo Laamarti, Eid e Saddik (2014, p. 4) um jogo sério é “uma aplicação com três componentes: experiência, entretenimento e multimídia”, através da experiência o jogador adquire conhecimento, habilidades ou outro conteúdo; entretenimento é o fator de imersão; e

multimídia, como texto, gráficos, animações e assim por diante, oferece oportunidades de aprendizado.

O jogo Colobot é um jogo sério para aprender uma linguagem de programação em sua concepção. Colobot fornece um ambiente de programação (Figura 11), possui um manual abrangente e acessível que ensina os conceitos básicos e apresenta em detalhes todas as funções disponíveis. Sua história principal tem como foco um astronauta procurando por um planeta habitável para a humanidade com a ajuda de robôs programáveis. Adicionalmente, o jogo também possui vários exercícios para ensinar o jogador a programar e entender suas mecânicas.

Deve-se enfatizar que não recomendamos os exercícios encontrados no jogo para o ensino introdutório de lógica de programação, porque eles não seguem um incremento gradual de dificuldade, nem contemplam todas as bases de um currículo introdutório, conseqüentemente, jogadores sem experiência em programação podem ter algumas dificuldades.

Figura 11 - Ambiente de programação do Colobot

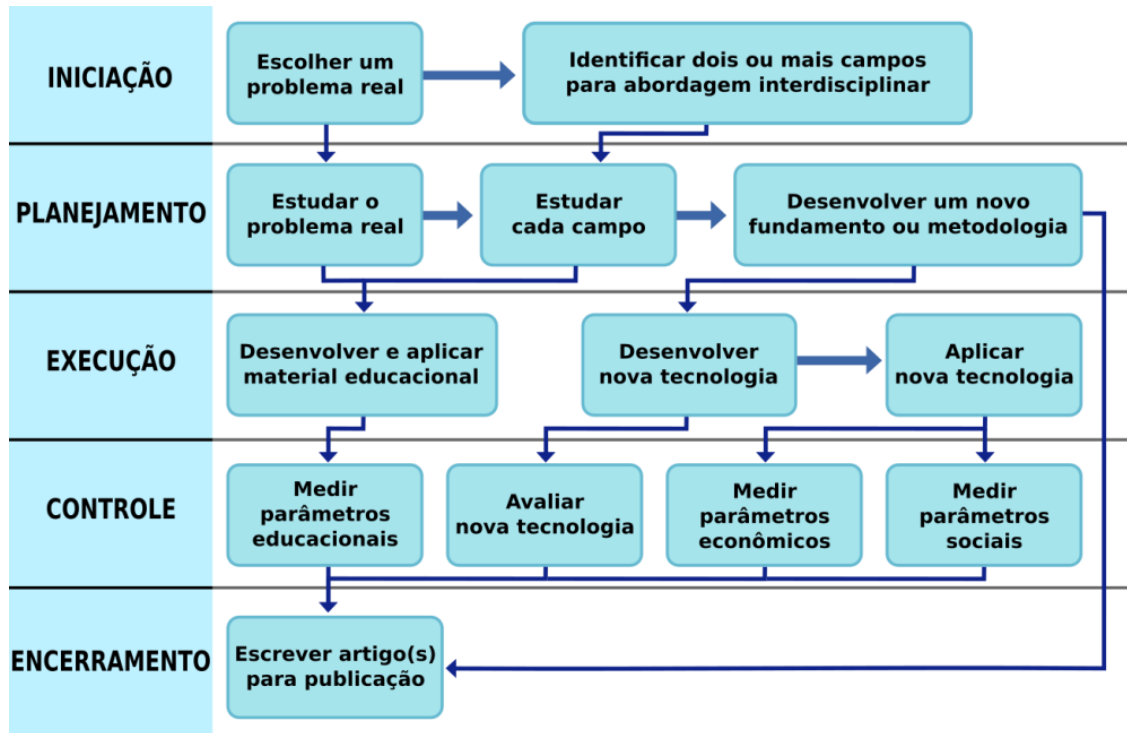


Fonte: Autor.

4.2 Gestão Interdisciplinar de Projetos de Pesquisa

A Gestão Interdisciplinar de Projetos de Pesquisa (Interdisciplinary Research Project Management – IRPM) é uma metodologia proposta por Letouze (2011) baseada em gerenciamento de projetos que tem como foco maximizar resultados acadêmicos (Figura 12).

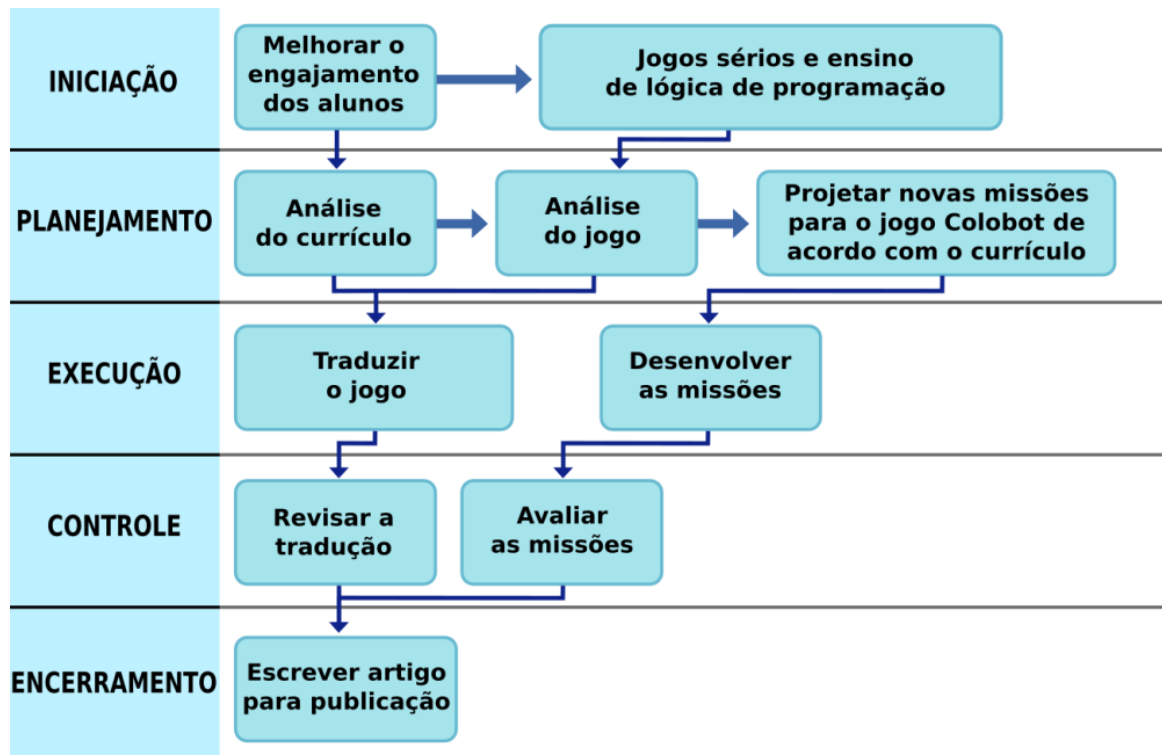
Figura 12 - Gestão Interdisciplinar de Projetos de Pesquisa



Fonte: Adaptado de Letouze (2011).

Apresentamos a aplicação do IRPM neste trabalho na Figura 13. O problema escolhido foi “melhorar o engajamento dos alunos” no estudo de lógica de programação, os campos escolhidos foram “jogos sérios” e “ensino de lógica de programação”, finalizando a fase de Iniciação do IRPM. Depois disso, temos a fase de Planejamento, aqui é realizada a análise do currículo da disciplina “Introdução a lógica de programação” do curso Técnico em Informática do Campus Avançado Lagoa da Confusão do IFTO. Na mesma fase houve o estudo sobre jogos sérios e ensino de lógica de programação, resultando na escolha do Colobot como o jogo sério a ser trabalhado. Neste momento, verificou-se que o Colobot não possuía uma versão em Português, um requisito para estudantes Brasileiros, porque eles não têm conhecimentos suficientes de outros idiomas, nem mesmo inglês. A fim de amenizar essa desvantagem, decidiu-se por traduzir o jogo Colobot para o idioma Português, com a exceção dos comandos da linguagem de programação, já que o uso de comandos em inglês irá familiarizar os alunos com futuras linguagens de programação.

Figura 13 - Aplicação do IRPM neste trabalho



Fonte: Autor.

Ainda na fase de Planejamento, temos a análise do jogo para o projeto de novas missões no Colobot com a integração curricular em mente. A tradução do jogo para Português é parte da fase de Execução, bem como o desenvolvimento das missões. A fase de Controle é responsável pela revisão da tradução e avaliação das missões. Na fase de encerramento tentamos publicar nossos resultados.

4.3 Integração Curricular

A abordagem utilizada para a integração curricular é baseada na experiência de Shapi'i e Ghulam (2016), que usam como fundamento teórico o trabalho de Freitas (2013). Eles afirmam que uma integração curricular de sucesso conecta os componentes do currículo de uma maneira significativa para os alunos, e para isso, eles fornecem três etapas:

1. Desenhar um curso baseado em componentes do currículo que podem se beneficiar de abordagens aprimoradas por jogos;
2. Gerar um ambiente de aprendizagem flexível;

3. Começar com tópicos que possam ser medidos com eficiência para que sejam expandidos posteriormente.

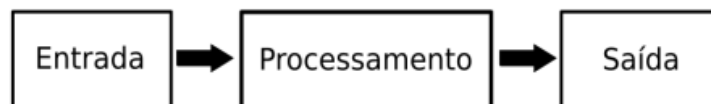
O ensino da lógica de programação gira em torno da criação de algoritmos para resolução de problemas, com a introdução incremental de novos conceitos para a construção de algoritmos cada vez mais complexos. Os instrutores usam a prática de escrever códigos como o meio principal para aumentar as taxas de aprendizado. O Colobot preserva essa prática, que é um recurso desejável para o envolvimento e a familiaridade do instrutor. Assim, neste caso, a integração curricular não altera a cultura de ensino dos instrutores. Até mesmo o design do jogo fornece a flexibilidade necessária para a introdução incremental de novos conceitos, ou seja, a cultura de ensino de lógica de programação é completamente preservada, enquanto aumenta o engajamento dos alunos no processo de aprendizagem.

4.4 As missões

Um algoritmo é uma lista ordenada de ações para resolver um problema e pode ser descrita pelo modelo Entrada-Processamento-Saída (Figura 14), também conhecido como modelo IPO (Input-Process-Output). Segundo Sentance, Barendsen e Schulte (2018):

- Entrada - são os dados obtidos do teclado, arquivos ou outra forma de entrada de dados;
- Processamento - é o processamento de entrada para gerar um resultado;
- Saída - é o resultado do processamento, pode ser impresso na tela, armazenado em um arquivo ou outra forma de saída.

Figura 14 - Modelo IPO



Fonte: Adaptado de Sentance, Barendsen e Schulte (2018).

Normalmente, no ensino introdutório da lógica de programação, a entrada é capturada do teclado para que o algoritmo possa processar esses dados e gerar a saída. No jogo Colobot, não há como o robô ler dados do teclado do usuário, de modo que, para obter um processo de entrada, serão usados postos de troca de informações (Figura 15). Sua finalidade é armazenar informações e fornecê-las ao robô quando solicitado. Todas as missões desenvolvidas usam

solicitações de informação à um posto de troca como "entrada", a lógica desenvolvida pelo aluno como "processo" e as ações executadas pelo robô como "saída", conforme o Quadro 1.

Quadro 1 - Categorias de ações para robôs no jogo Colobot

Ação	Objetivo
Movimento	Ir de um lugar para outro
Detecção	Detectar objetos próximos
Transporte	Transportar objetos
Luta	Destruir objetos ou se defender
Construção	Construir robôs ou edifícios
Comunicação	Receber ou enviar mensagens

Fonte: Autor.

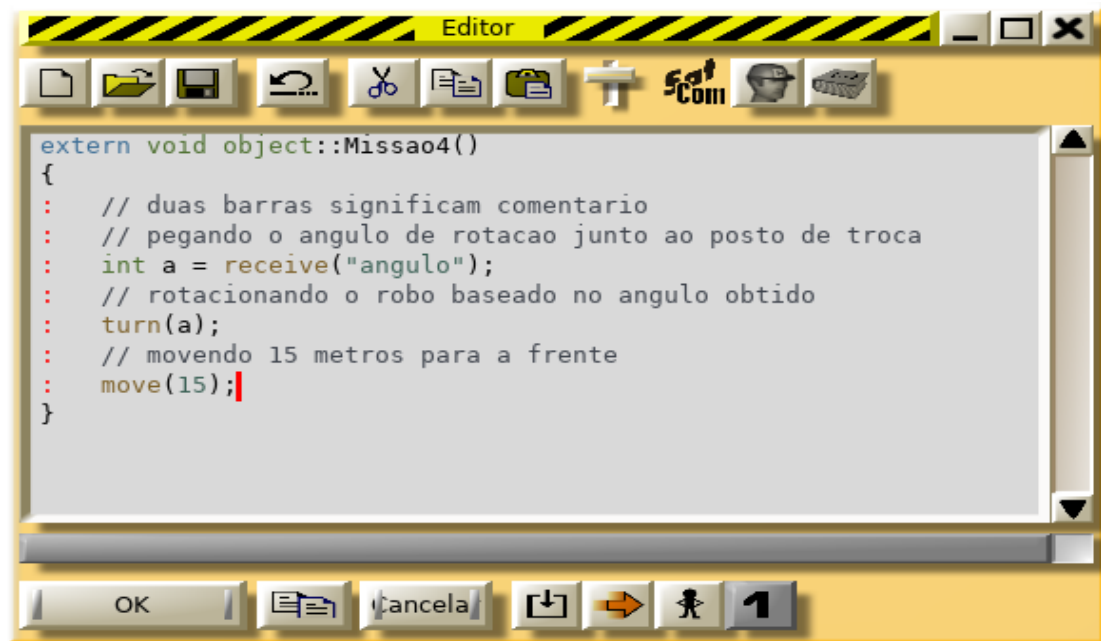
Assim, o processo de criação de missões começa com a escolha de um item do currículo, depois define-se o propósito da missão que irá, conseqüentemente, determinar a saída, e por último as entradas que serão fornecidas e as condições de vitória ou falha. Descrevemos este processo com uma missão simples que pode ser observada no Quadro 2, na Figura 15 apresentamos o código do processo, em seguida, na Figura 16, apresentamos uma imagem da missão.

Quadro 2 - Missão “Coleta de informações”

Propriedade	Detalhes
Nome da missão	Coleta de informações
Objetivo	Leve o robô até a cruz azul correta
Entrada	Ângulo de rotação
Processamento	Obter o ângulo de rotação junto ao posto de troca, girar o robô de acordo com o ângulo obtido e mover o robô 15 metros para a frente
Saída	O robô move em direção a cruz azul
Condições de vitória	O robô está em cima da cruz azul correta
Condições de falha	O robô está em cima de uma cruz azul incorreta

Fonte: Autor.

Figura 15 - Código de processamento para completar a missão “Coleta de informações”



```
extern void object::Missao4()
{
: // duas barras significam comentario
: // pegando o angulo de rotacao junto ao posto de troca
: int a = receive("angulo");
: // rotacionando o robo baseado no angulo obtido
: turn(a);
: // movendo 15 metros para a frente
: move(15);
}
```

Fonte: Autor.

Figura 16 - Missão “Coleta de informações”



Fonte: Autor.

4.5 A tradução

Um jogo é software e sua produção envolve várias áreas diferentes, desde a construção de histórias, design de personagens, desenvolvimento de arte, criação de som e programação de código, assim, para preparar a tradução de um software, é preciso ter em mente três fatores:

1. Localização;
2. Código-fonte;
3. Sintaxe.

Segundo Prudêncio, Valois e Lucca (2004), ao escrever software, o programador pode usar recursos para torná-lo multi-idioma, um processo denominado localização, o que pode ser conseguido através da separação de arquivos para cada idioma, externos ou internos ao código-fonte.

O código fonte é o código escrito pelos programadores, ou seja, o próprio software, pode ser fechado (acesso restrito) ou aberto (disponível para todos). Jogos comerciais são exemplos de software de código fechado, enquanto os jogos gratuitos podem ter seu código aberto.

A sintaxe se refere a expressões usadas na tradução para formatação ou estruturação do texto, geralmente são expressões em inglês que seguem um padrão e não devem ser traduzidas, pois perderiam o efeito.

Levando em conta a localização e o código-fonte, o Quadro 3 foi criado para demonstrar uma análise da "traduzibilidade" de um jogo, ou seja, a condição de uma possível tradução.

Quadro 3 - Traduzibilidade de um jogo

Localização	Código-fonte	Tradução
Não	Fechado	Não é possível
Não	Aberto	Muito difícil
Sim	Fechado	Possível
Sim	Aberto	Fácil

Fonte: Autor.

Se o software não implementa nenhum processo de localização e seu código-fonte é fechado, diz-se que a tradução não é possível porque não há arquivos de idioma ou acesso ao código para modificação.

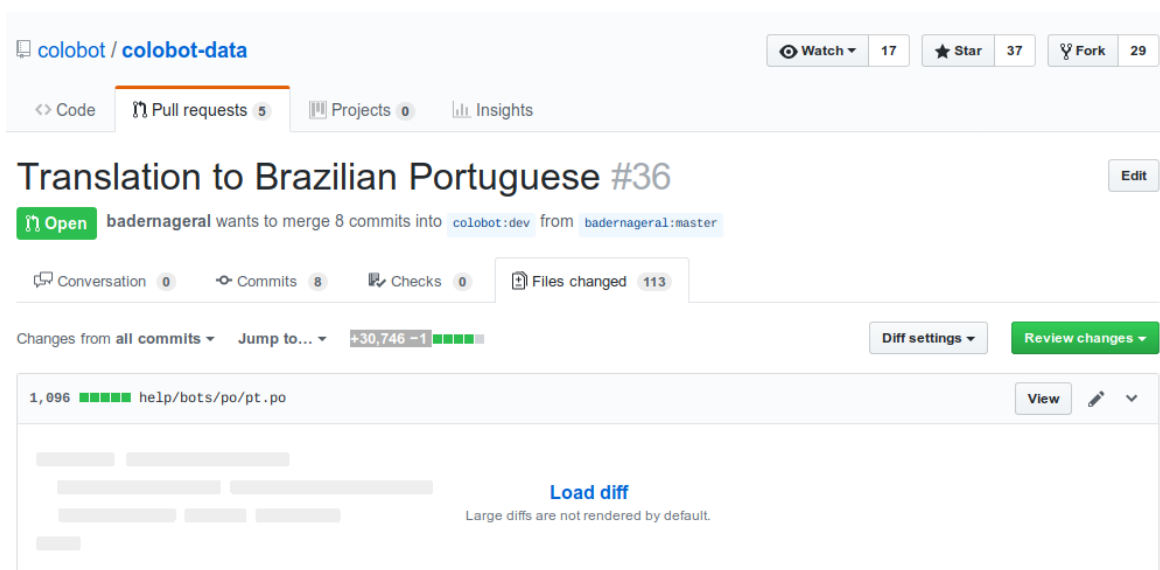
O fato de o software não usar nenhum processo de localização indica que ele não aceita vários idiomas, portanto, a única maneira de traduzir seria alterar o código-fonte e substituir o idioma original pelo idioma de destino ou implementar um processo de localização. Este seria um processo árduo, exigindo grandes habilidades de programação.

Quando o software implementa algum processo de localização, deve-se investigar como esse processo foi implementado. Alguns softwares usam arquivos externos, enquanto outros usam arquivos internos. Os arquivos internos estão dentro do código-fonte, portanto, se o código-fonte estiver fechado, não há como acessar esses arquivos. Normalmente, o software de código fechado usa arquivos externos para permitir a tradução sem acesso ao código-fonte, enquanto o software de código aberto usa ambas as abordagens.

O jogo Colobot implementa um processo de localização usando arquivos internos e possui código fonte aberto. A primeira etapa da tradução foi obter o código-fonte do jogo e a documentação da sintaxe usada nos arquivos de tradução que a equipe do projeto disponibiliza em seu site. Observe que, neste caso, é necessário um conhecimento básico de programação, pois é necessário alterar o código-fonte seguindo a sintaxe obtida para realizar a tradução.

A tradução do Colobot envolveu a adição de aproximadamente 30.000 novas linhas ao código fonte do projeto (Figura 17), em que pelo menos 30% dessas linhas correspondem a sentenças em português, com o resto das linhas representando quebras de linha ou identificadores. A Figura 18 apresenta uma frase traduzida, em que a primeira linha contém o identificador da frase, ou seja, a mensagem original em inglês e a segunda a tradução.

Figura 17 – *Pull Request* da tradução do Colobot para o idioma português



Fonte: Autor.

Figura 18 - Exemplo de tradução

```

1 msgid "Try to figure out how to survive in a hostile environment."
2 msgstr "Tente descobrir como sobreviver em um ambiente hostil."

```

Fonte: Autor.

4.6 Resultados e discussão

O currículo da disciplina foi analisado e dividido em cinco grupos de componentes que agrupam tópicos semelhantes. Como o Colobot permite a criação de uma hierarquia entre capítulos e missões, cada grupo é representado por um capítulo e cada capítulo tem várias missões especializadas, que são apresentadas na Tabela 1.

Tabela 1 - Componentes curriculares e seus capítulos

Componente do currículo	Capítulo	Número de missões
Variáveis	I – Primeiros passos	9
Estruturas de decisão	II – Tomando decisões	5
Estruturas de repetição	III – Hora de repetir	7
Matrizes	IV – Agrupando	5
Funções	V – Minhas instruções	4

Fonte: Autor.

O primeiro capítulo (I) trata da introdução da linguagem de programação através do uso de variáveis e exercícios variados que introduzem as ações de um robô para familiarizar o aluno com o processo de programação. O segundo capítulo (II) acrescenta à tomada de decisão, o terceiro capítulo (III) o conceito de repetição, o quarto capítulo (IV) a capacidade de agrupar informações e o último capítulo (V) a criação de funções personalizadas conforme o Quadro 4.

Quadro 4 - Objetivos das missões integradas no currículo

Capítulo	Missões	Objetivo
I	1, 2 e 3	Introduzir o robô, as mecânicas do jogo e o conceito de processamento e saída
I	4	Introduzir o conceito de entrada
I	5 e 6	Trabalhar com coordenadas (plano cartesiano)
I	7	Uso do radar para detecção de objetos
I	8	Aprender como transportar objetos
I	9	Aprender como atirar para destruir objetos
II	1 e 2	Tomar decisões e trabalhar com direções cardinais
II	3	Decisões com direções intercardinais
II	4	Decisões com múltiplas condições
II	5	Encadeamento de decisões
III	1	Aprender como repetir
III	2	Trabalhar com laços infinitos
III	3 e 4	Trabalhar com laços e detecção de objetos
III	5, 6 e 7	Trabalhar com laços e destruição de objetos
IV	1 e 2	Trabalhar com matrizes e detecção de objetos
IV	3 e 4	Trabalhar com matrizes e destruição de objetos
IV	5	Trabalhar com matrizes multidimensionais
V	1	Aprender como criar funções
V	2	Criar funções para destruição de objetos
V	3	Criar funções para trabalhar com direções cardinais
V	4	Otimização de código através da criação de funções

Fonte: Autor.

As missões estão disponíveis em um repositório GitHub (Figura 19) para que qualquer um possa baixar e utilizá-las no jogo. O objetivo secundário deste capítulo foi fornecer uma versão do Colobot em Português; a tradução do jogo para o português do Brasil foi concluída com sucesso e pode ser obtida no repositório oficial do jogo. As missões são disponibilizadas no idioma Inglês e Português do Brasil, a língua do jogo irá definir o idioma das missões.

Figura 19 - Repositório das missões no GitHub

The screenshot shows the GitHub repository page for 'badernageral / colobot-logic'. The repository description is 'A series of colobot missions for teach introductory programming logic'. It has 7 commits, 1 branch, 0 releases, 1 contributor, and is licensed under GPL-3.0. The latest commit is 'Update README.md' by badernageral, 3 hours ago. The repository contains the following files:

File Name	Commit Message	Time Ago
missions	Versão 1.0	3 hours ago
LICENSE	Initial commit	8 days ago
README.md	Update README.md	3 hours ago
screenshot.png	Screenshot	3 hours ago

Fonte: Autor.

As missões foram desenhadas com uma abordagem de dificuldade incremental, ou seja, a complexidade aumentou pouco a pouco, a fim de proporcionar o tempo necessário para o aluno se adaptar. Para demonstrar a quantidade de trabalho necessária para projetar e desenvolver as missões em inglês e português, apresentamos o status das missões no GitHub (Figura 20), foram produzidas 4.010 linhas de código divididas em 117 arquivos, adicionalmente, na Figura 21, apresentamos a tela do Colobot em Português com as missões.

Figura 20 - Status do repositório das missões

The screenshot shows the GitHub repository page for 'badernageral / colobot-logic' with the 'Insights' section selected. The overview for the period 'May 24, 2018 – May 31, 2018' shows:

- 0 Active Pull Requests
- 0 Active Issues
- 0 Merged Pull Requests
- 0 Proposed Pull Requests
- 0 Closed Issues
- 0 New Issues

Excluding merges, 1 author has pushed 4 commits to master and 4 commits to all branches. On master, 117 files have changed and there have been 4,010 additions and 3 deletions.

Fonte: Autor.

Figura 21 - Missões em português no Colobot

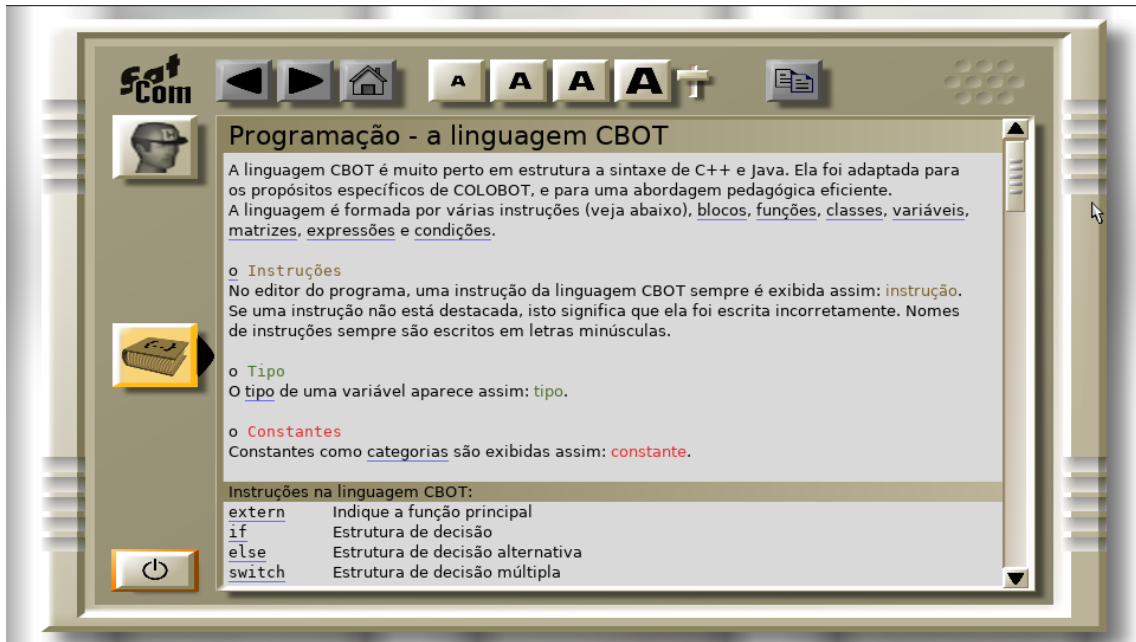


Fonte: Autor.

Encontramos um problema ao tentar criar missões mais dinâmicas: a impossibilidade de criar valores aleatórios, como posições de objetos ou condições de vitória. Isso poderia ser resolvido modificando a forma como o jogo cria as missões, porém a modificação das mecânicas do jogo não foi uma tarefa realizada neste trabalho, devido à sua complexidade e ao escopo deste trabalho.

Sobre a tradução, a contribuição mais significativa é a tradução do manual do jogo (Figura 22), que fará uma grande diferença quando o jogo for usado em salas de aula que usam o português como língua principal. O *Pull Request* (Figura 17), ou seja, a solicitação de aceitação da tradução enviada para o repositório oficial do jogo foi aceita, de forma que o próxima versão a ser disponibilizada do jogo já virá com o idioma português do Brasil.

Figura 22 - Manual do Colobot em português



Fonte: Autor.

5 CONSIDERAÇÕES FINAIS

Nos últimos anos a gamificação passou de um tópico de pesquisa para um próspero campo interdisciplinar, enquanto as primeiras pesquisas careciam de base teórica, os trabalhos atuais estão mais maduros, focando-se em compreender mais profundamente a gamificação, porém ainda há escassez de estudos mais rigorosos avaliando os diferentes métodos propostos. (KOIVISTO; HAMARI, 2017)

A gamificação cresce a cada dia e conhecer seus aspectos positivos, bem como os negativos, é importante para que a mesma seja tratada com cuidado. O trabalho de Pink (2010) é muito importante para a compreensão do poder da motivação intrínseca, enquanto o trabalho de Kapp (2012) e Marczewski (2015) ajudar a planejar a implementação de uma solução gamificada, não esquecendo o trabalho de Burke (2015) que cita diversos casos de uso que podem clarificar o uso da gamificação.

Um bom programa de gamificação utiliza os elementos de jogos como parte da estratégia para alcançar seus objetivos e não os próprios elementos como a estratégia em si, desta forma a aplicação da gamificação, principalmente na educação, não se deve esquecer de levar os fatores do ambiente, os objetivos a serem alcançados e os jogadores em consideração.

Os desafios e dinâmicas que a gamificação oferece ao estudante incentivam a aprendizagem e proporcionam pensamento lógico, o que a torna uma boa adição a “caixa de ferramentas” do professor, como também demonstra a mudança que o papel do professor sofreu nas últimas décadas com o advento das novas tecnologias, conforme o pensamento de Lévy na década de 90:

“A principal função do professor não pode mais ser uma difusão dos conhecimentos, que agora é feita de forma mais eficaz por outros meios. Sua competência deve deslocar-se no sentido de incentivar a aprendizagem e o pensamento”. (LÉVY, 1999, p.171)

Segundo Fardo (2013) a gamificação na educação é um fenômeno que merece atenção, pois a popularidade dos jogos só aumenta e conseqüentemente torna seu uso mais comum a cada dia, porém ressalta que ela não é a solução para todos os problemas da educação.

Apesar da gamificação contribuir para o engajamento dos estudantes, é preciso ter cuidado e não exagerar em sua utilização, por exemplo, muitas restrições, narrativas extensas, desafios impossíveis, entre outros. O Moodle oferece diversas possibilidades para o uso de gamificação que dependem principalmente do esforço e da imaginação do professor, pode ser

um pouco desgastante no início, mas os resultados são compensadores enquanto que o esforço e a diversão caminham juntos com o processo.

As missões desenvolvidas para o Colobot podem ser usadas em um curso de 'Introdução à Lógica de Programação', tanto em inglês, quanto em português, por causa da tradução do jogo para o português. Esse resultado foi obtido a partir da integração do currículo baseada na experiência de Shapi'i e Ghulam (2016), o modelo IPO e a estratégia de gerenciamento de projetos de pesquisa interdisciplinar proposta por Letouze (2011).

O jogo Colobot mostra um grande potencial para o ensino de lógica de programação. A integração curricular feita a partir da criação das missões abre um leque de possibilidades, pois permite, além do uso em sala de aula, que outros professores as utilizem como base para produzir suas próprias missões, por exemplo. Essas missões podem ser usadas como base para o desenvolvimento de missões interdisciplinares como as apresentadas por Batista et al. (2017) que utilizou o jogo Colobot para o ensino de geometria, também utilizando o IRPM.

Sugerimos uma análise baseada no estudo de Freitas e Oliver (2006) como trabalho futuro, pois não foi realizada uma análise formal da efetividade dessas missões para engajamento e aprendizagem dos alunos.

Os objetivos foram alcançados com sucesso com a elaboração das estratégias de gamificação e o desenvolvimento das missões, de forma que a utilização conjunta de gamificação e jogos sérios contribua para o aumento do engajamento de alunos em disciplinas de lógica de programação.

REFERÊNCIAS

- ALBINO, João Pedro et al. **Gamification em Ambientes Virtuais de Aprendizagem no Ensino Superior Presencial**. In: Congresso Internacional de Educação e Tecnologias: Encontro de Pesquisadores em Educação a Distância, 2018.
- ALMEIDA, Cíntia et al. **Avaliação do processo de Gamificação acerca do tema Direitos Humanos**. In: Anais do Workshop de Informática na Escola. 2016. p. 379.
- ALVES, Flora. **Gamification**: como criar experiências de aprendizagem engajadoras. Um guia completo do conceito a prática. 2º ed. São Paulo: DVS, 2015. 209 p.
- ANNETTA, Leonard A. et al. **Investigating the impact of video games on high school students' engagement and learning about genetics**. Computers & Education, v. 53, n. 1, p. 74-85, 2009.
- BARTLE, Richard. **Hearts, clubs, diamonds, spades: Players who suit MUDs**. Journal of MUD research, v. 1, n. 1, p. 19, 1996.
- BATISTA, Rosivânia Batista; FERRAZ, Elvis Ribeiro; LETOUZE, Patrick. **Uma Abordagem Interdisciplinar para o Ensino de Geometria com o Serious Game 3D Colobot**. Proceedings of the XVI SBGAMES – Simpósio Brasileiro de Games, 2017.
- BURKE, Brian. **Gamificar**: como a gamificação motiva as pessoas a fazerem coisas extraordinárias. São Paulo: DVS, 2015. 192 p.
- BUSARELLO, Raul Inácio. **Gamification**: princípios e estratégias. São Paulo: Pimenta Cultural, 2016.
- CONTRERAS, Giovanni J.; SIU, Kin Wai Michael. **Computer programming for all: A case-study in product design education**. Procedia-Social and Behavioral Sciences, v. 182, p. 388-394, 2015.
- DE FREITAS, Sara; OLIVER, Martin. **How can exploratory learning with games and simulations within the curriculum be most effectively evaluated?**. Computers & education, v. 46, n. 3, p. 249-264, 2006.
- DE FREITAS, Sara; OTT, Michela. **New pedagogical approaches in game enhanced learning: Curriculum integration**. IGI Global, 2013. 310 p.
- DE FREITAS, Sérgio et al. **Gamificação e avaliação do engajamento dos estudantes em uma disciplina técnica de curso de graduação**. In: Simpósio Brasileiro de Informática na Educação - SBIE. 2016. p. 370.
- DENMEADE, Natalie. **Gamification with Moodle**. Reino Unido: Packt, 2015. 134 p.

- DETERDING, Sebastian et al. **Gamification: Toward a Definition**. In: Proceedings of the 2011 ACM Conference on Human Factors in Computing Systems (CHI 2011), Vancouver, BC, Canadá, p. 12-15, Mai. 2011.
- FARDO, Marcelo Luis. **A Gamificação aplicada em Ambientes de Aprendizagem**. Revista Novas Tecnologias na Educação, Porto Alegre, v. 11, n. 1, 2013.
- HENRICK, Gavin; HOLLAND, Karen. **Moodle Administration Essentials**. Reino Unido: Packt, 2015. 156 p.
- KAPP, Karl M. **The gamification of learning and instructions: game based methods and strategies for training and education**. Estados unidos: Pfeiffer, 2012. 336 p.
- KAZIMOGLU, Cagin et al. **A serious game for developing computational thinking and learning introductory computer programming**. Procedia-Social and Behavioural Sciences, v. 47, p. 1991-1999, 2012.
- KOIVISTO, Jonna; HAMARI, Juho. **The rise of motivational information systems: A review of gamification research**. Working paper, 2017.
- LAAMARTI, Fedwa; EID, Mohamad; SADDIK, Abdulmotaleb El. **An overview of serious games**. International Journal of Computer Games Technology, v. 2014, p. 11, 2014.
- LETOUZE, Patrick. **Interdisciplinary research project management**. International Proceedings of Economics Development and Research, v. 14, p. 338-342, 2011.
- LÉVY, Pierre. **Cibercultura**. São Paulo: 34, 1999. 264 p.
- LINEHAN, Conor; KIRMAN, Ben; ROCHE, Bryan. **Gamification as behavioral psychology**. In: The gameful world: Approaches, issues, applications. MIT Press, 2015.
- MARCZEWSKI, Andrzej. (2015). **Even Ninja Monkeys Like to Play: Gamification, Game Thinking and Motivational**. CreateSpace Independent Publishing Platform, 2015, 220 p.
- MURATET, Mathieu et al. **Towards a serious game to help students learn computer programming**. International Journal of Computer Games Technology, v. 2009, p. 3, 2009.
- NETO, Henrique Ruiz Poyatos. **Gamificação: engajando pessoas de maneira lúdica**. São Paulo: Fiap, 2015. 98 p.
- PINK, Daniel. **Motivação 3.0: os novos fatores motivacionais para a realização pessoal e profissional**. Brasil: Campus, 2010. 240 p.
- PRUDÊNCIO, Achilles Colombo; VALOIS, Djali Avelino; DE LUCCA, José Eduardo. **Introdução à internacionalização e à localização de softwares**. Cadernos de tradução, v. 2, n. 14, p. 211-242, 2004.
- RAHMAT, Masura et al. **Major problems in basic programming that influence student performance**. Procedia-Social and Behavioral Sciences, v. 59, p. 287-296, 2012.

SENTANCE, Sue; BARENDSSEN, Erik; SCHULTE, Carsten. **Computer Science Education: Perspectives on Teaching and Learning in School**. Bloomsbury Academic, 2018.

SHAPI'I, Azrulhizam; GHULAM, Sychol. **Model for educational game using natural user interface**. International Journal of Computer Games Technology, v. 2016, 2016.

SILVA, Rodrigo Ribeiro; FERNANDES, Juliana; SANTOS, Rodrigo. Panorama da Utilização de Jogos Digitais no Ensino de Programação no Nível Superior na Última Década: Uma Revisão Sistemática da Literatura. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. 2018. p. 535.

SKINNER, Burrhus Frederic. **Ciência e comportamento humano**. 11º ed. São Paulo: Martins Fontes, (1953), 2003. 489p.

TODA, Armando M.; VALLE, Pedro H. D.; ISOTANI, Seiji. **The Dark Side of Gamification: An Overview of Negative Effects of Gamification in Education**. In: Researcher Links Workshop: Higher Education for All. Springer, Cham, 2017. p. 143-156.

WERBACH, Kevin; HUNTER, Dan. **The Gamification Toolkit: Dynamics, Mechanics, and Components for the Win**. Estados unidos: Wharton Digital Press, 2015, 47 p.

APÊNDICES

**APÊNDICE A – THE CURRICULUM INTEGRATION OF A COURSE OF
'INTRODUCTION TO PROGRAMMING LOGIC' WITH A SERIOUS GAME -
COLOBOT**

The Curriculum Integration of a course of 'Introduction to Programming Logic' with a Serious Game - Colobot

José R. M. Alves and Patrick Letouze

Abstract— Programming logic is a knowledge and skill that find applications in daily lives, since nowadays most products, processes and systems are intrinsically technological. Therefore, it is reasonable to argue in favour of teaching it to all students, especially high school students. An engaging medium to accomplish that would be the use of a serious game to facilitate the teaching of programming logic. It is desirable that the learning process happens in a structured way respecting the traditional approach to teaching programming logic, which should avoid the resistance of instructors. If it is taken into consideration that, the Curriculum Integration is a success factor for educational games acceptance by teachers and its subsequent adoption in class, then it provides the desired structured way to teach programming logic with a serious game. Hence, this work presents a Curriculum integration of the serious game Colobot Gold Edition with the discipline of 'Introduction to Programming Logic' for the level of technician. This is obtained through the combination of the Interdisciplinary Research Project Management strategy with the Input-Process-Output model, keeping focus on Curriculum integration. The developed curriculum integrated missions provide an alternative for educating all high school students in programming logic with an engaging game, besides that a translation to Portuguese of Colobot was produced allowing Portuguese speaking students to use their native language to learn programming logic in a “fun” way.

Index Terms—Serious Games, Programming Logic, Colobot, Education.

I. INTRODUCTION

Contreras and Siu in [1] verified the importance of tools that facilitate the teaching of programming logic. They argued that programming should be taught to students of all areas, since most products and processes are currently technological in nature, and students should not only know how to use technology, but also how to create technology. Introductory teaching of programming languages often encounters major barriers to student learning, and engagement is one such barrier [2]. A good alternative to solve this problem is the use of games [3], and particularly, they have already been used satisfactorily in teaching programming logic [4].

An example of a serious game for learning programming is the game Colobot: Gold Edition [5]. In 2009, a study conducted by Muratet et al.[6] about serious game for programming remarked that the game Colobot was not free

and there existed some difficulties in building new exercises by teachers. However, in 2014, its source code became open, and many members of the community made it available, including new game modes [5]. Therefore, the problems highlighted by Muratet et al. in [6] were solved, and Colobot has improved as an option to improve students' programming skills.

Colobot is a game that focuses on the colonization of planets with the help of robots that the player can program to perform various tasks, and for that, it is using a programming language called CBOT, very similar to C++. It also allows its users to create and to custom missions divided into chapters and levels.

The main purpose of our work is to present the Curriculum integration of the serious game Colobot Gold Edition with the discipline of “Introduction to Programming Logic” for the level of technician to be ministered at the Instituto Federal de Educação, Ciência e Tecnologia do Tocantins (IFTO - Federal Institute of Tocantins), campus Lagoa da Confusão, Brazil. The secondary purpose is to report that we have translated the game Colobot to Portuguese to be available in the next edition [7], which is a requirement for its use in Brazilian classrooms.

For the organization of this paper, we chose the following structure: Introduction, Material and Methods, Results and Discussion, Conclusions. In the Material and Methods section, we presented Colobot Gold Edition as the material used in our work; the Interdisciplinary Research Project Management (IRPM) as our research strategy method [8]; the method for Curriculum integration that was based on the experience of Shapi'i and Ghulam in [9]; and we used the Input-Process-Output model [10] for the development of Missions in Colobot. Then we present the results and we discuss them in Section III. Finally, we present our conclusion in Section IV.

II. MATERIAL AND METHODS

A. *Material: the serious game Colobot*

A serious game is “an application with three components: experience, entertainment, and multimedia” [11], where through experience the player acquires knowledge, skills and other contents; entertainment is the immersion factor; and multimedia such as text, graphics, animations, and so on provides learning opportunities.

The game Colobot is a serious game for learning

programming language in its conception. Colobot provides a programming environment, as in Fig. 1. It has a large and accessible manual within that teaches basic concepts and presents in details all available functions. Its main story focuses on an astronaut looking for a habitable planet for humanity and with the help of programmable robots. Additionally, the game has several exercises to teach the player to program and understand their mechanics.

It should be emphasized that we do not recommend the exercises found in the game for introductory teaching of programming logic, because they do not follow a gradual increase of difficulty, nor do they contemplate all the bases of an introductory curriculum. Consequently, players with no programming experience may have some difficulties.

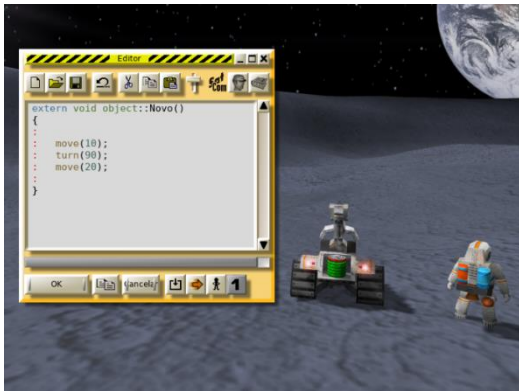


Fig. 1. An example of the programming environment of Colobot.

B. Method: the Interdisciplinary Research Project Management

The Interdisciplinary Research Project Management (IRPM) [8] is a methodology for developing interdisciplinary research based on project management [12] that intends to maximize academic results as shown in Fig. 2.

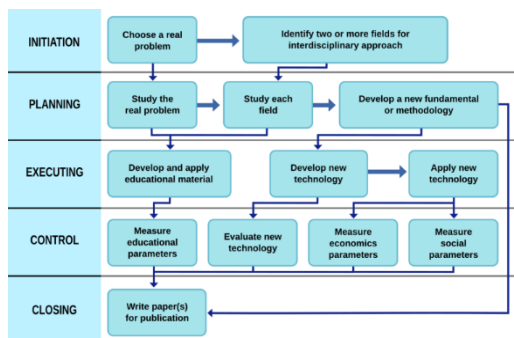


Fig. 2. Interdisciplinary Research Project Management.

We present the application of IRPM in our work in Fig. 3. We begin by choosing the problem of “enhancing students’ engagement in studying programming logic”, and we choose two fields, “serious games” and “teaching programming logic”, ending the Initiation phase of IRPM. Following that, we have the Planning phase, where we start with analyzing the curriculum of the discipline “Introduction of Programming Logic” of the technician course of the advanced campus of Lagoa da Confusão of IFTO, Brazil. Then we studied “serious games” and “teaching

programming logic”, resulting in the choice of Colobot (Gold Edition) as the serious game. Also at this moment, we verified that Colobot did not have a version in Portuguese, a requirement for Brazilian students, because they do not have enough knowledge of other languages, not even English. In order to soften this drawback, we decided to translate Colobot to Portuguese, with the exception of the programming language commands, since the use of English commands will familiarize them with future programming languages.

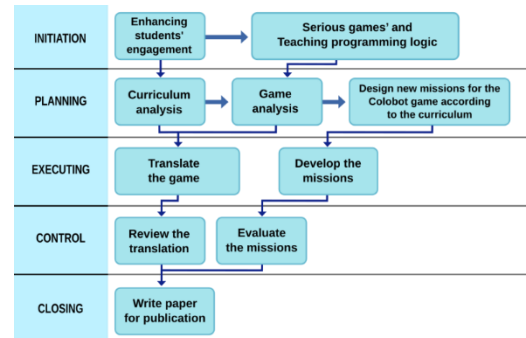


Fig. 3. The application of IRPM to Curriculum integration of ‘Introduction of Programming Logic’.

Still in Planning phase, we have the game analysis for designing new missions in Colobot with Curriculum integration in mind. The translation of the game to Portuguese is part of the Execution phase as much as the development of missions in Colobot. Control phase is responsible for the translation review and for mission’s evaluation. In the Closing phase, we attempt to publish our results.

C. Method: Curriculum Integration

Our approach to Curriculum integration is based on the experience of Shapi’i and Ghulam [9], who used as theoretical foundation the work of Freitas in [13]. Shapi’i and Ghulam state that a successful Curriculum integration connects curriculum components in a meaningful way to students [9], and for that, they provide three steps:

- 1) To design a course based on components of the curriculum that may benefit from game-enhanced approaches;
- 2) To generate an environment of flexible learning;
- 3) To start with topics that can be efficiently measured for latter to expand them.

The teaching of programming logic revolves around the creation of algorithms for problem solving, with the incremental introduction of new concepts for the construction of increasingly complex algorithms. Instructors use the practice of code writing as the main medium to increase learning rates. Colobot preserves this practice, which is a desirable feature for instructor’s engagement and familiarity. Hence, in this case, the curriculum integration does not change the teaching culture of instructors. Even the game design provides the necessary flexibility for incremental introduction of new concepts. That is, the culture of teaching programming logic is completely preserved, while increases students’ engagement in the learning process.

D. Method: the Missions

An algorithm is an ordered list of actions to solve a

problem and may be described by the Input-Process-Output model, Fig. 4, also known as IPO model [10]: Input - is the data obtained from the keyboard, files or other form of data entry; Processing - is the input processing in order to generate a result; Output - is the result of processing, may be printed on the screen, stored in a file or otherwise output.

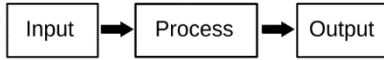


Fig. 4. IPO Model.

Normally, in the introductory teaching of programming logic, the input is captured from the keyboard so that the algorithm can process this data and generate the output. In the Colobot game there is no way for the robot to read data from the user's keyboard, so in order to get an input process, information exchange posts will be used, as in Fig. 5. Its purpose is to store information and provide it to the robot when requested. All developed missions use information requests from an exchange post as 'input', the logic developed by the student as 'process' and the actions performed by the robot as 'output' are presented in Table I.

TABLE I: CATEGORIES OF ACTIONS FOR ROBOTS IN THE GAME COLOBOT

Action	Objective
Movement	Go from one place to another
Detection	Detect nearby objects
Transport	Object transportation
Fight	Destroy objects or defend itself
Construction	Construct robots or buildings
Communication	Receive or send messages

Hence, the process of creating missions begins with the choice of an item of the curriculum. After that, we define the purpose for the mission that will consequently determine the output, and then we define the inputs that will be provided and the conditions of victory or failure. We describe this process with a simple mission in Table II and in Fig. 5 we present the process code, then in Fig. 6 we present a picture of the mission.

TABLE II: MISSION 'INFORMATION GATHERING'

Property	Details
Mission name	Information gathering
Objective	Take the robot to the correct waypoint
Input	Rotation angle
Process	Obtain the angle of rotation from the information exchange post, rotate the robot based on the obtained angle, move 15 meters forward
Output	The robot moves toward the waypoint
Victory condition	The robot is on top of the correct waypoint
Fail condition	The robot is on top of an incorrect waypoint

```

extern void object::Mission4()
{
  //two slashes mean a comment
  //getting the angle of rotation from the exchange post
  int a = receive("angle");
  //rotate the robot based on the obtained angle
  turn(a);
  //move 15 meters forward
  move(15);
}
  
```

Fig. 5. Process code to complete mission 'Information gathering'.

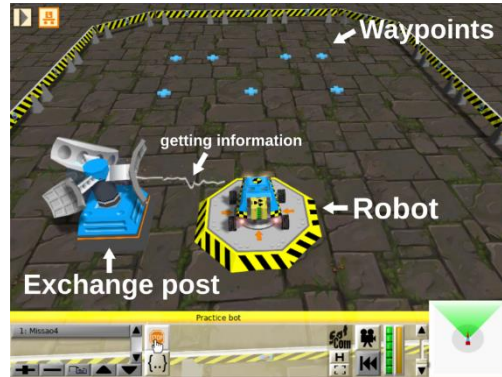


Fig. 6. Mission 'Information gathering'.

E. Method: the Translation

A game is software and its production involves several different areas, from story building, character design, art development, sound creation and code programming. Thus, to prepare the translation of software, one must keep in mind three factors:

- 1) Localization;
- 2) Source code;
- 3) Syntax.

When writing software, the programmer can use resources to make it multi-language, a process known as localization [14], which can be achieved by separating files for each language, external or internal to the source code.

The source code is the code written by the programmers, ie the software itself, it can be closed (restricted access) or open (available to all). Commercial games are examples of closed source software, while free games may have their code open.

The syntax refers to expressions used in translation for formatting or structuring the text, usually they are expressions in English that follow a pattern and should not be translated as they would lose their effect.

Taking into account the localization and the source code, Table I was created to demonstrate an analysis of the "translatability" of a game, that is, the condition of a possible translation.

TABLE III: TRANSLATABILITY OF A GAME

Localization	Source code	Translation
Not	Closed	Not possible
Not	Open	Very difficult
Yes	Closed	Possible
Yes	Open	Easy

If the software does not implement any localization process and its source code is closed, it is said that the translation is not possible because there are no language files or access to the code for modification.

The fact that the software does not use any localization process indicates that it does not accept multiple languages, so the only way to translate would be to change the source code and replace the original language with the target language or implement a localization process. This would be an arduous process, requiring great programming skills.

When the software implements some localization process, one should investigate how this process was implemented. Some software use external files, while others use internal files. Internal files are inside the source code, so if the source code is closed, there is no way to access these files. Typically closed source software uses external files to enable translation without source code access, while open source software uses both approaches.

The game Colobot implements a localization process using internal files and has open source code [15]. The first step in translating was to get the source code for the game and the documentation on the syntax used in the translation files that the project team makes available on their site. Note that in this case, a basic programming knowledge is required, since it is necessary to change the source code following the syntax obtained to perform the translation.

The Colobot translation involved the addition of approximately 30,000 new lines to the project source code (see Fig. 7), where at least 30% of these lines correspond to sentences in Portuguese, with the rest of the lines representing line breaks or identifiers. Fig. 8 represents a translated sentence, where the first line contains the identifier for the phrase, that is, the original message in English and the second the translation.

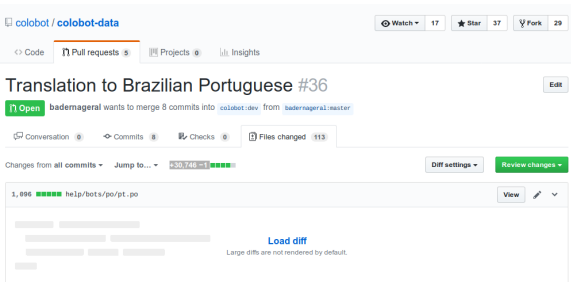


Fig. 7. The translation of Colobot Gold Edition to Portuguese.

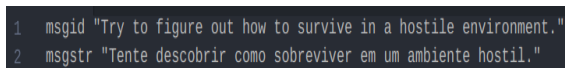


Fig. 8. Translation example.

III. RESULTS AND DISCUSSION

The curriculum of the discipline was analyzed and divided into five clusters of components that group similar topics. Since Colobot allows the creation of hierarchy between chapters and missions, each cluster is represented by a

chapter and each chapter has several specialized missions, which are presented in Table IV.

TABLE IV: CURRICULUM COMPONENTS AND THEIR CHAPTERS

Curriculum components	Chapters	Number of missions
Variables	I - First steps	9
Decision structures	II – Making decisions	5
Repetition structures	III -Time to repeat	7
Arrays	IV - Grouping	5
Functions	V - My instructions	4

The first chapter (I) deals with the introduction of the programming language through the use of variables and varied exercises that introduce the actions of a robot in order to familiarize the student with the programming process. The second chapter (II) adds to the decision making, the third chapter (III) the concept of repetition, the fourth chapter (IV) the ability to group information and the last chapter (V) to create custom functions (see Table V).

TABLE V: CURRICULUM INTEGRATED MISSIONS OBJECTIVES

Chapter	Mission(s)	Objective
I	1, 2 and 3	Introduce the robot and mechanics of the game, and the concept of process and output
I	4	Introduce the concept of input
I	5 and 6	Working with coordinates (Cartesian plane)
I	7	Use of radar, object detection
I	8	To learn how to carry objects
I	9	To learn how to fire, destruction of objects
II	1 and 2	Making decisions, working with cardinal directions
II	3	Decisions with intercardinal directions
II	4	Decisions with multiple conditions
II	5	Chaining of decisions
III	1	To learn how to repeat
III	2	To work with infinite loops
III	3 and 4	To work with loops and object detection
III	5, 6 and 7	To work with loops and destruction of objects
IV	1 and 2	To learn how to work with arrays and object detection
IV	3 and 4	To work with arrays and destruction of objects
IV	5	To work with multiple-dimensional arrays
V	1	To learn how to make functions
V	2	To make functions of destruction of objects
V	3	To make functions for work with cardinal directions

The missions are available in a GitHub repository [7] (see Fig. 9) so that anyone can download and use them in the game. The secondary objective of this work was to provide a version of Colobot in Portuguese; the translation of the game into Brazilian Portuguese has been successfully completed, and may be obtained from the official game repository [7]. Hence, we provide the missions in English and Brazilian Portuguese, the language of the game will define the language of the missions.

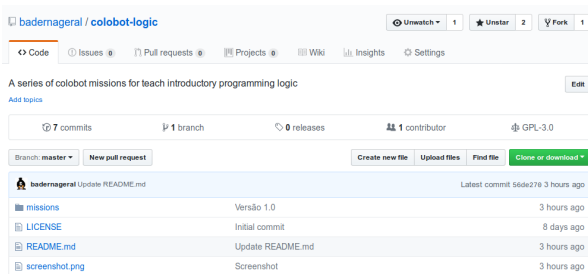


Fig. 9. Repository of missions in GitHub.

The missions were designed with an approach of incremental difficulty, that is, complexity increased little by little, in order to provide the time needed for the student to adapt. In order to demonstrate the amount of work necessary to design and develop the missions that are both in English and Portuguese, we present the missions status in the GitHub in Fig. 10. We have 4,010 code lines divided in 117 files. In addition to that, in Fig. 11, we have the Colobot screen in Portuguese of the missions.

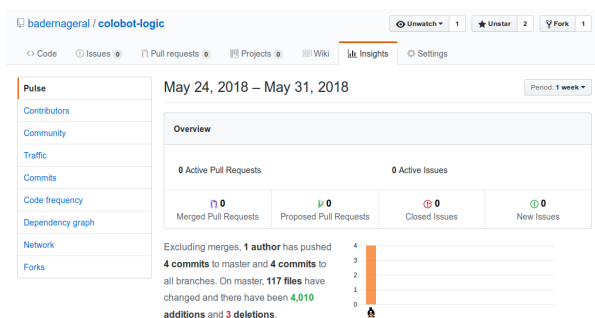


Fig. 10. Missions repository status.



Fig. 11. Missions in Portuguese in Colobot.

We encountered a problem when attempting to create more dynamic missions, the impossibility of creating random values, such as positions of objects, or conditions of victory, which could be solved only by modifying the way the missions are created. Unfortunately, the modification of the mechanics of the game was a task not performed in this work, due to its complexity and because of the scope of this work.

About the translation, the most significant contribution is the translation of the game manual (see Fig. 12), which will make a big difference when the game is used in classrooms that use Portuguese as the main language.

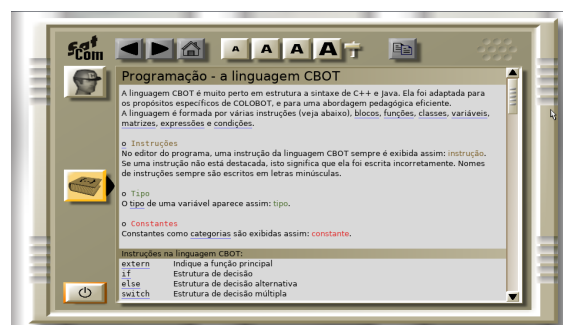


Fig. 12. Colobot's manual in Portuguese.

IV. CONCLUSIONS

We developed a series of missions in Colobot (Gold Edition) that may be used for a course of 'Introduction to Programming Logic', both in English and Portuguese, with the translation of the game to Portuguese [5,7,15]. We have accomplished that with a Curriculum integration based on the experience of Shapi'i and Ghulam in [9], the IPO Model [10] and the Interdisciplinary Research Project Management strategy [8].

We are convinced that the game Colobot shows great potential for teaching programming logic based on an informal assessment with students. The curricular integration made from the creation of the missions opens a range of possibilities, since it allows, in addition to classroom use, that other instructors use them as a basis to produce their own missions. For instance, these missions may be used as a basis for the development of interdisciplinary missions such as presented by Batista et al. in [16] (An Interdisciplinary Approach to Teaching Geometry with Serious Game 3D Colobot), which already used IRPM.

We suggest an analysis based on the study of Freitas and Oliver [17] as future work, because we did not make a formal analysis of the effectiveness of these missions for students' engagement and learning. Our main objective was to provide a guide for future mission development and to incentive others to translate the game to other languages. Hence, we expect that with the dissemination of this work, more people will join the cause and contribute to the development of the game, or see the possibilities of using this type of tool in the classroom, and that, the developed curriculum integrated missions provide an alternative for educating all high school students in programming logic with an engaging game.

REFERENCES

- [1] G. J. Contreras and K. W. M. Siu, "Computer Programming for All: A Case-Study in Product Design Education", *Procedia - Social and Behavioral Sciences*, Volume 182, pp 388-394, 2015.
- [2] M. Rahmat, S. Shahrani, R. Latih et al, "Major Problems in Basic Programming that Influence Student Performance", *Procedia - Social and Behavioral Sciences*, Volume 59, pp. 287-296, 2012.
- [3] L. A. Annetta, J. Minogue, S. Y. Holmes and M. Cheng, "Investigating the impact of video games on high school students' engagement and learning about genetics", *Computers & Education*, Volume 53, Issue 1, pp 74-85, 2009.
- [4] C. Kazimoglu, M. Kiernan, L. Bacon and L. Mackinnon, "A Serious Game for Developing Computational Thinking and Learning Introductory Computer Programming", *Procedia - Social and Behavioral Sciences*, Volume 47, pp. 1991-1999, 2012.
- [5] Colobot: Gold Edition, "3D real time game of strategy and adventure for learning programming", 2014, <http://colobot.info>.

- [6] M. Muratet, P. Torguet, J. Jessel and F. Viallet, "Towards a Serious Game to Help Students Learn Computer Programming," *International Journal of Computer Games Technology*, vol. 2009, Article ID 470590, 12 pages, 2009.
- [7] Colobot-logic, "A series of colobot missions for teach introductory programming logic", 2018, <https://github.com/bader nageral/colobot-logic>.
- [8] P. Letouze, "Interdisciplinary Research Project Management", *International Proceedings of Economics Development and Research*, vol. 14, pp 338-342, 2011.
- [9] A. Shapi'i and S. Ghulam, "Model for Educational Game Using Natural User Interface", *International Journal of Computer Games Technology*, vol. 2016, Article ID 6890351, 7 pages, 2016.
- [10] S. Sentance, E. Barendsen and C. Schulte, "Computer Science Education: Perspectives on Teaching and Learning in School", Bloomsbury Publishing, 264 pages, 2018.
- [11] F. Laamarti, M. Eid and A. E. Saddik, "An Overview of Serious Games", *International Journal of Computer Games Technology*, vol. 2014, Article ID 358152, 15 pages, 2014.
- [12] K. Heldman, "PMP project management professional exam deluxe study guide: updated for the 2015 Exam", John Wiley & Sons, 2015.
- [13] S. de Freitas, "New Pedagogical Approaches in Game Enhanced Learning: Curriculum Integration", Idea Group Inc (IGI) UnitedStatesofAmerica, 2013.
- [14] A. C. Prudêncio, D. A. Valois and J. E. Lucca, "Introdução à internacionalização e à localização de softwares," *Cadernos de Tradução*, vol. 2, n° 14, 2004, pp. 211-242.
- [15] Colobot: Gold Edition, "Source code of open-source Colobot: Gold Edition project", 2014, <https://github.com/colobot>.
- [16] R. R. Batista, E. R. Ferraz and P. Letouze, "Uma Abordagem Interdisciplinar para o Ensino de Geometria com o Serious Game 3D Colobot" (An Interdisciplinary Approach to Teaching Geometry with Serious Game 3D Colobot), *Proceedings of the XVI SBGAMES – Simpósio Brasileiro de Games (XVI Brazilian Games Symposium)*, 2017.
- [17] S. de Freitas and M. Oliver, "How can exploratory learning with games and simulations within the curriculum be most effectively evaluated?", *Computers & Education*, v. 46, n. 3, pp 249-264, 2006.



Born in São José de Piranhas, Paraíba, Brazil, in 1989. Postgraduate in Systems Engineering at High Open School of Brazil in 2016 and graduate in Analysis and Systems Development at University North of Paraná, Brazil, in 2014.

He is a professor in Federal Institute of Tocantins, Brazil and researches the development and use of games in education, as well as other minor research about software development.

Prof. Alves is a student of the Graduate Program in Computational Modeling of Systems in the Federal University of Tocantins, Brazil.



Patrick Letouze Moreira has a bachelor degree in Control and Automation Engineering (Universidade Federal de Santa Catarina,); a Master Science degree and a Doctor in Science degree in Electrical Engineering.

He is a professor at the Universidade Federal do Tocantins (UFT). He is the vice-director of the Graduate Program of Computational Modeling of Systems, since its beginning; and he was the director of the Graduate Program of Health and Science Education, since its creation until the first quarter of 2018. He is a fellow member of IEDRC and a member of IEEE.

**APÊNDICE B – GAMIFICATION NO AMBIENTE VIRTUAL DE APRENDIZAGEM
MOODLE: ELEMENTOS, RECURSOS E ESTRATÉGIAS**

GAMIFICATION NO AMBIENTE VIRTUAL DE APRENDIZAGEM MOODLE: ELEMENTOS, RECURSOS E ESTRATÉGIAS

Resumo – Ambientes virtuais de aprendizagem, também chamados de salas de aula online, são sistemas formados por várias ferramentas que proporcionam interação entre professor e aluno. O ambiente virtual de aprendizagem Moodle é um dos mais utilizados no mundo e possui uma grande comunidade que sempre está desenvolvendo novas funcionalidades. Como em toda sala de aula, há problemas de motivação, pensando nisso, atualmente muitos estão usando *gamification* para motivar seus usuários, isto é, utilizar elementos de jogos em ambientes que não são jogos. Apesar de ser uma abordagem nova na educação, já tem produzido resultados expressivos e muitos sistemas estão utilizando-a atualmente. Este artigo lista os elementos de jogos mais utilizados em *gamification* e realiza um *link* entre os recursos nativos e *plugins* da comunidade disponíveis no diretório de *plugins* do Moodle com os elementos que eles implementam, porém não basta somente adicionar os *plugins*, é preciso ter boas estratégias para não causar o efeito contrário, desta forma o artigo também levanta uma lista com boas estratégias para o uso de *gamification* no Moodle.

Palavras-chave: Gamification. Moodle. Plugins. Estratégias. Jogos.

1 Introdução

O sistema educativo sofre alterações constantes ao longo do tempo, atualmente é comum ver escolas utilizando Ambientes Virtuais de Aprendizagem (AVA), sistemas que proporcionam uma integração online entre professor e estudante. Apesar dos benefícios, continua sendo um desafio motivar os estudantes, especialmente em ambientes como esses, onde existe uma distância física.

Kapp (2012) nos diz que no quesito “motivar pessoas” existem áreas bem-sucedidas, por exemplo: jogos. Pare um pouco e olhe ao seu redor: seus amigos estão jogando no celular, seu filho está jogando no *tablet*, seu parceiro está jogando no *facebook*, seus avós estão jogando na mesa. Os jogos estão presentes no cotidiano das pessoas, sejam crianças ou adultos, até mesmo idosos, todos estão jogando, e o mais incrível nisso tudo, é que nenhum deles está sendo obrigado a jogar, todos estão jogando porque querem. Com isto em mente, profissionais estão tirando as características dos jogos que os fazem tão interessantes e implementando-as em outras áreas, principalmente na educação.

A utilização de mecânicas, aparência e pensamentos baseados em jogos para motivar pessoas, promover a aprendizagem e resolver problemas em contextos que não são jogos é conhecida como *gamification*. (KAPP, 2012).

O uso de *gamification* não é uma abordagem nova, são utilizadas em diversas áreas, bem como existem diversos exemplos de uso no ensino, com altos índices de aprovação pelos usuários, inclusive em AVAs.

É importante ressaltar que a má utilização de *gamification* pode provocar um efeito contrário, por exemplo, estudantes que se interessam somente pelos prêmios e não pelo conteúdo. (BUSARELLO, 2016)

2 Elementos dos jogos

Muitos modelos para o estudo de *gamification* dividem os elementos em três categorias: dinâmicas, mecânicas e componentes. (ALVES, 2015)

2.1 Dinâmicas

As dinâmicas são responsáveis por fornecer harmonia e padrões regulares à experiência. (ALVES, 2015)

Dinâmica	Descrição
Constrições	Restrições que impedem o jogador a ir pelo caminho mais óbvio, criando alternativas e estimulando o pensamento estratégico.
Emoções	A emoção de superar um desafio, ser motivado por <i>feedback</i> ou receber uma recompensa é essencial para fornecer uma boa experiência.
Narrativa	Não precisa ser uma história, mas essencialmente deve estar relacionada com o contexto de forma a dar propósito aos elementos presentes.
Progressão	Mecanismos que fornecem ao jogador a sensação de que está progredindo, incentivando-o a seguir em frente.
Relacionamento	Interação entre outros jogadores também é essencial.

Quadro 1 – Dinâmicas

2.2 Mecânicas

As mecânicas são as ações que levam o sistema adiante. (ALVES, 2015)

Mecânica	Descrição
Desafios	Objetivos propostos aos jogadores.
Cooperação e competição	Promovem ao jogador o desejo de estar junto de outras pessoas, seja para construir algo em conjunto ou para superá-las.
<i>Feedback</i>	Fundamental, pois permite ao jogador acompanhar seu progresso.
Aquisição de recursos	Os recursos adquiridos podem ser usados para conquistar algo maior ou utilizados em transações.
Transações	Compra, venda e troca são as mais comuns.
Recompensas	Prêmios ou benefícios conquistados.
Estados de vitória	Define quem ganhou o jogo, por exemplo, conquistar o maior número de pontos, eliminar todas as peças do adversário, entre outros.

Quadro 2 – Mecânicas

2.3 Componentes

Os componentes são a implementação do que a dinâmica e a mecânica representam. (ALVES, 2015)

Componente	Descrição
Realizações	Formas de recompensar o jogador por completar desafios.
<i>Badges</i>	Representações visuais das realizações. A palavra significa distintivo.
Avatares	Representação visual do seu personagem.
<i>BossFights</i>	Batalhas com o chefe são desafios maiores que normalmente representam a passagem de uma fase ou nível para outro.
Coleções	Coletar e colecionar objetos pelo jogo.
Combate	Travar uma batalha contra um oponente.
Desbloqueio de conteúdos	Requerer uma ação para ter acesso a um conteúdo.
Doar	Doações estimulam o altruísmo entre os jogadores.
Placar	Sistema de ranqueamento que gera competição entre os jogadores.
Níveis	Graus diferentes de dificuldade que o jogador encontra no decorrer do jogo enquanto evolui suas habilidades avançando de um nível a outro.
Pontos	Contagem de pontos acumulados no decorrer do jogo.
Investigação ou exploração	Alcance de resultados através de exploração ou investigação, incitando a curiosidade do jogador.
Bens virtuais	Coisas virtuais obtidas através do pagamento com moeda virtual ou dinheiro real, por exemplo, uma espada para seu personagem.

Quadro 3 – Componentes

3 Recursos e *plugins*

O *Modular Object-Oriented Dynamic Learning Environment* (Moodle) é um AVA livre e altamente personalizável. A maioria das funcionalidades disponíveis são *plugins* individuais ou coleções dos mesmos. É possível estender as funcionalidades através da instalação de *plugins* criados pela comunidade que podem ser obtidos no diretório de *plugins* do Moodle. (HENRICK; HOLLAND, 2015) Blocos são pequenos componentes personalizáveis adicionados ao *layout*, posicionam-se a esquerda e/ou

a direita do conteúdo e fornecem diversas funcionalidades, como um calendário, um relatório ou até mesmo uma lista de usuários *on-line*.



Figura 1 – Layout do Moodle.

Atividades são como o nome diz, interações que permitem avaliar a participação do estudante através da atribuição de notas. Segundo DENMEADE (2015), dentre os recursos nativos do Moodle, deve-se dar destaque a dois que podem ser aplicados a atividades e são extremamente importantes ao se trabalhar com *gamification*:

- Acompanhamento de conclusão – adiciona uma caixa de seleção ao lado da atividade que será marcada automaticamente quando o estudante satisfizer as condições definidas pelo professor. Caso não se use condições, o estudante pode marcar manualmente;
- Restrições de acesso – permite-se adicionar restrições que impedem o acesso a determinada atividade ou conteúdo, as restrições disponíveis incluem data, nota, conclusão de atividade e perfil de usuário. *Plugins* de restrições de disponibilidade adicionam novos tipos de restrições.

Através dos recursos de acompanhamento de conclusão e restrições de acesso obtém-se o componente citado anteriormente: **desbloqueio de conteúdos**.



Figura 2 – Acompanhamento de conclusão e restrições de acesso.

Alguns professores costumam dar “estrelinhas de papel” a seus alunos, essa estratégia é mais utilizada no ensino fundamental, esse “prêmio” conquistado através de boa conduta ou boas notas, incentiva o estudante. O Moodle possui suporte nativo a emblemas, ou seja, não é necessário instalar. Este recurso possibilita dar emblemas a estudantes de forma manual ou automática, desde que satisfaçam os requisitos estabelecidos. O uso de emblemas no Moodle caracteriza o componente *badges* (distintivos), que representa um tipo de **realização**, outro componente.



Figura 3 – Emblemas.

Level Up! e *Level Availability* são dois *plugins* criados pela comunidade que implementam os componentes **pontos**, **níveis**, **placar** e **desbloqueio de conteúdos**. Eles podem ser obtidos através do diretório de *plugins* do Moodle. O primeiro possibilita ao estudante acumular pontos e subir de nível quando completa atividades e interage com o AVA, bem como disponibiliza um *ranking* que lista os estudantes de acordo com a quantidade de pontos. O segundo permite criar restrições por nível nas atividades e conteúdos. A forma como o estudante ganha os pontos e a quantidade de pontos por nível pode ser personalizada pelo professor.

Missão secreta

Restrição

Disponível se: Você está no nível 2 ou maior.

Posição	Nome completo	Nível	Pontos de experiência	Progresso
1	José Robson Mariano Alves	2	150	30/156
2	Demétrio de Oliveira Rocha	1	50	50/120

Ranking

MEU NÍVEL

1

30/120

Participe do curso para ganhar pontos de experiência e passar de nível!

Informação - Visualizar o ranking

Bloco

Figura 4 – Level Up! e Level Availability.

Os componentes **exploração**, **coleções** e **desbloqueio de conteúdos** podem ser implementados através dos *plugins* *Stash* e *Stash Availability*, também disponíveis no diretório de *plugins* do Moodle. O primeiro permite ao professor criar “objetos” que podem ser colocados em qualquer lugar do curso, quando o estudante passar por ali ele poderá coletar o item e armazenar em sua coleção. O segundo permite restringir acesso ao conteúdo de acordo com objetos da coleção do usuário.

HTML básico? conheço demais!!!

Disponível se: Você tem 'Espada binária'

Restrição

Espada binária

Coletar!

Coleta

INVENTÁRIO

Bloco

Figura 5 – Stash e Stash Availability.

O *plugin* *ActiveQuiz* é um *plugin* do tipo atividade e pode ser utilizado para implementar o componente **combate** (competição) entre os estudantes. Ele permite criar questionários e sessões com os mesmos, os estudantes se juntam à sessão e começam a responder as perguntas antes do tempo limite

acabar. O professor tem acesso a um relatório de classificação após o fim da sessão.

Os *plugins* *NED Tabs Course Format* e *NED My Progress* aumentam o *feedback* ao estudante, o primeiro organiza o curso em formato de abas, sendo que as abas vermelhas simbolizam a existência de atividades não realizadas. O segundo cria um bloco que fornece um relatório ao estudante sobre as atividades realizadas e não realizadas, o título desse bloco é personalizável.

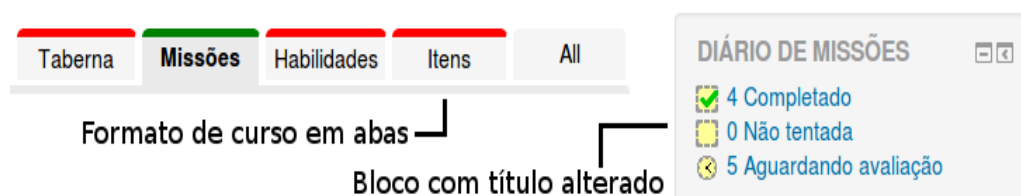


Figura 6 – *NED Tabs Course Format* e *NED My Progress*.

O sistema de mensagens do Moodle, a criação de grupos para realização de tarefas e atividades do tipo fórum, *wiki* e *chat* possibilitam interação e colaboração entre os estudantes.

O *pluginGame* adiciona vários tipos de atividades em forma de jogos ao Moodle, como cruzadinhas, jogo da forca, entre outros.

4 Estratégias

Um AVA que utiliza técnicas apropriadas de *gamification* consegue atrair mais a atenção dos estudantes, e conseqüentemente, aumenta a taxa de retenção de conteúdo, porém utilizar *gamification* no processo de aprendizagem é uma tarefa árdua. (BUSARELLO, 2016)

Abaixo você confere algumas estratégias e dicas para uso de *gamification* no Moodle. Deve-se frisar que muitas delas se aplicam a qualquer AVA e até mesmo ao mundo real.

4.1 Estabeleça um mundo novo

Neto (2015) diz que a temática ou a fantasia é um aspecto motivacional importante. Com isso em mente, personalizar o ambiente para que o mesmo se torne diferente do mundo real ajuda na motivação do estudante.

Criar uma nomenclatura única contribui para a fantasia, por exemplo, alterar o nome dos blocos, chamar o estudante de guerreiro, de missões as atividades, de habilidades o conteúdo a ser aprendido, de itens as ferramentas a serem utilizadas etc. A figura 6 mostra um ambiente com nomenclatura personalizada. Utilizar um sistema de notas por conceitos também contribui para a fantasia, por exemplo, no lugar de notas de 0 a 10, pode-se utilizar conceitos como Bronze, Prata, Ouro, Platina, Diamante etc. Os Rótulos são componentes adicionados ao curso que podem conter textos e imagens. Utilizá-los com sabedoria pode mudar completamente a aparência do curso e torná-lo mais atraente e convidativo.

4.2 Narrativa

A narrativa é muito importante em um sistema que utilize *gamification*, pois sem uma boa narrativa os componentes de jogos podem aparentar terem sido colocados sem um propósito claro. (NETO, 2015)

Ao criar atividades e nomeá-las como missões, cria-se a expectativa de que a missão ofereça uma recompensa, o conteúdo da missão pode descrever além dos propósitos da atividade, uma pequena história e as recompensas que o estudante obterá ao completar a missão, como pontos ou itens, dessa forma, a narrativa dá um propósito ao sistema de pontos e itens, que podem ser utilizados para desbloquear outras missões.

Ao começar uma narrativa em uma atividade e continuá-la em mais uma ou duas atividades, cria-se uma série ou campanha. Para que a narrativa seja seguida a risca, pode-se criar as atividades de forma a possuírem restrições de conclusão em relação as outras, por exemplo, o estudante não poderá realizar a segunda atividade sem completar a primeira e a narrativa da segunda atividade é uma continuação da narrativa da primeira.

Ao nomear atividades, recomenda-se utilizar títulos que representem o conteúdo das mesmas e se possível, com uma pitada de humor para atrair a atenção dos estudantes. A figura 2 exibe algumas atividades com títulos que representam seu conteúdo e possuem humor.

4.3 Competição saudável

Sistemas de ranqueamento são muito utilizados em *gamification*, eles geram competição entre os usuários que disputam o 1º lugar, porém nem todas as pessoas são competitivas, algumas podem até não gostar de competições.

O *ranking* do *pluginLevel Up!* leva em consideração os pontos obtidos pelos estudantes. A forma de obtenção de pontos varia desde a abertura do curso a completar uma tarefa, ou seja, é um ranqueamento por acessos, pois desde que o estudante abra o curso, ele ganhará pontos. Isso pode causar um comportamento conhecido como *cheat* (fraude), pois se o estudante abrir o curso diversas vezes ele ganhará pontos e criará uma distância entre ele e os demais, sem que o mesmo tenha realmente completado tarefas que mereçam o recebimento destes pontos.

Como a obtenção de pontos pode ser livremente configurada, caso o estudante só ganhe pontos quando o acompanhamento de conclusão for ativado, ou seja, quando completar uma atividade, cria-se um *ranking* “justo”, pois não avalia quem acessa mais o ambiente, mas sim quem completa mais atividades. Quando os estudantes completam todas as atividades, eles compartilham o 1º lugar. O *ranking* também pode ser utilizado pelo professor como *feedback* para acompanhar os alunos que estão com

dificuldades ou que não estão participando ativamente do curso.

4.4 Dificuldades e a teoria do fluxo

Segundo o psicólogo Csikszentmihalyi (1990, apud ALVES, 2015, p.171), quando totalmente focados em um jogo ou brincadeira estamos em um estado chamado “flow” (fluxo).

A dificuldade da tarefa também determina se ela é divertida. Para manter o estado de fluxo, deve-se regular a dificuldade da atividade de forma que não fique fácil demais, tornando-se tediosa, nem difícil demais, gerando ansiedade. É por esse motivo que os jogos vão aumentando a dificuldade gradativamente. (ALVES, 2015)

Ao criar atividades deve-se aumentar a dificuldade das mesmas ao longo do tempo de forma a manter o estado de fluxo. Restringir as atividades por níveis de experiência deixa claro ao estudante que atividades que requerem o nível 2 são mais difíceis do que as que requerem o nível 1.

Desafios como *BossFights*, atividades com um nível de dificuldade maior em relação às outras, entretanto com recompensas maiores, desafiam o aluno e proporcionam uma sensação maior de superação ao completá-las. Como recompensas, por exemplo, pode-se oferecer um emblema que dá algum bônus à nota do estudante ou itens que desbloqueiam um “baú do tesouro”.

Para estimular o esforço e a pontualidade pode-se utilizar emblemas que somente são obtidos caso o estudante complete um desafio até uma data limite.

4.5 Metas

É importante estabelecer metas que os jogadores devam alcançar e fazer com que elas sejam adotadas por eles. (BURKE, 2015)

Não é interessante deixar claro ao estudante que ele é obrigado a fazer as atividades, pode-se criá-las de forma que o mesmo não tenha muita pressão e instigá-lo a realizar as atividades com uma meta que possa ser alcançada dessa forma, por exemplo, a meta definida pelo professor é “chegar ao nível 10”, o estudante adota a meta e realiza todas as atividades para chegar ao nível 10.

A meta anterior pode ser alcançada se não existirem datas para entrega em nenhuma atividade (sem pressão), mas existirem restrições que impeçam a realização das atividades mais recentes, caso ele não complete as antigas (obrigação). Pode-se fazer isso utilizando as restrições por **nível** do *pluginLevel Availability*. O estudante pensará que, por exemplo, porque ele não é nível 2, ele não pode realizar as atividades mais avançadas, logo ele tem de “jogar” mais para subir de nível e ter acesso às atividades. Veja a figura 4 que demonstra uma atividade com restrição de nível.

4.6 Curiosidade e surpresa

Segundo BUSARELLO (2016) as motivações intrínsecas originam-se do próprio sujeito, quando o indivíduo interage com as coisas por vontade própria, pois elas despertam interesse, desafio, envolvimento e prazer, incitando sua curiosidade e busca por novos conhecimentos. A exploração, os desafios e a narrativa são algumas das formas para incentivar a motivação intrínseca.

Para incitar a curiosidade pode-se criar “baús do tesouro”, conteúdos que possuem múltiplas restrições, porém contém grandes recompensas. No Moodle o recurso **página** é uma boa escolha para criação de baús do tesouro, pois permite colocar conteúdo estático sem necessidade de avaliação, porque não é um tipo de atividade. Para limitar o acesso ao “baú” pode-se criar uma restrição de um objeto do *Stash*, como uma “chave”, obtida através de uma atividade. Caso queira aumentar o desafio pode-se utilizar múltiplas restrições, como múltiplas “chaves” para abrir um “baú”. Ao abrir esse “baú” o estudante encontraria itens ou o final de uma narrativa, por exemplo.

A surpresa também é um elemento fundamental em ambientes que utilizam *gamification*. O recurso **Restrições de acesso** permite obter esse comportamento no Moodle, com ele pode-se criar atividades ocultas que aparecem quando o estudante satisfaz condições pré-estabelecidas.

4.7 Criando hábitos

Segundo BUSARELLO (2016) as motivações extrínsecas originam-se do mundo que envolve o usuário, ou seja, são externas a ele e têm como ponto de partida o desejo de obter uma recompensa como reconhecimento social ou prêmios. Os pontos, o *ranking*, os emblemas e as coleções são algumas formas de incentivar a motivação extrínseca.

Para motivar os estudantes a visitar os fóruns regularmente e conseqüentemente adquirir esse hábito, pode-se colocar objetos do *Stash* nos fóruns e regularmente postar atividades que necessitem desses objetos para desbloqueá-las.

Conforme o exemplo acima, a motivação extrínseca foi utilizada para instigar hábitos nos estudantes, porém é necessário frisar que um efeito negativo pode ser obtido, por exemplo, os estudantes visitam os fóruns somente para coletar os itens e nunca olham os tópicos. Aliar alguma ação do estudante à obtenção do prêmio pode ajudar a evitar esses problemas. O Moodle oferece fóruns do tipo **perguntas e respostas**, nos quais somente após o estudante responder, ele poderá ver as respostas dos outros, dessa forma é possível esconder o objeto e liberar o acesso somente quando o estudante contribuir para a discussão.

Criar atividades do tipo **lição** permite, além do conteúdo para estudo, a inserção de perguntas. Pode-se personalizar o *feedback* positivo destas perguntas e colocar um item do *Stash* que desbloqueie uma atividade, dessa forma, para que o estudante tenha acesso à atividade ele deverá visualizar a lição,

ler o conteúdo e responder à pergunta corretamente para coletar o item.

5 Considerações finais

Os desafios e dinâmicas que a *gamification* oferece ao estudante incentivam a aprendizagem e proporcionam pensamento lógico, o que a torna uma boa adição a “caixa de ferramentas” do professor, como também demonstra a mudança que o papel do professor sofreu nas últimas décadas com o advento das novas tecnologias, conforme o pensamento de Lévy na década de 90:

“A principal função do professor não pode mais ser uma difusão dos conhecimentos, que agora é feita de forma mais eficaz por outros meios. Sua competência deve deslocar-se no sentido de incentivar a aprendizagem e o pensamento”. (LÉVY, 1999, p.171)

Sobre a relação entre *gamification* e educação, Fardo (2013) diz que a mesma é um fenômeno que merece atenção, pois a popularidade dos jogos só aumenta e conseqüentemente torna seu uso mais comum a cada dia, porém ressalta que ela não é a solução para todos os problemas da educação.

Apesar da *gamification* contribuir para o engajamento dos estudantes, é preciso ter cuidado e não exagerar em sua utilização, por exemplo, muitas restrições, narrativas extensas, desafios impossíveis, entre outros.

Os componentes **avatares**, **doações** e **bens virtuais** não possuem *plugins* que possibilitem sua implementação no Moodle. Observando o fato de que os últimos *plugins* de *gamification* foram adicionados recentemente ao diretório, pode-se concluir que a comunidade está ativa e no futuro estes componentes podem ser disponibilizados.

O Moodle oferece diversas possibilidades para o uso de *gamification* que dependem principalmente do esforço e da imaginação do professor, pode ser um pouco desgastante no início, mas os resultados são compensadores enquanto que o esforço e a diversão caminham juntos com o processo.

Referências

- ALVES, Flora. Gamification: como criar experiências de aprendizagem engajadoras. Um guia completo do conceito a prática. 2º ed. São Paulo: DVS, 2015. 209 p.
- BURKE, Brian. Gamificar: como a gamificação motiva as pessoas a fazerem coisas extraordinárias. São Paulo: DVS, 2015. 192 p.
- BUSARELLO, Raul Inácio. Gamification: princípios e estratégias. São Paulo: Pimenta Cultural, 2016. Disponível em: <<http://www.pimentacultural.com/#!gamification/c1et2>>. Acesso em: 05 jan. 2017.
- DENMEADE, Natalie. Gamification with Moodle. Reino Unido: Packt, 2015. 134 p.

FARDO, Marcelo Luis. A Gamificação aplicada em Ambientes de Aprendizagem. Revista Novas Tecnologias na Educação, Porto Alegre, v. 11, n. 1, 2013. Disponível em: <<http://seer.ufrgs.br/index.php/renote/article/view/41629/26409>>. Acesso em: 09 jan. 2017.

HENRICK, Gavin; HOLLAND, Karen. Moodle Administration Essentials. Reino Unido: Packt, 2015. 156 p.

KAPP, Karl M.. The gamification of learning and instructions: game based methods and strategies for training and education. Estados unidos: Pfeiffer, 2012. 336 p.

LÉVY, Pierre. Cibercultura. São Paulo: 34, 1999. 264 p.

NETO, Henrique Ruiz Poyatos. Gamificação: engajando pessoas de maneira lúdica. São Paulo: Fiap, 2015. 98 p.

APÊNDICE C – GAMIFICAÇÃO: A SOLUÇÃO PARA TODOS OS PROBLEMAS?

Quando você tira as diferenças de gênero e as complexidades tecnológicas, todos os jogos compartilham quatro características definidoras: um objetivo, regras, um sistema de feedback e participação voluntária.

Jane McGonigal

Gamificação: a solução para todos os problemas?

Os jogos estão presentes na sociedade humana desde a antiguidade, são utilizados principalmente para fins de entretenimento, pois possuem grande capacidade de engajamento, ou seja, evocam dedicação nos jogadores, resultando em uma participação com afinco e vontade. A popularidade dos jogos está relacionada a diversão que eles proporcionam, mas qual a fórmula do seu sucesso? Pode-se usar essa fórmula em outras áreas? A resposta para estas perguntas é a Gamificação. Com isso em mente, este capítulo se propõe a conceituar a Gamificação, levantar os componentes que a compõe e as teorias motivacionais que a suportam, bem como apresentar sua utilização na educação.

O termo Gamificação vem do inglês: *gamification*, e foi usado pela primeira vez em 2002, porém só passou a ser difundido no segundo semestre de 2010, quando se popularizou. O termo gerou muita controvérsia por causa de múltiplas interpretações e o trabalho de Deterding et al. (2011) foi o primeiro a apresentar uma definição formal para a gamificação: uso de elementos de design de jogos em contextos não relacionados a jogos. No ano seguinte Kapp (2012) define gamificação como a utilização de mecânicas, aparência e pensamentos baseados em jogos para motivar pessoas, promover aprendizagem e resolver problemas em contextos que não são jogos.

Perceba que na definição de Kapp a motivação entra em cena, pois um dos propósitos da gamificação é utilizar a fórmula dos jogos (mecânicas, aparência e pensamentos) para engajar os envolvidos. É comum encontrar professores no ensino fundamental que dão estrelinhas para alunos exemplares, esse é um exemplo básico de gamificação na sala de aula - contexto não relacionado a jogos – nesse caso as estrelinhas são medalhas que representam uma mecânica encontrada em jogos para promover engajamento.

É importante frisar que jogos sérios, jogos educacionais ou ambientes que simplesmente tenham a aparência de jogos, porém não tem como foco motivar seus usuários a cumprir um objetivo, não são exemplos de gamificação. A gamificação vai além de simplesmente inserir um sistema de pontos ou um avatar, ela deve motivar seus usuários. Seu objetivo final não é a diversão e a mesma não é um jogo, ou seja, seu propósito não é ser jogável.

Tudo se resume a motivação

Segundo Skinner (2003)² em sua teoria do condicionamento operante, é possível modelar o comportamento de um indivíduo a partir de estímulos. Ele fez experimentos com ratos e desenvolveu uma caixa conhecida como “Caixa de Skinner”, onde um rato é colocado no interior da mesma junto a uma alavanca, que quando era puxada pelo rato fornecia alimento ao mesmo, dessa forma o rato aprendia a utilizar a alavanca quando necessitasse de alimento, caso o alimento parasse de ser fornecido quando o rato puxasse a alavanca, o mesmo cessaria esse comportamento, mas se o fornecimento fosse mantido com intervalos regulares, por exemplo, puxar a alavanca 10 vezes para obter alimento, o comportamento seria preservado.

Este conceito de condicionamento operante é muito utilizado em jogos e gamificação, por exemplo, alguns aplicativos de compras dão parte de um cupom (estímulo) para os clientes que *logam* todo dia (comportamento), e um cupom completo para cada 7 dias (reforço com intervalo fixo), com isso induzem o cliente a entrar no aplicativo diariamente.

A recompensa de Skinner gera uma motivação denominada extrínseca, pois é resultante de um estímulo externo, caso o estímulo seja removido, a motivação deixa de existir. Motivações resultantes de estímulos internos são denominadas intrínsecas. Tome como exemplo a doação de sangue, se a pessoa doar sangue por altruísmo, a motivação é intrínseca, caso a doação seja motivada por uma recompensa, será extrínseca.

Pink (2010) nos diz que motivações extrínsecas duram pouco tempo e podem causar o efeito contrário, pois o usuário foca-se apenas na recompensa, diferente das motivações intrínsecas, que promovem um maior envolvimento, pois são motivadas por satisfação própria. A gamificação promove principalmente motivações intrínsecas, o que a diferencia de outras abordagens.

Ainda falando sobre motivação, cada pessoa possui particularidades que influenciam sua motivação para jogar, Bartle (1996) dividiu os jogadores de MUDs (Multi-user Dungeon) – um tipo de jogo para múltiplos jogadores - em 4 tipos de acordo com sua motivação: Assassinos, Exploradores, Socializadores e Conquistadores.

Os assassinos são os mais competitivos, estão interessados em demonstrar sua superioridade para outros humanos, são orgulhosos de suas habilidades e acham mais divertido usá-las em uma pessoa real ao invés de uma entidade computadorizada sem emoção.

Os exploradores são os mais curiosos, querem descobrir o máximo que puderem, gostam de interagir com o jogo, deixam que o jogo os surpreendam, exploram o mundo, descobrem novas áreas, são orgulhosos de seu conhecimento.

Os socializadores são os mais sociáveis, estão interessados em interagir com outras pessoas, gostam de conversar e conhecer pessoas, o jogo é apenas um fundo, são os personagens que o tornam interessante, são orgulhosos de suas amizades, contatos e influências.

Os conquistadores são os mais ambiciosos, estão interessados em fazer coisas no jogo, em conquistar os objetivos, subir no ranking, conseguir pontos e medalhas, são orgulhosos de seu status na hierarquia do jogo.



Figura 1 – Tipos de jogadores segundo Bartle

É importante entender o que motiva os jogadores a jogar, pois você pode (e deve) explorar essas fontes de motivação. O estudo de Bartle não é o único, existem estudos posteriores mais abrangentes, em especial o de MARCZEWSKI (2015) que além de diversificar o assunto, fornece um *framework* para identificação de tipos de jogadores.

Elementos de jogo

Para gamificar algo é necessário utilizar elementos de jogos, nesse caso, é preciso conhecê-los, Werbach e Hunter (2015) constroem uma hierarquia e dividem os elementos em 3 tipos: Dinâmicas, Mecânicas e Componentes.



Figura 2 – Hierarquia de elementos de jogos por Werbach e Hunter

A proposta de Werbach e Hunter (2015) não aborda todos os componentes existentes, porém os mais importantes foram incluídos. As dinâmicas são os elementos do topo da hierarquia, são complexas para projetar e úteis para apontar a direção correta para onde você deve ir. Existem 5 dinâmicas:

Restrições – restringem as ações do jogador, forçando que o mesmo escolha somente as opções disponíveis. Os jogos são compostos por regras e limitações, restringir as ações do usuário aumentam o engajamento, por exemplo, uma limitação de tempo gera tensão que pode aumentar o foco do usuário na tarefa.

Emoções – como chave da motivação intrínseca, as emoções são fundamentais para motivar os jogadores. Curiosidade, competitividade, felicidade, orgulho, até emoções negativas como frustração podem ser utilizadas para gerar motivação.

Narrativa – é responsável por dar sentido ao jogo, pode ser explícita – uma história está sendo contada – ou implícita, estabelecendo lógica entre os componentes do jogo. Uma narrativa mal construída pode passar a sensação de que os componentes e os personagens do jogo não se encaixam, enquanto uma boa narrativa passa a sensação de que cada ação está conectada a um evento maior.

Progressão – jogos não podem ser estáticos, é necessário que haja alguma dinâmica de progressão e nela o jogador perceba que ele está avançando. Esse *feedback* engaja o jogador a seguir em frente.

Relacionamentos – jogos são sociais, apesar de existirem jogos para um só jogador, a grande maioria implementa interações entre jogadores, desde competições a jogos cooperativos. Com o advento dos jogos online essa dinâmica é cada vez mais utilizada, basta olhar para os jogos de maior sucesso.

As mecânicas são responsáveis por levar o jogo adiante e criar o engajamento, normalmente são meios para implementar uma ou mais dinâmicas. As mecânicas são:

Desafios – são tarefas que requerem mais esforço para serem resolvidas, dão uma sensação de superação ao serem completadas, quanto maior a dificuldade, maior essa sensação, respeitando-se o limite do impossível.

Sorte – significa que deve haver um pouco de aleatoriedade, e pode ser utilizada para diminuir a previsibilidade ou como fator de equilíbrio, podendo ser frustrante se mal gerenciada.

Competição – um jogador ou grupo ganha enquanto o outro perde, apesar de haver jogos não competitivos, a grande maioria implementa algum tipo de mecânica de competição.

Cooperação – pode acontecer em conjunto com a competição, quando dois ou mais jogadores se juntam para cumprir um objetivo comum o qual não conseguiriam individualmente.

Feedback – é a resposta dada ao jogador informando-o sobre como ele está indo, pode ser uma mensagem ou um sistema completo que o informa em tempo real.

Aquisição de recursos – é o ato dos jogadores coletarem itens que possam ser úteis no jogo, utilizando-os para troca entre outros jogadores, ou simplesmente porque é divertido.

Recompensas – é um benefício dado ao jogador por suas ações ou conquistas. Podem ter valor no jogo como itens ou pontos.

Transações – é a troca de itens que pode acontecer entre jogadores, lojas no jogo ou com o próprio sistema. Alguns jogos possuem até casas de leilão, criando assim uma economia.

Turnos – muito utilizados em jogos de cartas e de tabuleiros, exigem que cada jogador faça todas as suas jogadas antes de passar a vez para o próximo jogador.

Estados de vitória – são os objetivos que definem a vitória do jogador, por exemplo, destruir todos os inimigos ou acumular 100 pontos.

Por último temos os componentes que são instâncias das mecânicas que por sua vez são instâncias das dinâmicas, esses são realmente vistos pelos jogadores, diferente das mecânicas e dinâmicas. Os componentes mais utilizados são pontos, medalhas e tabelas de classificação.

Conquistas – são objetivos definidos, podem oferecer recompensas e normalmente são utilizados para premiar feitos memoráveis, por exemplo, matar 500 inimigos sem ser morto ou explorar todo o mapa do mundo.

Avatares – são as representações visuais dos jogadores, variam desde uma foto a um modelo 3D. Devem representar o jogador de maneira única, garantindo-o uma identidade.

Medalhas – são representações visuais das conquistas, normalmente podem ser exibidas a outros jogadores de maneira a demonstrar os méritos do jogador, assim como as medalhas dadas em competições esportivas no mundo real.

Lutas contra chefes – nos jogos, geralmente um “chefe” é um inimigo muito forte em comparação aos demais e normalmente é encontrado ao fim de um nível, onde a sua derrota permite que o jogador avance até o próximo nível, nesse caso, representa um desafio maior que os demais, podendo oferecer recompensas a altura para recompensar o esforço.

Coleções – são conjuntos de itens, equipamentos ou outros recursos do jogo, algumas conquistas podem exigir, por exemplo, que o jogador reúna todos os itens de uma coleção.

Combate – uma forma de competição, pode acontecer entre jogadores ou entre o jogador e um monstro do jogo, por exemplo.

Desbloqueio de conteúdo – quando o jogo libera novos conteúdos, caso o jogador atinja objetivos pré-definidos. Estes conteúdos podem ser novas áreas do jogo, novas missões ou até novos modos de jogar.

Presentear – o jogo permite que o jogador dê itens ou recursos para outro jogador de forma voluntária e altruísta.

Tabelas de classificação – também conhecidos como rankings, são tabelas que mostram a classificação do jogador em relação a outros jogadores seguindo critérios específicos.

Níveis – passos na progressão do jogador, o ajudam a perceber como ele está em relação aos outros e facilita a implementação de restrições e recompensas.

Pontos – representações numéricas da progressão do jogo, os pontos normalmente são utilizados como recompensas e sua acumulação possibilita subir de nível.

Missões – desafios que normalmente possuem uma história e dão recompensas quando completadas, missões podem ser encadeadas de forma a unir seus enredos compondo uma narrativa maior.

Gráfico social – exhibe as conexões sociais do jogador. Muito utilizados em redes sociais, os jogadores podem compartilhar conquistas com os amigos, verificar o status dos competidores e até solicitar recursos.

Times – grupos de jogadores que trabalham juntos em prol de um objetivo comum.

Bens virtuais – recursos do jogo que são obtidos com dinheiro virtual, ou até mesmo com dinheiro real, podem ser desde itens simples como uma arma até uma propriedade, como uma casa.

Um caso de sucesso

Se você conhece os elementos de jogos e tem em mente que gamificação não é somente pegar qualquer elemento e sair utilizando, mas sim organizá-los de maneira lógica e estruturada, com uma boa narrativa de forma a motivar intrinsecamente os usuários levando em conta as diferenças que os motivam, então você conseguiu compreender o que é gamificação. Deve-se reforçar que o objetivo deste capítulo é conceituar a gamificação, ainda há muito para se falar sobre a criação de sistemas gamificados.

Para solidificar sua compreensão, Burke (2015) em seu livro “Gamificar: como a gamificação motiva as pessoas a fazerem coisas extraordinárias” cita vários casos de sucesso que demonstram bem a aplicabilidade dos conceitos vistos até agora, pegarei como exemplo o caso do Hospital for Sick Children (Hospital para Crianças Doentes) em Toronto no Canadá que trata de crianças que lutam contra o câncer.

O hospital está sempre tentando achar as melhores terapias enquanto minimiza a dor dos pacientes, para isto ele precisa receber relatórios diários das crianças sobre o nível de dor que estão sentindo, porém o tratamento é doloroso e as crianças acabam na maioria das vezes não preenchendo corretamente os relatórios, gerando informações incongruentes.

Os pesquisadores decidiram tentar outra abordagem, visualizaram o problema do **ponto de vista das crianças** e junto a uma agência de comunicação chamada Cundari desenvolveram um aplicativo para celular chamado Esquadrão da Dor, cujo objetivo era coletar diariamente os níveis de dor das crianças. Como narrativa o aplicativo lista todos os pacientes como membros de um esquadrão especial da polícia cuja missão é capturar e destruir a dor.

O aplicativo notifica as crianças para que elas reportem os níveis de dor duas vezes ao dia. Antes do aplicativo as crianças recebiam alguma recompensa física, porém motivações extrínsecas não ajudariam neste caso, a solução encontrada foi criar um sistema de patentes (níveis), de forma que caso a criança preenchesse o relatório três dias seguidos, passaria de cabo para sargento e assim por diante. Desta forma lembrando um jogo, as crianças conseguiam ver seu progresso, ver as condecorações que já ganharam e quando teriam que preencher seu próximo relatório. E equipe também pegou atores de seriados policiais da TV e criaram vídeos de encorajamento que são exibidos juntos às missões.



Figura 3 – Aplicativo Esquadrão da Dor

Fonte: www.cundari.com

Este é um exemplo de sucesso da aplicação de gamificação para motivar usuários, as crianças sentem que são parte de algo maior e isso as motiva, a taxa de participação no preenchimento dos relatórios aumentou, bem como a obtenção de respostas mais precisas.

Gamificação na educação

A gamificação já vem sendo utilizada na educação a algum tempo, com vários casos de sucesso quando bem empregada, por exemplo, Freitas et al. (2016) gamificaram uma disciplina de um curso de graduação de forma que os alunos tivessem que produzir conteúdo em casa e levar para a sala de aula,

com isso obtiveram resultados satisfatórios como: maior empenho dos alunos, um aumento do tempo gasto em casa preparando-se para a aula, maior uso da bibliográfica da disciplina e um aumento na taxa de aprendizagem. A motivação obtida foi tamanha que em alguns casos foi necessário “desmotivar” o jogador pois as outras disciplinas estavam sendo ignoradas. Almeida et al. (2016) também obtém resultados satisfatórios com sua avaliação do processo de gamificação de uma disciplina, reforçando ainda mais a ideia de que a gamificação pode ajudar a melhorar o engajamento dos alunos, conseqüentemente aumentando a taxa de aprendizagem.

O uso da gamificação não implica no uso de ferramentas como um computador, um *tablet* ou um *smartphone*, no entanto a maioria das implementações utiliza alguma forma de tecnologia devido a facilidade de quantificação de dados e feedback imediato que a mesma possibilita. Na educação a distância é comum o uso de AVAs (Ambientes Virtual de Aprendizagem) para gerenciamento das disciplinas, conteúdos e atividades, porém estes ambientes também podem ser utilizados no ensino presencial, não só pelos benefícios que os mesmos proporcionam, mas sim pelo seu potencial para gamificação. Albino et al. (2018) fala sobre a gamificação de AVAs e defende o seu uso para gamificação, independente se o curso é a distância ou não e cita que pode ser um bom “ponto de partida” para professores que desejam gamificar suas disciplinas.

É importante frisar que a gamificação não veio para resolver todos os problemas da educação e que apesar de resultados positivos terem sido obtidos com seu uso na educação, ela também pode ser prejudicial. Toda, Valle e Isotani (2017) em seu estudo sobre o lado negro da gamificação identificam 4 resultados negativos que a gamificação pode causar: perda de desempenho, comportamento indesejado, indiferença e efeitos de declínio.

A perda de desempenho nos estudos que Toda, Valle e Isotani (2017) analisaram é decorrente do não entendimento das regras, de jogadores que não gostaram de serem penalizados, de jogadores mais ativos ganharem menos pontos do que outros menos ativos, de jogadores focarem mais nas mecânicas que no conteúdo e da dificuldade das atividades gamificadas. O comportamento indesejado ocorreu porque a gamificação causou um efeito diferente do esperado, devido a mau planejamento ou a falta do mesmo, causando por exemplo, competição excessiva, frustração por não conseguir completar os objetivos e sistemas pouco atraentes e/ou com erros. A indiferença acontece quando a gamificação não consegue influenciar os jogadores, sua implementação não engajou os alunos, nem os afastou, chegando ao ponto de que os alunos prefeririam o sistema tradicional. Por fim, os efeitos de declínio referem-se a perda gradual da motivação por causa da implantação da gamificação, que levam a perda de desempenho e de motivação com o tempo.

Toda, Valle e Isotani (2017) concluem que “esses problemas ocorreram devido à falta de métodos apropriados e/ou estruturas para planejar e implantar gamificação em um contexto de aprendizado”. Ele

também relata a falta de teorias instrucionais para suportar a implantação da gamificação, essas teorias são utilizadas para produção de estratégias de gamificação otimizadas para cada situação.

Com isto observamos que uma estratégia de gamificação mal planejada pode ser desastrosa e que sua utilização deve ser tratada com cuidado, afinal cada pessoa tem um perfil motivacional que deve ser levado em conta, bem como cada elemento de jogo tem sua situação ideal para aplicação. Espero que o leitor entenda que é necessário estudar a gamificação a fundo e construir um bom plano antes de sair aplicando-a em suas salas de aula.

Considerações finais

Nos últimos anos a gamificação passou de um tópico de pesquisa para um próspero campo interdisciplinar, enquanto as primeiras pesquisas careciam de base teórica, os trabalhos atuais estão mais maduros, focando-se em compreender mais profundamente a gamificação, porém ainda há escassez de estudos mais rigorosos avaliando os diferentes métodos propostos. (KOIVISTO; HAMARI, 2017)

A gamificação cresce a cada dia e conhecer seus aspectos positivos, bem como os negativos, é importante para que a mesma seja tratada com cuidado. O trabalho de Pink (2010) é muito importante para a compreensão do poder da motivação intrínseca, enquanto o trabalho de Kapp (2012) e Marczewski (2015) podem lhe ajudar a planejar sua solução gamificada, não esquecendo o trabalho de Burke (2015) que cita diversos casos de uso que podem clarificar o uso da gamificação.

Um bom programa de gamificação utiliza os elementos de jogos como parte da estratégia para alcançar seus objetivos e não os próprios elementos como a estratégia em si, desta forma a aplicação da gamificação, principalmente na educação, não se deve esquecer de levar os fatores do ambiente, os objetivos a serem alcançados e os jogadores em consideração.

Referências

DETERDING, Sebastian et al. Gamification: Toward a Definition. In: Proceedings of the 2011 ACM Conference on Human Factors in Computing Systems (CHI 2011), Vancouver, BC, Canadá, p. 12-15, Mai. 2011.

KAPP, Karl M. The gamification of learning and instructions: game based methods and strategies for training and education. Estados unidos: Pfeiffer, 2012. 336 p.

SKINNER, Burrhus Frederic. Ciência e comportamento humano. 11º ed. São Paulo: Martins Fontes, (1953), 2003. 489p.

PINK, Daniel. Motivação 3.0: os novos fatores motivacionais para a realização pessoal e profissional. Brasil: Campus, 2010. 240 p.

BARTLE, Richard. Hearts, clubs, diamonds, spades: Players who suit MUDs. *Journal of MUD research*, v. 1, n. 1, p. 19, 1996.

MARCZEWSKI, Andrzej. (2015). Even Ninja Monkeys Like to Play: Gamification, Game Thinking and Motivational. CreateSpace Independent Publishing Platform, 2015, 220 p.

WERBACH, Kevin; HUNTER, Dan. *The Gamification Toolkit: Dynamics, Mechanics, and Components for the Win*. Estados unidos: Wharton Digital Press, 2015, 47 p.

BURKE, Brian. *Gamificar: como a gamificação motiva as pessoas a fazerem coisas extraordinárias*. São Paulo: DVS, 2015. 192 p.

FREITAS, Sérgio A. A. et al. Gamificação e avaliação do engajamento dos estudantes em uma disciplina técnica de curso de graduação. In: *Simpósio Brasileiro de Informática na Educação - SBIE*. 2016. p. 370.

ALMEIDA, Cíntia et al. Avaliação do processo de Gamificação acerca do tema Direitos Humanos. In: *Anais do Workshop de Informática na Escola*. 2016. p. 379.

ALBINO, João Pedro et al. Gamification em Ambientes Virtuais de Aprendizagem no Ensino Superior Presencial. In: *Congresso Internacional de Educação e Tecnologias: Encontro de Pesquisadores em Educação a Distância*, 2018.

TODA, Armando M.; VALLE, Pedro H. D.; ISOTANI, Seiji. The Dark Side of Gamification: An Overview of Negative Effects of Gamification in Education. In: *Researcher Links Workshop: Higher Education for All*. Springer, Cham, 2017. p. 143-156.

KOIVISTO, Jonna; HAMARI, Juh. The rise of motivational information systems: A review of gamification research. Working paper, 2017.

**APÊNDICE D – AN IMPLEMENTATION METHODOLOGY OF BIOLOGICAL COLLECTIONS
TO SPECIFY BASED ON IRPM**

An Implementation Methodology of Biological Collections to Specify Based on IRPM

José R. M. Alves ¹, David Prata ¹, Patrick Letouze ¹

UFT – Federal University of Tocantins, Brazil ¹

Abstract. Biological collections are collections of organisms, or parts thereof, organized in order to provide information from their origin, collection and identification of each of their specimens. Collections may accumulate sets of knowledge generated over time, a comprehensive geographic and temporal landscape representing the cultural heritage and historical wealth of ecosystems. These records have to be codified and organized in associated data sets, in order to provide information. The purpose of this work is to implement collections in databases by generating material to be used in academic courses and by increasing the scientific productivity in research institutions. As a result, an implementation methodology is proposed for the importation of biological collections to a database management software based on IRPM - Interdisciplinary Research Project Management. IRPM is a methodology for developing interdisciplinary research based on project management to maximize research productivity.

Palavras-chave: Specify, IRPM, Biological Collections, Specimen, and Taxonomy.

1. Introduction

Biological collections are defined as a set of fossil or current organisms, and may be complete or only part of them, properly preserved and cataloged for the purpose of didactic-scientific studies. They function as a starting point for ecologists, zoologists, botanists, and other specialists to study the diversity of a region, from the information and identification of its objects of study.

Biological collections are one of the most important tools for obtaining information on the composition, distribution and content of biodiversity in a given environment. In addition to scientific research, the information contained in the collections may have other uses, such as subsidizing decision-making by public authorities on issues of biodiversity conservation, use of the country's natural resources and land use planning.

As cumulative sets of knowledge generated over time, collections are also a comprehensive geographic and temporal landscape, representing the cultural heritage and historical wealth of a given territory. For this, the records of the biological collections are codified and organized, in order to provide information such as origin, collection and identification.

Specify is a software dedicated to the cataloging and management of zoological and botanical collections. Created by the University of Kansas, United States, it enables cataloging, research, loan management, reporting, and other usual activities in the management of a collection. Specify is free and open source software used by 404 collections in 43 US States and 26 countries worldwide (Coetzer et al., 2012).

Field research of biological collection are costly and time-consuming. The performance and decisions the scientists exhibit and make to their effort can increase research productivity, recognition, and impact to their results. Scientists seeking to establish their works should elaborate strategies to publish a mix of short and long papers in highest quality journals. Researches should have the potential to generate several new and exciting results for each methodology and findings invested in setting up each field study undertake. The IRPM (Letouze, 2011a) is a methodology for developing interdisciplinary research based on project management that intends to maximize academic results. One example that uses IRPM was presented in (Letouze, 2011b), which is a simulation of a software house in a classroom to develop a system for academic project management.

This work has the purpose to introduce a methodology for importing collections into the Specify software based on IRPM, in order to improve the academic productivity by the management of collections in institutions.

2. Methodology

2.1. Collections Importation

According to the Specify Software Project (2013), Specify supports the importation of large collections of specimens from spreadsheets, in accordance with the steps process showed in Figure 1.

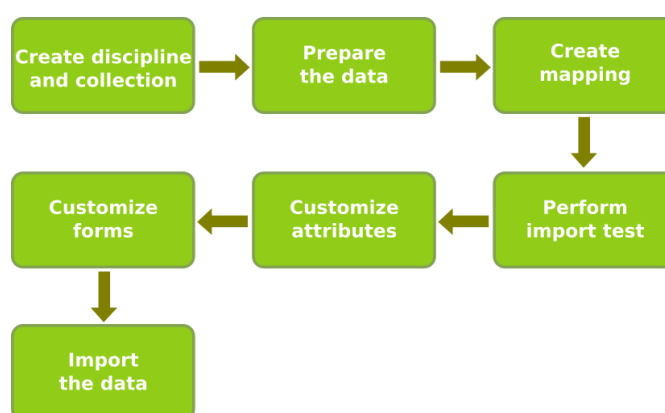


Figure 1 – The steps process to import data in Specify.

The disciplines and collections are organized in a hierarchical schema, the collections being subtrees of the disciplines. Therefore, one discipline can have one or many collections. Among others profiles, the disciplines define the type of the collection, like ornithology, botany, and others. If the type of collection is not already registered, it is necessary to create it. After all, the biological collection may be inserted, Figure 2.

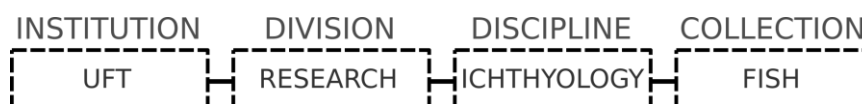


Figure 2 – Hierarchical example of a collection in Specify

The second step is to group all the data into a single worksheet and split it into smaller files with a maximum of 4000 rows because it is the line quantity boundary imposed by the program for each import.

The mapping created in the third step is a match performed between the columns of the worksheet with the data to be stored in Specify, where each data of the worksheet will be stored in Specify, Figure 3. Probably there will not be correspondence for all the columns of the worksheet. In these cases, a generic field available in Specify may be used for personalization.

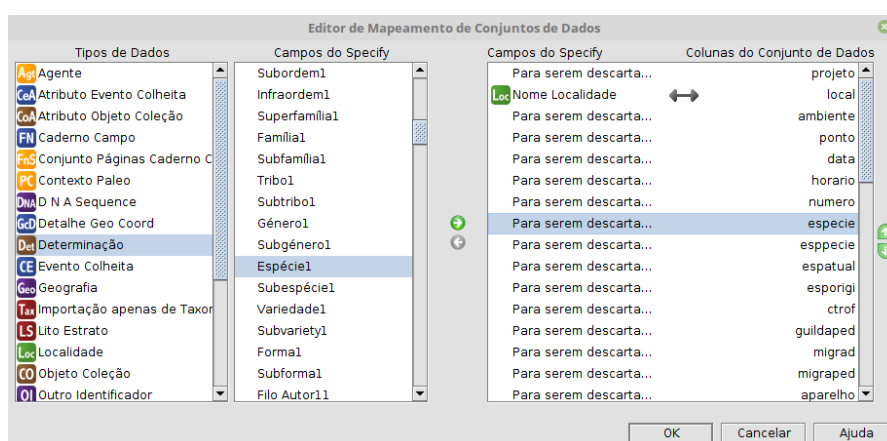


Figure 3 – Data mapping.

The fourth step is a recommendation, because carrying out an import test before importing all the data, allows us to check for errors, allowing them to be corrected before the final importation.

The fifth and sixth steps relate to the generic fields used in the third step, in the fifth change the display names of the attributes and in the sixth change the forms so that they are displayed. Finally, in the seventh step the data is imported.

2.2. Interdisciplinary Research Projects Management

The IRPM applies the 5 phases of the PMBOK (Project Management Body of Knowledge) project management concepts pointed out by Heldman (2009):

- **Initiation:** to determine project objectives, process deliveries and outputs, determine project constraints and assumptions, define strategy, identify performance criteria, resource requirements, set expenditures, and produce formal documentation;
- **Planning:** to refine the project, create the Analytical Framework for the Project (AFP), develop the resource management plan, refine time and cost estimates, stabilize project controls, develop the project plan, and obtain project approval;
- **Execution:** to commit resources, to implement resources, to manage progress, to communicate progress and to implement quality assurance procedures.
- **Control:** to measure performance, to refine control limits, to take corrective action, to evaluate effectiveness of corrective action, to ensure plan compliance, to reassess control plans, to respond to risk event triggers and to monitor project activity.
- **Closing:** to obtain acceptance of deliverables, to document lessons learned, to facilitate closure, to preserve product records and tools, and to release resources.

In IRPM – Figure 4, Initiation phase starts from choosing the real problem to solve and identifying at least two fields for an interdisciplinary approach to seek its solution. These fields are necessary to document the real problem constraints and assumptions, to define strategy, to identify performance criteria, to determine resource requirements, to define the budget and to produce a formal documentation.

The Planning phase consists of refining the project and of doing a more profound study of the problem by using the chosen fields. These studies may produce a new fundamental or methodology. Therefore, in the Execution phase, even if new concepts are not generated, an educational material may be prepared and applied in a classroom for a Problem-Based Learning (PBL) approach, which is an instructional learner centered approach to help students in conducting research that integrates theory and practice, to develop a viable solution to a defined problem [6]. Still in the Execution phase, if a new fundamental or methodology was developed, then a new technology may be implemented and used.

Moreover, if in the Planning phase controls were established then educational, technological, economics and social parameters may be available for measurement, allowing the Control phase to be performed. Finally, after analyzing the measurements results, papers should be written, and that is the Closing phase.

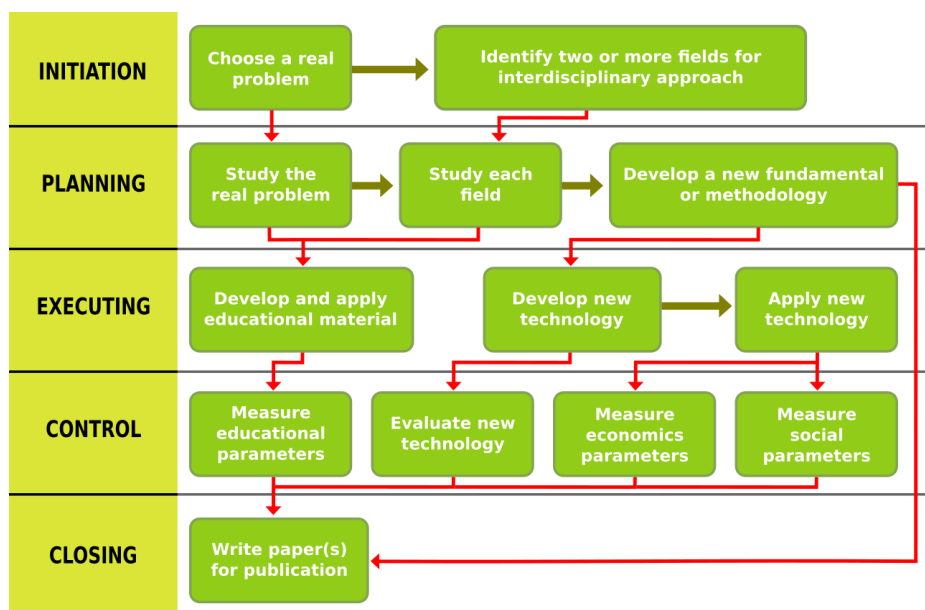


Figure 4 - Management Model of Interdisciplinary Research Projects

3. Resultados

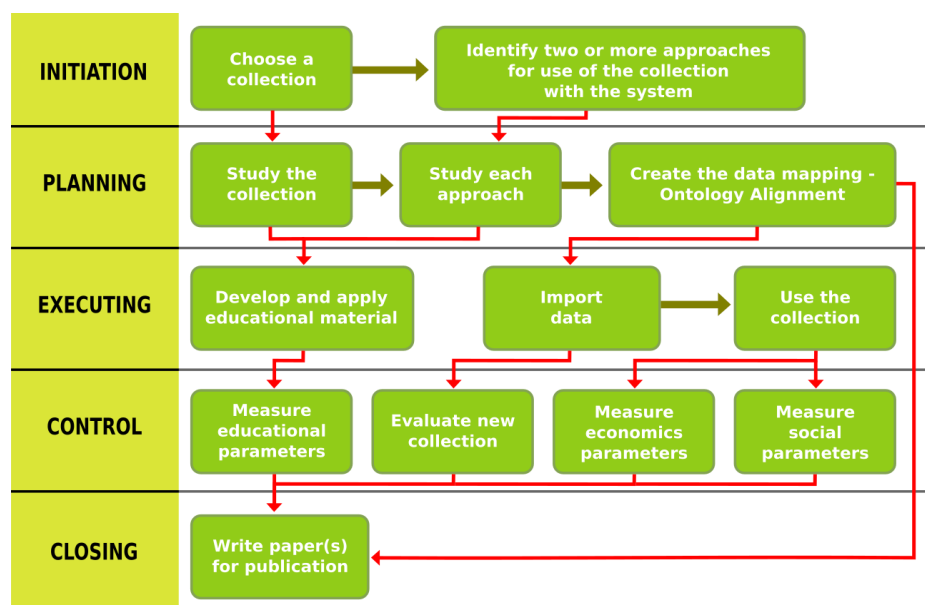


Figure 5 – Implementation methodology of biological collections to Specify based on IRPM.

Applying the IRMP methodology to implementation of biological collections to Specify established a strategy for managing ecosystems ecology projects, as shown in Figure 3. The initiation phase begins with the choice of collection, and then two or more approaches to using the collection with the system should be chosen, because although Specify is designed to manage stored collections of specimens as in

a museum or university, it also can be used for other purposes like database for research, cataloging, among others.

In the planning phase, the collection is studied in accordance to each approach chosen, to then perform the data mapping. The data mapping use techniques like ontology alignment. Ontology alignment, or ontology matching, is the process of determining correspondences between concepts in ontologies. This phase is strategic in order to decide on what will be the research objectives to the specific database collection studies. One example is to connect a botanical ontology with other ontologies such as PO (plant ontology, which connects plant anatomy, morphology, plant growth and development to plant genotype data) or plant phenotypic traits ontology (PTO), and many others, which express the myriad structures of relationships between humans and plants (e.g., scientific taxonomies, local medicine, etc.) (Aravind et al., 2014). By default, the mapping schema of Specify organizes outgoing copies of specimen records to produce data formats needed by other projects. For example, the specimen records are organized as Taxonomic Databases Working Group (TDWG) Darwin Core (DwC) archive files for publication at Global Biodiversity Information Facility (GBIF).

In the execution phase, educational material is generated, the data is imported and the collection begins to be used. In the control phase the results are obtained and analyzed. Finally, in the closing phase the articles are written.

4. Final Considerations

The proposed methodology intends to be a platform that facilitates future systematic reviews and meta-analysis. When data types, procedures and their sequence are standardized, different purposes can be strategically planned to maximize research methods for research productivity, and minimizing experimentation discrepancies in execution. In addition, the methodology highlight how sequencing may affect results.

Only a clear and well defined methodology allows reproducibility of an experiment, and to use IRPM helps to accomplish that. However, to develop a research following all the steps laid down by IRPM requires a longer time for planning activities, a greater amount of data is gathered, and more documents are stored, not to mention the need for training beginners.

Documents stored throughout the development of implementation collections to Specify based on IRPM may generate classroom material for educational purposes. Hence, the elaboration of such material is a future activity that may be applied in a classroom using a problem-based learning approach to generate papers for submission.

The Project-Based Learning (PBL) is one of the methodologies available to use collections in classrooms. According to Boss and Krauss (2014), the PBL requires that students seek data to solve the problems defined. In this way, Specify collections can be used as a research source.

5. References

Aravind Venkatesan, Pierre Larmande, Clement Jonquet, Manuel Ruiz & Patrick Valduriez. Facilitating efficient knowledge management and discovery in the Agronomic Sciences, In 4th Plenary Meeting of the Research Data Alliance. Amsterdam, The Netherlands, September 2014.

BOSS, Suzie; KRAUSS, Jane. **Reinventing Project-Based Learning: Your Field Guide to Real-World Projects in the Digital Age**. 2° ed. International Society for Technology in Education: Washington, 2014.

COETZER, Willem; GON, Ofer; HAMER, Michelle; PARKER-ALLIE, Fatima. **A new era for specimen databases and biodiversity information management in South Africa**. Biodiversity Informatics, 2012, vol. 8: 1-11.

LETOUZE, Patrick. **Interdisciplinary Research Project Management**. International Proceedings of Economics Development and Research, 2011, vol. 14: 338-342.

LETOUZE, P., R. A. Ronzani and A. H. M. Oliveira, An Academic Project Management Web System Developed through a Software House Simulation in a Classroom, IPEDR, 2011, vol. 14: 587-592.

HELDMAN, Kim. **PMP Project Management Professional Exam Study Guide**. 5° ed. John Wiley and Sons: Hoboken. 2009.

Specify Software Project. **Complete Specify Help**. 2013. Disponível em: <http://www.sustain.specifysoftware.org/wp-content/uploads/2017/03/complete_specify_help.pdf>.

**APÊNDICE E – IMPLEMENTAÇÃO DE ALGORÍTMO NSGA-III COM OPERADOR DE
ESTIMATIVA DE DISTRIBUIÇÃO**

Implementação de algoritmo NSGA-III com operador de estimativa de distribuição

Francirley Resendes Borges Costa ¹, José Robson Mariano Alves ¹

¹Instituto Federal de Ciência e Tecnologia do Tocantins (IFTO)

Abstract. *This work presents the optimization algorithm for many objectives NSGA-III (Genetic Algorithm of Unclassified Classification) with the evolutionary estimation distribution algorithm (EDA). In the work development was proposed a way for obtaining and preserving solutions, guided by reference points. As selection operator was chosen by EDA with creation of probabilistic models using normal distribution around the projection distances between the individuals of the search space and reference points. The results and comparisons with the original implementations of the NSGA-III and the MOEAD/PBI are presented in the scope of the paper, a proposal was inferior to the results defined by the idealizers of the other methods. At the end a critical analysis is carried out, as well as proposals for improvements for future work.*

Resumo. *Este trabalho apresenta o algoritmo de otimização de muitos objetivos NSGA-III (Non-dominated Sorting Genetic Algoritm) com o método de seleção evolucionário de estimativa de distribuição (EDA). No desenvolvimento do trabalho foi proposto um mecanismo de obtenção e preservação de soluções, orientado por pontos de referência. Como operador de seleção foi escolhido o EDA com criação de modelos probabilísticos usando distribuição normal em torno das distâncias de projeção entre os indivíduos do espaço de busca e pontos de referência. No decorrer do trabalho são apresentados os resultados e comparações com as implementações originais do NSGA-III e MOEAD/PBI, a proposta mostrou-se inferior frente aos resultados apresentados pelos idealizadores dos outros métodos. Ao final uma análise crítica é realizada, bem como propostas de melhorias para trabalhos futuros.*

1. Introdução

Métodos de otimização evolucionária multi/muitos objetivos (*Evolutionary multi/many-objective optimization* - EMO) surgiram no início dos anos noventa e desde então é crescente a quantidade de pesquisadores nessa área. Impulsionado cada vez mais pela evolução dos computadores, sua popularização também aconteceu na área industrial com o aparecimento de softwares EMO comerciais. No cotidiano os métodos EMO tem demonstrado cada vez mais capacidade para resolver problemas com muitos objetivos. Os métodos de otimização multi/muitos objetivos têm basicamente dois conceitos principais: diminuir a distância entre a frente não dominada e a frente pareto-ótimo e encontrar um conjunto de soluções diversificadas. Já os algoritmos genéticos/evolucionários (AEs) buscam solucionar os problemas de seleção e geração das soluções de forma a garantir uma busca eficaz para o conjunto pareto-ótimo. O foco principal das pesquisas com EMO é buscar maneiras de garantir a manutenção de soluções distintas e mais próximas da

fronteira pareto-ótimo durante o processo evolucionário, o que mantém um bom espalhamento das soluções [Barbosa et al. 2010]. Com base na característica dinâmica dos processos evolucionários, este trabalho propõe a implementação de um EMO fundamentado no NSGA-III [Deb et al. 2005] com operador de geração de soluções EDA. Nesse contexto, o algoritmo evolucionário NSGAEDA foi desenvolvido. Na seção II é apresentada uma revisão da literatura sobre alguns dos principais algoritmos de EMO. Na seção III a estrutura do algoritmo proposto e suas características são apresentadas. Na seção IV são apresentados os problemas usados para testes com o algoritmo proposto. Na seção V os resultados obtidos são confrontados e analisados em comparação com os algoritmos NSGA-III e MOEA/D/PBI. Por fim, na sessão VI são apresentadas as conclusões e uma análise crítica do modelo proposto, bem como melhorias não exploradas para trabalhos futuros.

2. Algoritmos de otimização multi/muitos objetivos

Nesta seção são apresentados dois algoritmos EMO: NSGA-III (*Non-dominated Sorting Genetic Algorithm*) e o MOEA/D [Deb and Jain 2014], ambos são ditos como estado da arte em métodos de otimização multiobjetivo.

2.1. NSGA-III

Non-Dominated Sorting Genetic Algorithm (NSGA III) [Deb et al. 2005], proposto para tentar diminuir a complexidade computacional pela técnica de classificação por não-dominância, que faz o elitismo com base em frentes de soluções ditas não-dominadas. O algoritmo NSGA III (Figura 1) é uma evolução do NSGA-II e trabalha com uma população filha, criada a partir de uma população pai inicial, utiliza um operador de seleção/cruzamento/mutação e mantém sua diversidade por meio de um mecanismo de preservação de nichos que busca soluções próximas da projeção criada por pontos de referência.

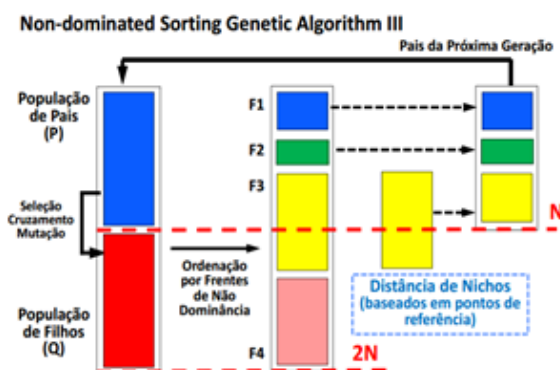


Figura 1. Algoritmo NSGA-III

Os pontos de referência buscam dar a direção para manter a diversidade das soluções de busca e também podem dar suporte para convergência do algoritmo. O próprio usuário pode gerar os pontos de referência, bem como eles podem ser gerados com base em um ponto ideal balanceando os objetivos [Deb et al. 2005].

Para a preservação de soluções é criado um mecanismo de preservação de nichos que através da distância de projeção do espaço normalizado das soluções para o ponto

de referência, associa cada indivíduo da população a um desses pontos. Em seguida os nichos com o menor número de soluções associadas são favorecidos [Deb et al. 2005].

2.2. MOEA/D

O algoritmo MOEA/D (*Multiobjective Evolutionary Algorithm Based in Decomposition*) utiliza a técnica de decompor um problema em subproblemas. O MOEA/D decompõe o problema em N subproblemas escalares de otimização, resolvendo-os simultaneamente. Isto é feito a partir da utilização de um conjunto de N vetores de peso, destinados a obtenção e manutenção de um conjunto diverso de soluções "trade-off" em paralelo. Inicialmente, cada membro da população (N) é associado a um dos vetores de ponderação aleatoriamente [Deb and Jain 2014]. Para tornar o problema escalar, o algoritmo utiliza-se de métodos de agregação de funções que podem ser uma simples soma ponderada. A soma ponderada torna um problema multiobjetivo de minimização em um problema escalar, conforme:

$$\min g^{soma}(x|\lambda) = \sum_{i=1}^{nf} \lambda_i f_i(x) \quad (1)$$

Onde g^{soma} é a nova função a ser otimizada, λ é o vetor de pesos associados aos objetivos, f , do problema multiobjetivo. A única condição imposta aos pesos é a de que $\sum_1^{nf} \lambda_i = 1$. As relações de vizinhança entre os subproblemas são definidas pelas distâncias euclidianas entre os vetores de ponderação. Para cada geração, as soluções associadas aos vetores vizinhos são usadas para geração de novas soluções, conforme a Figura 2.

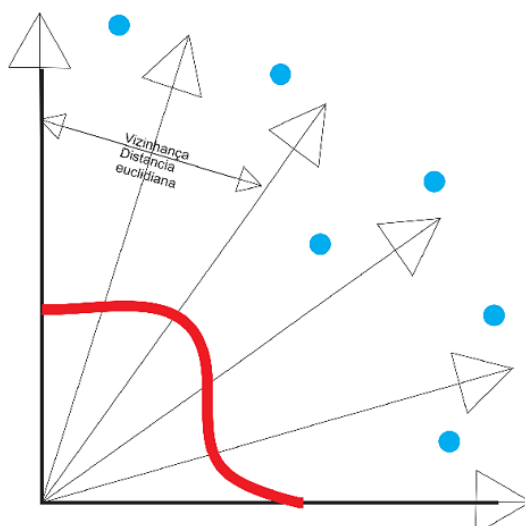


Figura 2. Relação de vizinhança

O Algoritmo 1 mostra o procedimento utilizado pelo MOEA/D para otimizar uma população de soluções.

Algoritmo 1: Algoritmo MOEA/D

início

1. Defina $EP = \emptyset$.
2. Calcula a distância euclidiana entre dois vetores de peso e retorna aqueles T vetores de peso mais próximos de cada vetor. Para cada $B(i) = i_1, \dots, i_T$ onde $\lambda^{i_1}, \dots, \lambda^{i_T}$ são os T vetores de peso associados a λ^i .
3. Gera a população inicial x^1, \dots, x^N randomicamente, e defina $FV^i = F(x^i)$.
4. Inicializa o ponto ideal $z = \{z_1, \dots, z_m\}^T$ (menor valor de objetivo em cada direção).

para $i=1$ até N **faça**

5. Reprodução: Randomicamente elege dois índices k e l de $B(i)$ e gera uma nova solução y de x^k e x^l por uso de operadores genéticos.
6. Atualiza ponto ideal z : Para cada $j = 1, \dots, m$ se $z_j < f_i(y)$, define $z_j = f_i(y)$
7. Atualiza soluções de vizinhança: Para cada índice $j \in B(i)$ se $g^{PBI}(y|\lambda^j, z) \leq g^{PBI}(y|x^j, z)$, então define $x^j = y$ e $FV^j = F(y)$
8. Atualiza EP: Remove todos os vetores dominados por $F(y)$ e adiciona $F(y)$ se nenhum vetor em EP domina $F(y)$

fim**fim**

O algoritmo inicializa o arquivo de soluções não-dominadas EP e o ponto ideal z . Em N gerações, para cada solução em P , dois indivíduos pais x_n^1 e x_n^2 são selecionados da vizinhança e uma nova solução y é gerada através de operadores genéticos/evolucionários. Avaliado o objetivo y , o ponto ideal z é atualizado com os menores valores para cada objetivo encontrado até o momento. Em seguida a vizinhança é atualizada substituindo por y os indivíduos que possuem valor de função de agregação maior que o novo indivíduo, considerando o seu respectivo valor de pesos. O arquivo de soluções não dominadas (EP) é atualizado removendo as soluções não dominadas, adicionando y se esta é não dominada. Por fim o processo é repetido [Deb and Jain 2014].

3. Algoritmo Proposto NSGAEDA

O algoritmo proposto baseia-se na estrutura do NSGA-III e usa um operador de estimativa de distribuição para gerar os filhos para a próxima geração. A proposta cria a nova geração de filhos através de um operador que usa estimativa de distribuição. O operador EDA proposto cria modelos probabilísticos com base em cada ponto de referência, calculando a distância de projeção entre os " $n = 4$ " (valor com melhores resultados) indivíduos mais próximos, por fim, com a média e desvio padrão obtidos pelo modelo é aplicada uma distribuição normal para gerar um novo filho.

3.1. Operador EDA

Para criar os modelos probabilísticos é necessária a informação de distância de projeção de todos os indivíduos, logo, como é proposto no NSGA-III, o algoritmo comporta-se como ilustra a Figura 3.

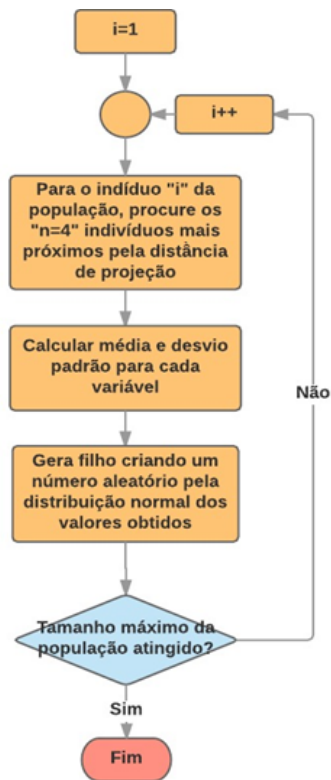


Figura 3. Operador EDA

Depois de gerados os filhos, uma perturbação chamada de mutação gaussiana é aplicada para ajudar na convergência do algoritmo. Aleatoriamente é gerado um valor entre o limite mínimo e máximo do espaço de busca do problema, em seguida, para cada valor de variável j de cada indivíduo X_i da população, verifica-se se o valor gerado é maior que o valor X_{ij} , se sim, o valor da variável em questão é definido como média e define-se um desvio padrão de $\frac{1}{l}$ sendo que l varia de 20 até 1000, de acordo com a evolução das gerações, quanto maior o valor de l , maior é a perturbação na variável, por fim, a mutação é aplicada.

3.2. Mutação Gaussiana

Dado $X_{ij} \in [a, b]$, a mutação gaussiana M_G é calculada por: $M_G(x) = \min(\max(N(x_{ij}, \sigma), a), b)$ onde $\sigma = \frac{1}{l}$ mede o tamanho da perturbação, a e b são os limites superiores e inferiores do espaço de busca e x_{ij} é o valor da variável j do indivíduo X_i .

3.3. Preservação de nichos

Para a preservação de nichos, após calcular a distância de projeção para cada ponto de referência, um vetor com a quantidade de soluções associadas é criado para cada ponto de referência, assim, para cada nicho/ponto de referência que não possui uma solução associada a ele, é escolhida a com menor distância ortogonal.

3.4. Distância de projeção

A distância de projeção d ou projeção ortogonal é calculada por:

$$d(s, w) = \left\| \frac{s - w^t s w}{\|w\|^2} \right\| \quad (2)$$

Onde s é o valor da função objetivo e w o ponto de referência, ambos no espaço normalizado. Assim é possível definir qual o ponto de referência mais próximo para cada solução, como ilustra a Figura 4.

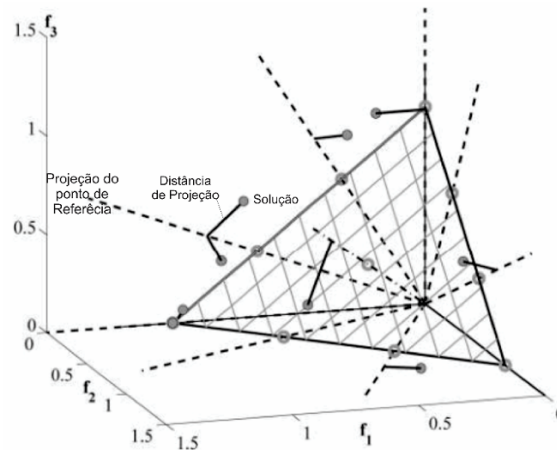


Figura 4. Distância de projeção

4. Resultados

Para medir os resultados do algoritmo proposto foram utilizados os testes de benchmark DTLZ1 e DTLZ2 [Deb et al. 2005] com 3 e 5 objetivos bem conhecidos da literatura. Além disso, o algoritmo foi comparado com o NSGA-III e MOEA/D-PBI, com os mesmos requisitos de testes, apresentados em [Deb et al. 2005]. Primeiramente são apresentados os gráficos com a diversidade de distribuição das soluções sobre a fronteira pareto. Para comparar o desempenho foi usado o IGD (*Inverted Generational Distance*) que demonstra o quão distante os elementos da fronteira pareto real estão da fronteira pareto estimada produzida pelo algoritmo [Zhang and Li 2007]

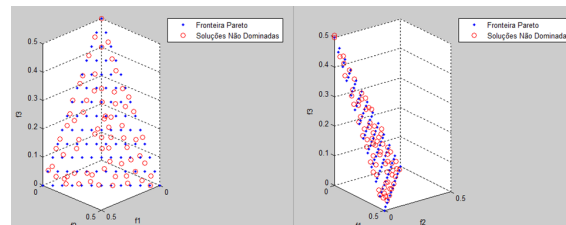


Figura 5. DTLZ1 com 3 objetivos

Como pode ser observado nas figuras 5, 6, 7 e 8, o algoritmo conseguiu uma boa diversidade, tanto nos problemas com 3 quanto nos com 5 variáveis, porém no DTLZ1 não houve uma maior homogeneidade na diversidade como houve no DTLZ2.

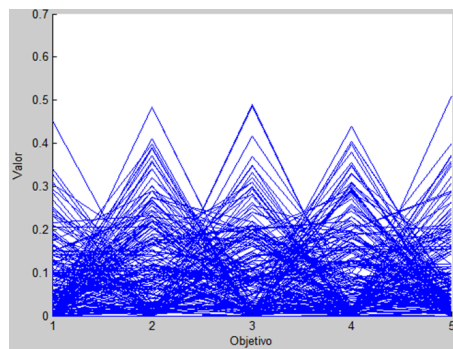


Figura 6. DTLZ1 com 5 objetivos

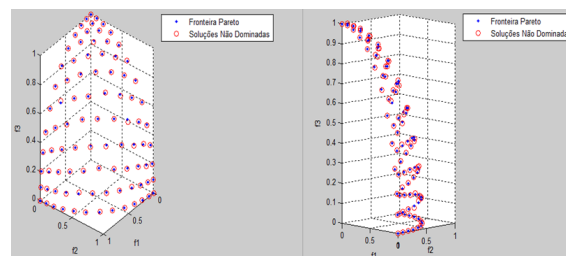


Figura 7. DTLZ2 com 3 objetivos

Os valores de IGD (tabela 1) obtidos demonstram o desempenho do algoritmo em alcançar a fronteira pareto-ótimo, percebe-se que esses valores não conseguiram superar os algoritmos do estado da arte, porém demonstraram certa estabilidade quanto a variação dos resultados obtidos entre melhor, média e pior caso.

5. Conclusão

A proposta de utilização de EDA como operador de geração de soluções obteve resultados levemente inferiores quanto ao estado da arte, porém comportou-se com certa estabilidade quanto as soluções obtidas. Percebeu-se também que para o problema DTLZ2 o resultado obtido foi próximo do estado da arte. Durante o processo de desenvolvimento do problema foram realizados testes com o uso de outros operadores de geração de soluções como o *Diferencial Evolution* (DE) e o Cruzamento Binário Simulado (*Simulated Binary Crossover* - SBX), com os quais melhores resultados foram obtidos, provavelmente pela abordagem do operador, assim percebe-se que este ainda pode ser refinado. Para geração de modelos probabilísticos, investigar outras soluções também poderia mostrar resultados melhores, por exemplo, identificar melhor as soluções vizinhas usando proximidade vetorial, ou verificar dominância entre as soluções e identificar a direção em relação ao ponto de referência, podendo assim definir pontos que circundam os pontos de referência. Com base nos resultados, recomenda-se para trabalhos futuros uma abordagem de geração de prole com um algoritmo híbrido, visto que EDA, DE e SBX comportam-se de forma diferente para problemas de mesma natureza.

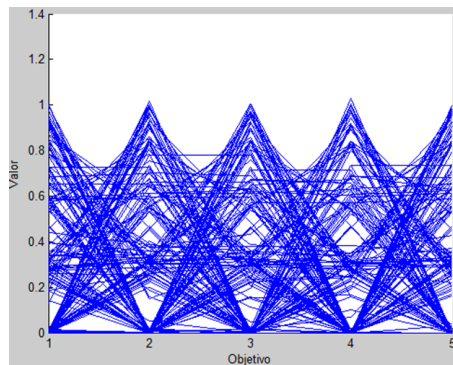


Figura 8. DTLZ2 com 5 objetivos

Tabela 1. Valores de Avaliação IGD

DTLZ	IGD	MOEAD/DD	NSGA-III	EADNS
DTLZ1 m = 3	Melhor	3,19E-04	4,88E-04	2.64E-02
	Média	5,84E-04	1,31E-03	2.77E-02
	Pior	6,57E-04	4,88E-03	2.94E-02
DTLZ1 m = 5	Melhor	2,63E-04	5,12E-04	5,82E-02
	Média	2,92E-04	9,79E-04	6,15E-02
	Pior	3,11E-04	1,98E-03	6,34E-02
DTLZ2 m = 3	Melhor	6,66E-04	1,26E-03	5,42E-03
	Média	8,07E-04	1,36E-03	6.01E-03
	Pior	1,24E-03	2,11E-03	6.46E-03
DTLZ2 m = 5	Melhor	1,13E-03	4,25E-03	4,91E-02
	Média	1,29E-03	4,98E-03	5,42E-02
	Pior	1,42E-03	5,86E-03	6,44E-02

Referências

- Barbosa, A. M., de Carvalho Ribeiro, L., and de Oliveira Arantes, J. M. (2010). Algoritmo genético multiobjetivo: sistema adaptativo com elitismo. *Citado*, 2:16.
- Deb, K. and Jain, H. (2014). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints. *IEEE Trans. Evolutionary Computation*, 18(4):577–601.
- Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. (2005). *Scalable test problems for evolutionary multiobjective optimization*. Springer.
- Zhang, Q. and Li, H. (2007). Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731.

**APÊNDICE F –DESEMPENHO PRODUTIVO E TEOR DE AMIDO DE VARIEDADES DE
MANDIOCA EM PLINTOSSOLO NA REGIÃO SUDOESTE DO TOCANTINS**



Desempenho produtivo e teor de amido de variedades de mandioca em plintossolo na região sudoeste do Tocantins

Jardel Barbosa dos Santos¹, Ubiranei de Freitas Marinho¹, José Robson Mariano Alves², Plínio Cardoso de Oliveira², Luana Carvalho dos Santos³, Gisele Mendanha Nascimento⁴.

¹ Professor do curso de Técnico em Agricultura do IFTO - C.A. Lagoa da Confusão. e-mail: <jardel.santos@ifto.edu.br>; <ubiranei.marinho@ifto.edu.br>

² Professor do curso de Técnico em Informática do IFTO - C.A. Lagoa da Confusão. e-mail: <jose.alves@ifto.edu.br>; <plinio.oliveira@ifto.edu.br>

³ Discente do curso de Técnico em Agricultura do IFTO - C.A. Lagoa da Confusão. Bolsista-proex. e-mail: <luanaimpossivel@gmail.com>

⁴ Autônoma. E-mail: <gisele_zoo@hotmail.com>

Resumo: Cultivada em todas as regiões brasileiras, é no norte e nordeste brasileiro que se concentra o maior consumo de mandioca e seus derivados. Nestas regiões, no entanto, existem poucos cultivos comerciais, devido principalmente a ausência de informações sobre o rendimento produtivo e industrial. Assim, avaliou-se nove variedades de mandioca com baixo teor de ácido cianídrico “mansa” no município da Lagoa da Confusão. Foram testadas sete variedades cedidas pelo grupo de pesquisa “Manirede”, coletadas regionalmente no Estado, e as outras duas coletadas, localmente, em dois produtores. Foram avaliados a altura da primeira ramificação, a massa da parte aérea, massa de raízes comerciais, a produtividade, a massa em água, o teor de amido e a matéria seca. Três variedades, Fécula 1, Fécula 2 e Mesa ARN apresentam maior altura à 1ª ramificação, e às variedades local 1 e olho junto apresentam menor massa de parte aérea. Maiores produtividades foram observadas nas variedades Fécula 1, Fécula 2, olho junto e Mesa ARN. Por outro lado, as variedades Pão, Pão preto, brancona, Mesa ARN e Local 2 apresentaram maiores massa em água, matéria seca e teores de amido. Observa-se que, apenas a variedade mesa ARN associou produtividade de raiz com qualidade industrial, ou seja, apresentou alta produtividade e alto teor de amido. Portanto, nas condições dessa região, a variedade mesa ARN pode ter maior probabilidade de retorno econômico para os agricultores.

Palavras-chave: *Manihot esculentun*, produtividade, rendimento industrial

1 INTRODUÇÃO

A mandioca (*Manihot esculenta* Crantz) é um arbusto pertencente a família Euphorbeaceae, da qual é a única, dentre as 98 espécies conhecidas, cultivada para fins de alimentação humana. A planta ancestral da mandioca é natural de vegetação de galeria associada a rios, na zona de transição entre a floresta Amazônica e o Cerrado, próxima às fronteiras entre Peru e Brasil (CARVALHO, 2005). Sendo assim, a mandioca é favoravelmente cultivada em climas tropicais (CARDOSO; SOUZA, 1999).

Dentre as principais culturas do Brasil, a mandioca destaca-se pela sua importância social e econômica, sendo cultivada em 2,3 milhão de hectares. Sua produção é destinada, principalmente, à fabricação de farinha, de fécula e ao consumo “in natura” (IBGE, 2017). Os Estados que veem se



destacando na produção da raiz são Pará, Bahia e Paraná em área plantada, e Pará e Paraná pela produção, alcançando 6.034.714 e 3.744.351 toneladas, respectivamente. Bons resultados estão sendo observados no Paraná com a consolidação da cadeia produtiva. Atualmente, o estado é o maior produtor de fécula de mandioca do Brasil, possuindo um parque industrial de 43 indústrias em 2002 (SOUZA; ANDRONIO; STADUTO, 2005).

Em contraste ao estado do Paraná, o Tocantins ainda é pouco expressivo no cultivo da mandioca, mesmo culturalmente utilizando a raiz como base alimentar. O Estado do Tocantins possui apenas uma indústria de fécula, localizada em Aparecida do Rio Negro e numerosas casas de farinha que atualmente são subutilizadas ou destinadas para outros fins. Além disso, a cadeia agrícola da cultura não está consolidada, sendo os pequenos agricultores responsáveis pelos distintos trabalhos na cadeia. Por exemplo, responsáveis pela produção da raiz, pela transformação, distribuição e comércio (JUCÁ, 2006; GRANJA JÚNIOR; SILVA; BOURSCHIEDT, 2010).

Não apenas no estado, porém em várias regiões do Brasil e do mundo, a baixa produtividade agrícola é apontada como principal entrave do cultivo. A desconstrução da ideia que a mandioca é cultura de “quintal” passa a ser importante, e ao ser discutida como cultura agrícola de alta rentabilidade pela intensificação é preciso compreender aspectos agrônômicos dos cultivos agrícolas (GNAHOUA et al., 2016). Assim, define-se que a cultura precisa ser planejada quanto a: escolha de variedades, planejamento do espaçamento, controle de pragas, doenças e plantas daninhas, época de plantio, adubação e calagem, e outros.

A escolha de variedades é baseada, inicialmente, no objetivo da produção agrícola. Para genótipos de mesa, ou seja, com baixo teor de ácido cianídrico, a informação do teor de amido da variedade é muito importante, devido sua correlação positiva com o seu rendimento industrial. Assim, obter informações sobre o teor de amido e rendimento produtivo da variedade a ser plantada pode permitir maiores lucros nas lavouras. Para o Estado do Tocantins são poucas as informações sobre o desempenho produtivo e industrial de mandioca em solos plínticos, muito comuns no estado. Além disso, é importante salientar, que avaliar os genótipos regionais contribui com o melhoramento participativo, no qual coloca-se em estudo, as variedades já cultivadas na região e que a maior parte dos produtores têm acesso (FUKUDA et al., 1997). Assim, objetiva-se avaliar o desempenho produtivo e a produção de amido de variedades regionais e locais de mandioca.

2 METODOLOGIA



O experimento foi realizado em PLINTOSSOLO PÉTRICO no município de Lagoa da Confusão, latitude = 10,804179°S, longitude = 49, 636707°W e altitude = 172 m. A área não apresentou necessidade de calagem e a adubação foi realizada com 30, 80 e 40 Kg/ha de N, P e K no plantio e 50 e 50 Kg/ha de N e K em cobertura, respectivamente. O plantio foi realizado no dia 18 de dezembro de 2016, com plantio manual, regime hídrico natural e capina manual no controle de plantas daninhas. Não houve necessidade de controle de pragas e doenças.

O experimento foi executado em delineamento em blocos ao acaso, com nove tratamentos e quatro repetições. A parcela foi constituída de cinco linhas de 7,0 m, com espaçamento entre linhas de 1,0 m e entre plantas 0,7 m. As variedades utilizadas foram cedidas parte pelo projeto de pesquisa “Manirede” (7) e parte por produtores locais (2). As características morfológicas das raízes variedades são apresentadas no quadro abaixo.

Tabela 1 – Descrição fenotípica das raízes das variedades de mandioca avaliadas em PLINTOSSOLO, no município da lagoa da Confusão, 2017

Variedades	Cor externa da raiz	Cor do córtex da raiz
Fécula 1	Branco	Amarelo
Fécula 2	Branco	Amarelo
Pão	Marrom	Roxo
Pão Preto	Marrom	Roxo
Brancona	Branco	Amarelo
Olho junto	Marrom	Branco
Mesa ARN	Marrom	Rosa
Local 1	Branco	Amarelo
Local 2	Marrom	Rosa

Foram avaliadas a altura da primeira ramificação, massa da parte aérea e massa da raiz em cinco plantas por parcela. A massa expressa constitui a soma das cinco plantas. A produtividade foi calculada com base na massa de cinco plantas e na área ocupada por elas (3,5 m²). Além disso, a produtividade foi corrigida em função do estande de plantas na colheita, segundo o modelo matemático (Produtividade Corrigida = Produtividade * (100 - Estande Médio) / (100 - Estande da parcela)). As variáveis relacionadas à qualidade industrial foram avaliadas pelo método da balança hidrostática (GROSSMAN; FREITAS, 1950), pesando-se 3 kg de raiz e a submergindo em água,



para obter a massa em água. O teor de massa seca (%) foi estimado pela equação: $MS (\%) = 15,75 + 0,0564 * \text{Massa em água}$, e o teor de amido estimado sendo a $MS - 4,65$.

Os resultados foram submetidos à análise de variância em modelo linear generalizado com todas as pressuposições da ANOVA testadas pelo software livre R (R DEVELOPMENT CORE TEAM, 2011). As variáveis que apresentaram efeito significativo foram comparadas pelo método Scott-Knott com o pacote “ExpDes” (FERREIRA; CAVALCANTI; NOGUEIRA, 2011).

3 RESULTADOS E DISCUSSÕES

Foi observada diferença significativa nos componentes morfológicos da parte aérea e altura da primeira ramificação ($F_{1,35} = 15,5$ e $P > F = <0,0001$) e massa da parte aérea ($F_{1,35} = 3,96$ e $P > F = 0,0054$). Maiores alturas da primeira ramificação foram observadas nos genótipos Fécula 1, Fécula 2 e Mesa ARN (Tabela 1). É compreendido que quanto maior a altura da primeira ramificação maior é a prolificidade para plantios mecanizados, já que na plantadeira, quanto mais comprida a rama, maior o rendimento operacional do equipamento (FIALHO; ANDRADE; VIEIRA, 2009). As variedades locais avaliadas não são promissoras para o plantio mecanizado.

Tabela 1 – Altura da 1ª ramificação, massa da parte aérea, massa da raiz e produtividade de diferentes variedades de mandioca em PLINTOSSOLO PÉTRICO na Lagoa da Confusão, 2017

Variedades	Altura 1ª Ramificação (m)		Massa Parte Aérea (kg)		Massa Raiz (kg)		Produtividade (ton/ha)	
	Média	EP	Média	EP	Média	EP	Média	EP
Fécula 1	1,86 ±	0,13a	5,62 ±	0,95b	10,92±	0,83a	33,10 ±	1,88a
Fécula 2	1,74 ±	0,08a	6,34 ±	0,66b	12,27 ±	0,97a	34,25 ±	2,34a
Pão	0,90 ±	0,06b	8,47 ±	0,52a	6,38 ±	0,66b	19,14 ±	1,91b
Pão Preto	0,86 ±	0,02b	9,60 ±	0,42a	5,25 ±	0,34b	13,99 ±	0,78b
Brancona	1,08 ±	0,09b	7,88 ±	1,52a	8,84 ±	1,59b	27,38 ±	4,33b
Olho junto	0,93 ±	0,59b	6,74 ±	2,50b	14,59 ±	2,69a	40,63 ±	11,28a
Mesa ARN	1,55 ±	0,14a	6,17 ±	0,94b	9,49 ±	0,57b	29,48 ±	0,73a
Local 1	0,41 ±	0,02b	4,07 ±	0,90b	8,85 ±	1,74b	24,40 ±	3,71b
Local 2	0,72 ±	0,03b	10,57 ±	2,36a	7,80 ±	1,80b	20,16 ±	4,81b
C.V.	21,82		27,83		21,36		21,00	
F (1,35)	15,02		3,96		6,55		7,1	
F > P	<0,0001**		0,0054**		0,0003**		0,0002**	



** = $P < (0,01)$; NS = não significativo. A massa expressa a soma de cinco plantas.

Contraoando a altura da primeira ramificação, quanto mais ramificada a variedade, maior a probabilidade desta apresentar alto peso da parte aérea (FUKUDA et al., 2006). Assim, as variedades que apresentaram maior altura da primeira ramificação tiveram menor massa da parte aérea, como as variedades Fécula 1, Fécula 2 e Mesa ARN e Olho junto (Tabela 1). Exceto, o genótipo Local 1, que apresenta pouca altura e massa da parte aérea por ser uma variedade de baixo porte.

Massa de raiz significativamente maiores ($F_{1,35} = 6,55$ e $P > F = 0,0003$) foram observadas nas variedades Fécula 1, Fécula 2, e Olho junto. Essas mesmas variedades junto com a variedade Mesa ARN, apresentaram as maiores produtividades. Foi alcançada produtividade de $40,63 \pm 11,28$ ton/ha na variedade Olho junto, $33,10 \pm 1,88$ ton/ha na variedade Fécula 1, $34,25 \pm 2,34$ ton/ha na variedade Fécula 2, e $29,48 \pm 0,73$ ton/ha na variedade Mesa ARN (Tabela 1).

Foram observados efeitos significativos sobre todas as variáveis relacionadas ao rendimento industrial (Tabela 2), massa em água ($F_{1,35} = 5,89$ e $P > F = < 0,0001$), teor de amido ($F_{1,35} = 3,32$ e $P > F = < 0,0001$) e matéria seca ($F_{1,35} = 2,80$ e $P > F = < 0,0001$). As maiores massas em água foram observadas nos genótipos Pão, Pão preto, Brancona, Mesa ARN e Local 2.

Tabela 2 – Massa em água, teor de amido e matéria seca de diferentes variedades de mandioca em PLINTOSSOLO PÉTRICO na Lagoa da Confusão, 2017

Variedades	Massa em Água (g)		Amido (%)		Matéria Seca (%)	
	Média	EP	Média	EP	Média	EP
Fécula 1	234,30 ±	2,79c	24,31 ±	0,16c	28,96 ±	0,16c
Fécula 2	233,70 ±	9,14c	24,28 ±	0,51c	28,93 ±	0,51c
Pão	287,60 ±	7,28a	27,32 ±	0,41a	31,97 ±	0,41a
Pão Preto	282,60 ±	6,05a	27,04 ±	0,34a	31,69 ±	0,34a
Brancona	278,40 ±	6,10a	26,80 ±	0,35a	31,45 ±	0,35a
Olho junto	250,80 ±	3,02b	25,25 ±	0,17b	29,90 ±	0,17b
Mesa ARN	272,40 ±	9,22a	26,46 ±	0,52a	31,11 ±	0,52a
Local 1	178,80 ±	4,65d	21,19 ±	0,26d	25,84 ±	0,26d
Local 2	275,70 ±	11,56a	26,65 ±	0,65a	31,30 ±	0,65a
C.V.	21,24		21,27		22,27	
F (1,35)	5,89		3,32		2,80	



F > P	<0,0001**	<0,0001**	<0,0001**
-------	-----------	-----------	-----------

** = P<(0,01); NS = não significativo. A massa expressa o peso em água de 3 kg de raiz.

Os maiores teores de massa seca e de amido também foram observados nas variedades Brancona, Pão, Pão preto, Mesa ARN e Local 2. Nota-se que a resposta das variáveis relacionadas ao rendimento industrial foram idênticas. Assim, em trabalhos futuros pode ser adotado apenas uma dessas variáveis para descrever a resposta do rendimento industrial. Menor teor de amido, ou seja, rendimento industrial foi observado na variedade Local 1, demonstrando que mesmo já sendo cultivada na região, para a obtenção de lucro é fundamental o conhecimento sobre a produtividade e rendimento industrial comparado com outras variedades.

4 CONSIDERAÇÕES FINAIS

Resultados promissores de produtividade foram encontrados nas variedades Fécula 1, Fécula 2, Olho Junto e Mesa ARN, e bem como altos teores de amido nas variedades Pão, Pão Preto, Brancona, Mesa ARN e Local 2. A variedade regional Mesa ARN destacou-se por apresentar três características importantes para o cultivo da mandioca sendo prolificidade, produtividade e rendimento industrial. Além disso, é importante salientar que conhecer as características produtivas e industriais são importantes para a solidificação da cadeia, para consolidar o melhoramento participativo e para proporcionar aumento da rentabilidade da mandiocultura na região.

REFERÊNCIAS

- CARDOSO, C. E. L.; SOUZA, J. S. **Aspectos agro-econômicos da cultura da mandioca:** Potencialidades e limitações. Cruz das Almas: Embrapa mandioca e Fruticultura Tropical, 1999.
- CARVALHO, L. J. C. B. Biodiversidade e biotecnologia em mandioca. XI Congresso Brasileiro de Mandioca. 2005, Campo Grande. **Anais...** Campo Grande: Embrapa.
- FERREIRA, E. B.; CAVALCANTI, P. P.; NOGUEIRA, D. A. Experimental designs: um pacote R para análise de experimentos. **Revista da Estatística UFOP**, v. 1, n. 1, p. 1–8, 2011.
- FIALHO, J. F.; ANDRADE, R. F. R.; VIEIRA, E. A. **Mandioca no cerrado:** Questões práticas. Planaltina: Embrapa Cerrados, 2009.
- FUKUDA, W.M.G.; MAGALHÃES, J.A.; CAVACANTI, J.; PINA, P.R.; TAVARES, J.A.; IGLESIAS, C.; HERNANDEZ, L.A.; MONTENEGRO, E.E. Pesquisa participativa em



melhoramento de mandioca: Uma experiência no semiárido do nordeste do Brasil. **Documentos 73**, Cruz das Almas, 46 p. 1997.

FUKUDA, W. M. G.; FUKUDA, C.; DIAS, M. C.; XAVIER, J. J. B. N.; FIALHO, J. F. Variedades. In: **Aspectos Socioeconômicos e Agronômicos da Mandioca**. Editor: Luciano da Silva Souza... [et al.]. – Cruz das Almas: Embrapa Mandioca e Fruticultura Tropical, 2006. p.433-454.

GNAHOUA, J. B. G. et al. Intensification Pathway for Improvement of Smallholder Cassava Production Systems in Southern Côte D'Ivoire. **Experimental Agriculture**, v. 53, n. 1, p. 44–58, 2016.

GRANJA JÚNIOR, J.; SILVA, T. T.; BOURSCHIEDT, J. R. O Estudo da competitividade da cadeia produtiva da mandioca no Tocantins : Uma análise comparativa com a cadeia produtiva agroindustrial do Paraná. V CONGRESSO NORTE-NORDESTE DE PESQUISA E INOVAÇÃO. **2010, Maceió , Anais...** Maceió: IFAL..

GROSSMAN, J.; FREITAS, A. G. DE. Determinação do teor de matéria seca pelo método de peso específico em raízes de mandioca. **Revista Agronômica**, v. 14, n. 160–162, p. 75–80, 1950.

IBGE. Instituto Brasileiro de Geografia e Estatística. **Previsão da safra 2016/2017**. Disponível em: <[http:// https://sidra.ibge.gov.br/tabela/1618#resultado](https://sidra.ibge.gov.br/tabela/1618#resultado)>. Acesso em: 09 de abril de 2017.

JUCÁ, J. V. **Mandioca**. SEAGRO/TO. Palmas: 2006.

SOUZA, E. F. DE; ANDRONIO, J.; STADUTO, R. A cultura da mandioca na região oeste do Paraná : Um estudo da coordenação da cadeia sob a ótica da teoria de contrato. **Organizações rurais e Agroindustriais**, v. 7, n. 1, p. 11–22, 2005.