

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE CIÊNCIAS APLICADAS A EDUCAÇÃO
DEPARTAMENTO DE CIÊNCIAS EXATAS
LICENCIATURA EM CIÊNCIA DA COMPUTAÇÃO

**Trivia, objeto de aprendizagem com gamificação para apoio à
disciplina de Introdução à Programação**

Mayara Wanessa Alves dos Santos
Orientadora: Prof^ª. MSc. Ana Liz Souto O. de Araújo

RIO TINTO - PB
2015

Mayara Wanessa Alves dos Santos

**Trivia, objeto de aprendizagem com gamificação para apoio à
disciplina de Introdução à Programação**

Monografia apresentada para obtenção do título de Licenciado à banca examinadora no Curso de Licenciatura em Ciência da Computação do Centro de Ciências Aplicadas e Educação (CCAEE), Campus IV da Universidade Federal da Paraíba.

Orientadora: Prof^a. MSc. Ana Liz Souto O. de Araújo.

S237t Santos, Mayara Wanessa Alves dos.

Trivia, objeto de aprendizagem com gamificação para apoio à disciplina de introdução à programação. / Mayara Wanessa Alves dos Santos. – Rio Tinto: [s.n.], 2015.

53 f. : il.

*Orientador(a): Prof. Msc. Ana Liz Souto O. de Araújo.
Monografia (Graduação) – UFPB/CCAÉ.*

1. Programação - computação. 2. Programação - estudo e ensino. 3. Ciência da computação.

UFPB/BS-CCAÉ

CDU: 004.021(043.2)

ASSINATURAS

Este trabalho é o resultado dos esforços da estudante Mayara Wanessa Alves dos Santos sob a orientação do Prof. MSc. Ana Liz Souto Oliveira de Araújo “Trivia, objeto de aprendizagem com gamificação para apoio à disciplina de Introdução à Programação” e conduzido no Centro de Ciências Aplicadas e Educação da Universidade Federal da Paraíba (CCAUE-UFPB). As pessoas listadas abaixo reconhecem o conteúdo deste documento e os resultados do Trabalho de Graduação.

Orientadora: Profa. MSc. Ana Liz Souto Oliveira de Araújo
Universidade Federal da Paraíba – Campus IV

Profa. DSc. Ayla Débora Dantas de Souza Rebouças
Universidade Federal da Paraíba – Campus IV

Profa. DSc. Flávia Veloso Costa Souza
Universidade Federal da Paraíba – Campus IV

Agradecimentos

Agradeço a Deus, primeiramente, por me dar forças e oportunidade de concluir esta etapa. A minha irmã Laryssa e ao meu irmão Williams, que apesar de não estar mais entre nós, torceram para que esse dia chegasse. Agradeço aos meus pais, Leudis e Claudia, por todo incentivo e apoio durante todo o curso, sempre acreditando no meu potencial sem medir esforços.

Aos amigos que conquistei, em especial, Clebia, Jakeline, Rafaela Fischer, Aline e Edilânio. Aos integrantes da casa das sete mulheres, agradeço todos momentos maravilhosos que partilhamos, momentos esses que tornaram essa caminhada melhor. Ao meu namorado Dorgival por toda paciência, companheirismo, e por toda força durante boa parte desta caminhada.

A minha orientadora Ana Liz, por ter acreditado na minha ideia e ajudado a desenvolvê-la, obrigada por todo apoio e conselhos em momentos importantes, por ter sido uma verdadeira amiga. Agradeço aos professores do Campus IV que compartilharam comigo seus conhecimentos e me guiaram nesta caminhada. Enfim a todos que de certa forma contribuíram para realização desse sonho.

RESUMO

Nos cursos da área de computação, devido à complexidade das suas disciplinas básicas, diversos problemas afetam os alunos iniciantes, prejudicando a permanência destes nos cursos. Por exemplo, é alta a complexidade de disciplinas em que se exige raciocínio lógico, entendimento de conceitos abstratos e um alto nível de abstração. Esses fatores podem ser determinantes para o aluno reprovar a disciplina ou até abandonar ou trocar de curso. Entre estas disciplinas pode-se citar as disciplinas de programação, que são vistas como difíceis e desestimulantes. Estas disciplinas possuem altos índices de reprovação e, um dos fatores que pode contribuir para esses índices são os métodos utilizados para o ensino. Buscando modificar o cenário de desmotivação no processo de ensino-aprendizagem de programação, este trabalho propõe um objeto de aprendizagem utilizando elementos de gamificação para o ensino de programação introdutória. Inicialmente foi realizada uma busca por objetos de aprendizagem voltados para a disciplina de Introdução à Programação. Após essa etapa, iniciou-se o planejamento e a prototipação do objeto de aprendizagem proposto. Por fim, avaliou-se o objeto considerando aspectos pedagógicos.

Palavras chave: Introdução à Programação, Objetos de aprendizagem, Gamificação

ABSTRACT

In the computing area courses, due to the complexity of its basic disciplines, many problems affect the beginners, damaging the permanence in the courses. For example, high complexity of disciplines which requires logical thinking, understanding of abstract concepts and a high level of abstraction. These factors can be decisive for the student to fail on the discipline or even abandon or change course. Among these disciplines, we can cite programming disciplines, which are seen as difficult and discouraging. These disciplines have high failure rates and one of the factors that may contribute to these indices are the methods used for teaching. Seeking to modify the motivation scenario in the teaching-learning process for programming, this paper proposes a learning object using gamification elements for the introductory programming education. Initially a search for learning objects focused on the subject Introduction to Programming was done. After this stage, we began the planning and prototyping of the proposed learning object. Finally, we assessed the object considering pedagogical aspects.

Keywords: Introduction to Programming, Learning Objects, Gamification

LISTA DE FIGURAS

Figura 1 Índice de reprovação das disciplinas nos períodos de 2011.1 a 2012.1	11
Figura 2 Índice de reprovação das disciplinas nos períodos de 2012.2 a 2013.2	12
Figura 3 Índices da disciplina de Introdução à Programação	14
Figura 4 Fase do Programe	29
Figura 5 Desafio do Programe	30
Figura 6 Tela do Objeto de Aprendizagem Klouro.....	31
Figura 7 Tela do Objeto de Aprendizagem Monitor.....	32
Figura 8 Árvore de habilidades do Trivia	34
Figura 9 Tela Insert Object do Construct 2D.....	35
Figura 11 Tela <i>Add Event</i> do Construct 2D	36
Figura 12 Tela inicial do Trivia	37
Figura 13 Menu do Trivia	37
Figura 14 Tela de lições relacionadas ao conteúdo variáveis	38
Figura 15 Tela de lições com explicação do Trivia	38
Figura 16 Tela de acerto do Trivia.....	39
Figura 17 Tela de erro do Trivia	40
Figura 18 Relatório do desempenho do usuário no Trivia	41

LISTA DE TABELAS

Tabela 1 Critérios de LORI, adaptado de LORI (2002)	26
Tabela 2 Avaliação do Trivia em relação a LORI (2002).....	42
Tabela 3 Avaliação do Trivia em relação a Reatengui, Boff e Finco (2010).....	44
Tabela 4 Avaliação do Trivia em relação a Reatengui, Boff e Finco (2010).....	45

LISTA DE SIGLAS

OA	Objeto de Aprendizagem
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
LCC	Licenciatura em Ciências da Computação

SUMÁRIO

1. INTRODUÇÃO	9
1.1 MOTIVAÇÃO.....	9
1.2 OBJETIVOS.....	16
1.2.1 OBJETIVO GERAL	16
1.2.2 OBJETIVOS ESPECÍFICOS	16
1.3 METODOLOGIA.....	17
1.4 ESTRUTURA DO TRABALHO.....	18
2. FUNDAMENTAÇÃO TEÓRICA.....	19
2.1 DIFICULDADES NO ENSINO DE PROGRAMAÇÃO	19
2.2 OBJETO DE APRENDIZAGEM.....	21
2.3 GAMIFICAÇÃO	22
2.4 AVALIAÇÃO DE OBJETOS DE APRENDIZAGEM	24
2.4.1 Proposta de avaliação LORI	24
3. TRABALHOS RELACIONADOS	29
3.1. PROGAME	29
3.2 ENTRANDO PELO CANO	30
3.3 KLOURO	31
3.4 MONITOR	32
4. TRIVIA	33
4.1 PLANEJAMENTO	33
4.2 RECURSOS DO CONSTRUCT 2D.....	35
4.2 DESCRIÇÃO DO TRIVIA	36
4.3. CONSIDERAÇÕES FINAIS	41
5. AVALIAÇÃO	42
6. CONCLUSÃO	46
SUGESTÕES DE TRABALHOS FUTUROS	46
REFERÊNCIAS BIBLIOGRÁFICAS	48

1. INTRODUÇÃO

Este capítulo apresenta as motivações para o trabalho, os objetivos, a metodologia utilizada no desenvolvimento e a estrutura deste documento.

1.1 MOTIVAÇÃO

O setor de Tecnologia da Informação e Comunicação (TIC) no Brasil tem apresentado crescimento frente a outras potências mundiais. De acordo com a BRASSCOM¹ (Associação Brasileira das Empresas de Tecnologia da Informação e Comunicação), o Brasil possui o 5º maior mercado interno de TIC do mundo, movimentando aproximadamente 123 bilhões de dólares (5,2% de participação no PIB nacional).

O setor apresenta taxa de crescimento quando comparado com anos anteriores. Segundo a ABES (Associação Brasileira das Empresas de Software), ocorreu em 2013 um aumento de 15% em relação a 2012 nos investimentos em Tecnologia da Informação (TI) no país (Mercado Brasileiro de Software, 2014). Tal crescimento demanda profissionais qualificados para o trabalho.

A disciplina de Introdução a Programação é uma disciplina básica para os cursos de Computação e é pré-requisito de várias disciplinas. Pois ela é fundamental para os estudantes desses cursos, pois ajuda no desenvolvimento do raciocínio lógico, além de auxiliar no entendimento de elementos da linguagem de programação python.

Em pesquisa realizada pelo Sindicato das Instituições Privadas de Ensino Superior do Estado de São Paulo (Semesp)², Rodrigo Capelato (diretor-executivo do Semesp) afirma que 73% dos alunos de Ciência da Computação desistem da faculdade. Esse percentual em cursos de Sistemas de Informação chega a 67,3%. Dados do Instituto Nacional de Estudos e Pesquisas Anísio Teixeira (Inep 2013) indicam que dois de cada 10 estudantes desistem do curso superior, sendo o maior índice de evasão nas graduações de Processamento da

¹ <http://www.brasscom.org.br/brasscom/Portugues/detInstitucional.php?codArea=3&codCategoria=21>. Acesso em: 21 nov. 2014.

² <http://www.criativamarketing.com.br/index.php/inicio/item/508-mercado-aquecido-faz-alunos-abandonarem-cursos-de-tecnologia> Acesso em 21 nov. 2014

Informação (36%), Marketing (35%) e Ciência da Computação (32%)³. Esses números podem sofrer alterações de acordo com as instituições e estados, mas são dados preocupantes principalmente para os cursos da área de tecnologia.

É possível perceber, através da pesquisa realizada por Duarte (2013), que o estado da Paraíba segue a tendência do cenário nacional. Duarte identificou as taxas de evasão e reprovação do curso de Ciência da Computação da UFPB no campus I entre os anos de 2000 a 2013. O período 2001.1 apresentou taxa média de retenção de 24,2%; essa taxa no período 2012.2 é de mais de 41%. Pode-se observar também que houve uma redução nas taxas de trancamentos, que passou de 17,6% em 2000.1 para 7,3% em 2012.2. Porém houve um forte crescimento na taxa de reprovação por faltas, que passou de 7,0% em 2000.1 para 18,2% em 2012.2. Segundo o autor, um a cada cinco alunos matriculados em uma disciplina do curso é reprovado por falta, o que geralmente ocorre devido ao abandono da disciplina.

Ainda segundo Duarte (2013), dentre as disciplinas que compõem a grade curricular deste curso, Introdução à Programação está entre as 20 disciplinas que se destacam com altos índices de evasão e reprovação. O autor revela que a disciplina possui a taxa de evasão de 29,1%, resultante da taxa de reprovação por faltas de 20,8% e de trancamento de 8,3%.

Buscando identificar esses indicadores na UFPB-Campus IV, foram levantados dados referentes aos índices de reprovação e trancamento das disciplinas introdutórias nos cursos de LCC (Licenciatura em Ciência da Computação). É possível perceber que a disciplina de Introdução à Programação, juntamente com Cálculo I, Matemática Elementar e Introdução ao Computador representam as disciplinas que mais reprovam. Os índices de reprovações e trancamentos dessas disciplinas são altos e se repetem por diversos períodos como mostram as figuras abaixo (Figura 1 e Figura 2).

³ <http://www.gazetadopovo.com.br/vida-universidade/nocampus/conteudo.phtml?id=1248860> Acesso em 23 nov.2014

Figura 1 Índice de reprovação das disciplinas nos períodos de 2011.1 a 2012.1

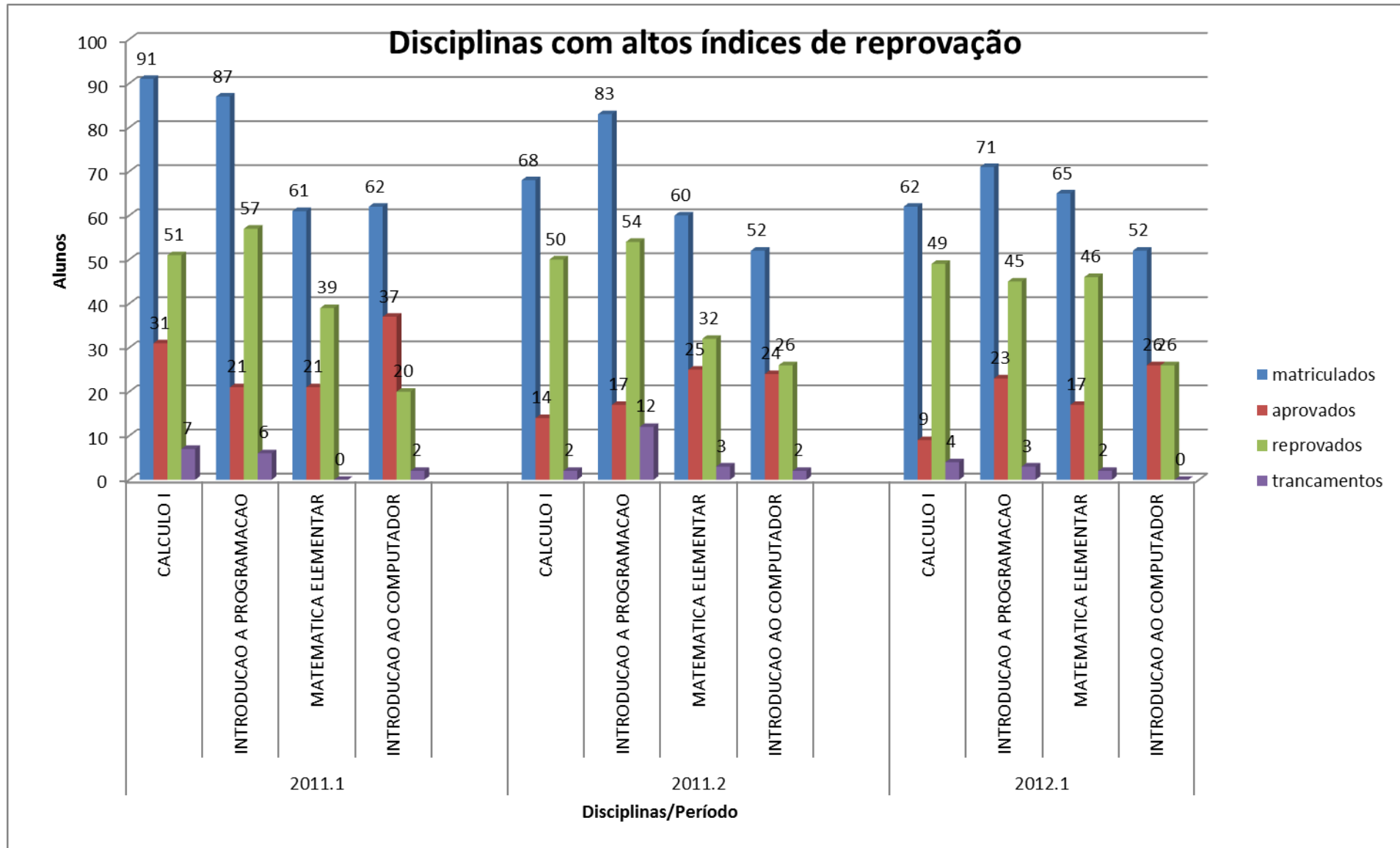
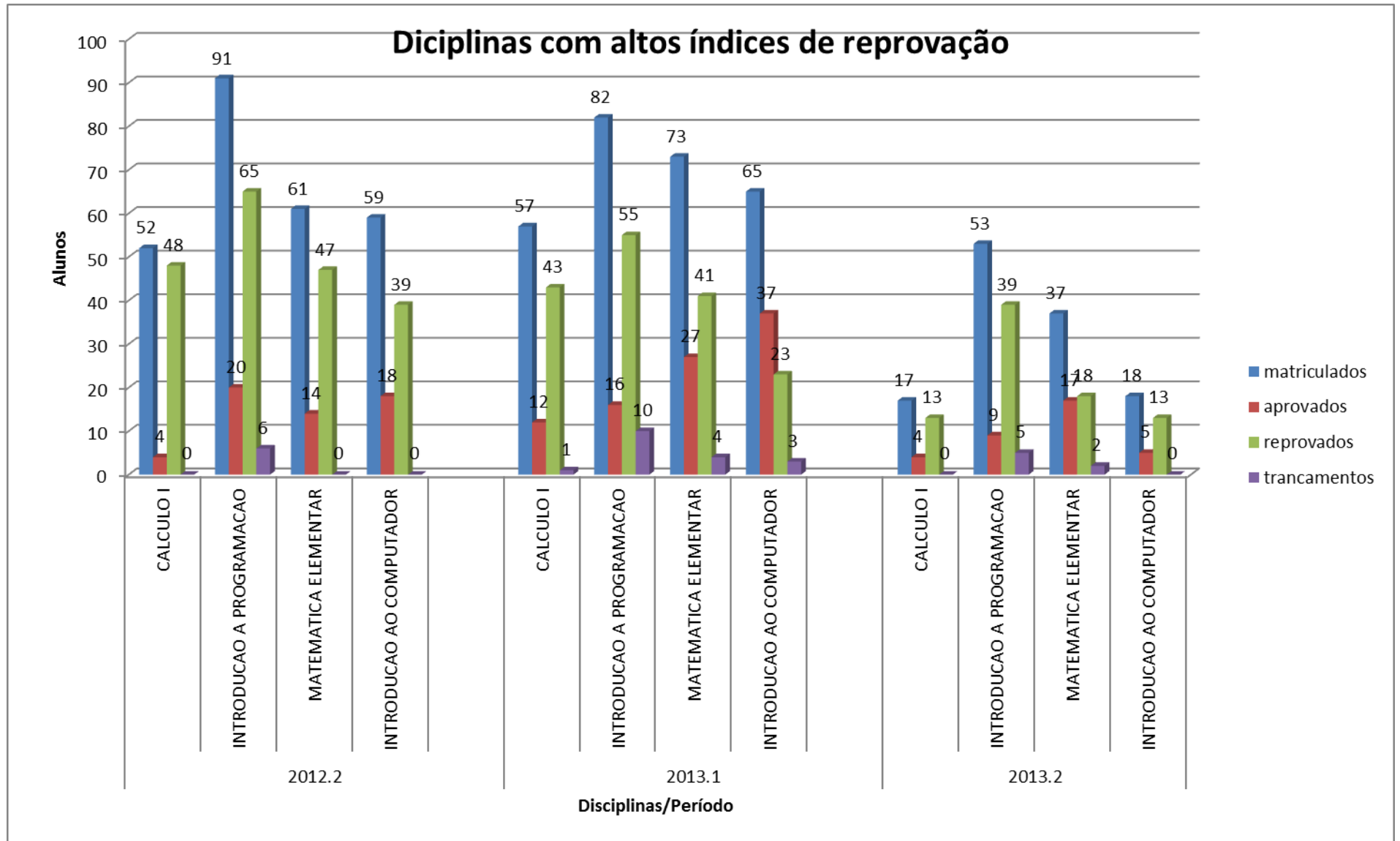


Figura 2 Índice de reprovação das disciplinas nos períodos de 2012.2 a 2013.2



Analisando a Figura 1 percebe-se que do período 2011.1 ao período 2012.1 houve uma pequena diminuição na quantidade de alunos reprovados, mas, também é possível perceber que o número de alunos matriculados nestas disciplinas diminuiu. A disciplina de Introdução ao Computador é a que apresenta menor desigualdade entre o número de alunos aprovados e reprovados. Na disciplina de Introdução à Programação, a quantidade de alunos reprovados chega a ser, em média, 40% maior que o de aprovados. Nos períodos 2012.2 e 2013.2 (Figura 2) a disciplina de Introdução à Programação conseguiu atingir o índice de mais de 70% de alunos reprovados. Se somarmos os alunos que trancaram a disciplina nestes períodos, esses índices chegam perto dos 80%.

A disciplina de Introdução à Programação é considerada por diversos alunos como difícil e desestimulante, pois estes sentem dificuldades em entender e aplicar os conceitos mostrados em sala de aula. Conforme Hinterholz (2009), dentre os problemas relacionados a disciplina estão (i) a dificuldade no desenvolvimento do raciocínio lógico, (ii) a falta de motivação, pois o aluno acredita que a disciplina é um obstáculo difícil de ser superado, (iii) a dificuldade em relacionar a teoria e a prática e, (iv) a dificuldade na assimilação de abstrações e no acompanhamento de algoritmos durante a explicação do professor.

Segundo Branco Neto e Schuvartz (2007, p. 521),

os cursos da área de computação enfrentam um grande problema com as disciplinas de Introdução à Programação. Acadêmicos iniciantes, ao se depararem com a disciplina, sentem-se incapazes de programar, devido ao conjunto de habilidades que a programação exige como capacidade para solucionar problemas, raciocínio lógico, habilidade matemática, capacidade de abstração, entre outras.

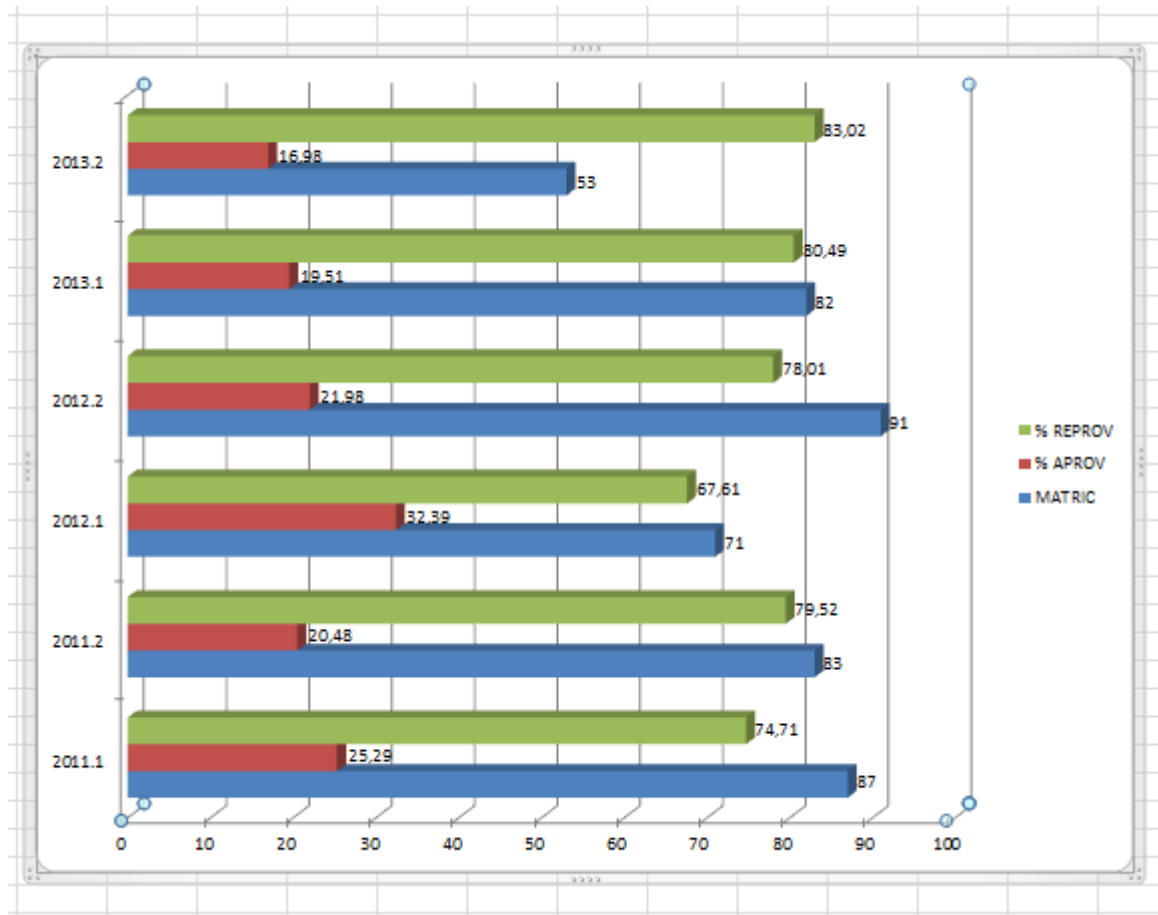
Além das dificuldades relacionadas aos conteúdos da disciplina, alguns autores afirmam que os alunos sentem-se desmotivados quando não conseguem compreender os conteúdos de forma satisfatória. Sajaniemi e Kuittinen (2003, p.7 apud Mota et al, 2008, p.127) afirma que,

a aprendizagem de conceitos iniciais de programação é difícil para muitos estudantes iniciantes. Uma das razões é que os programas envolvem entidades abstratas como: construções formais de loops, ponteiros, arrays, etc., conceitos estes que os estudantes não estão familiarizados.

Devido aos fatores citados anteriormente, é possível perceber que esta disciplina possui problemas relacionados ao processo de ensino-aprendizagem que podem ser o motivo dos altos índices de reprovação. Essas questões também afetam os cursos de tecnologia do Campus IV da UFPB. Em uma análise da disciplina de Introdução à Programação realizada pela Coordenação do curso de Licenciatura em Ciências da Computação é possível perceber

que esta disciplina possui altos índices de reprovação e evasão. No gráfico abaixo são apresentados dados referentes aos períodos de 2011.1 à 2013.2. Apesar de haver pouca variação na quantidade de alunos matriculados, é possível perceber o aumento da taxa de reprovação entre os períodos de 2011.1 à 2013.2.

Figura 3 Índices da disciplina de Introdução à Programação



Conforme Martins (2010) parece existir um consenso entre professores e pesquisadores de que aprender conceitos de programação não é uma atividade simples e requer prática, uma vez que exigem do aluno uma percepção maior e requisitos técnicos. O método de ensino e os recursos didáticos utilizados pelo professor também são determinantes para o desempenho dos alunos na disciplina. Em função disso, a forma tradicional de ensino excessivamente centrada no professor, faz com que falte aos estudantes oportunidades para aplicação prática dos conceitos (Chen et al., 2008). Assim, é necessário propor métodos de ensino de programação que despertem o interesse e motivem os alunos para o aprendizado dos conteúdos da disciplina, buscando modificar o cenário de desmotivação no processo de ensino-aprendizagem.

Fazer com que o aluno compreenda os conteúdos apenas na teoria não é a solução ideal para o problema. Por isso, acredita-se que o uso de ferramentas e objetos de aprendizagem (OA) possa fazer com que os alunos compreendam melhor os conteúdos e ainda se sintam mais motivados a aprender. De acordo com o *IEEE Learning Technology Standards Committee* (LTSC), objetos de aprendizagem podem ser definidos como sendo qualquer recurso, digital ou não, que pode ser utilizado, reutilizado ou referenciado durante o processo de aprendizagem que utilize tecnologia.

Em meio a vários conceitos sobre OAs, Wiley (2004) define objeto de aprendizagem como qualquer fonte digital que poderá ser reutilizada para aprendizagem. Nessa vertente, espera-se que um OA sirva de apoio às práticas pedagógicas, facilite e estimule a capacidade dos seus utilizadores e colabore com o processo de ensino-aprendizagem, tanto em sala de aula como além dela. Segundo Nunes (2004), objetos de aprendizagem, além de servir como meio facilitador de aprendizagem, despertam curiosidade e motivam os alunos a ampliar o conhecimento.

Os benefícios do uso de objetos de aprendizagem vão além da simulação de conteúdos, pois eles permitem ser reutilizados e acessados em diversos ambientes de aprendizagem, e não apenas na sala de aula, além de permitir a integração de diversas ferramentas e plataformas (Fabre, 2003).

Buscando mudar o atual cenário desmotivador de ensino de programação, o presente trabalho propõe a utilização de objetos de aprendizagem como ferramenta de auxílio para o ensino da disciplina de Introdução à Programação, colaborando com os alunos para superar suas dificuldades e fornecendo ao professor mais um recurso para auxiliá-los nas aulas da disciplina. Iremos apresentar desenvolvimento do protótipo inicial do Trivia, Objeto de Aprendizagem com gamificação para servir de apoio a disciplina de introdução a programação com exercício de conteúdos iniciais como variáveis. Através destes objetos pretende-se proporcionar uma forma de estimular os alunos a praticar e exercitar o desenvolvimento do raciocínio lógico, através de perguntas diretas e simples, de um forma simples com dinâmicas de jogos aliada aos conteúdos

1.2 OBJETIVOS

1.2.1 OBJETIVO GERAL

O objetivo deste trabalho é propor um objeto de aprendizagem que possa auxiliar alunos e professores no processo de ensino-aprendizagem de conteúdos específicos que os alunos sentem mais dificuldades na disciplina de Introdução à Programação, como variáveis, operadores matemáticos, operadores lógicos, operadores relacionais, estrutura de seleção (if, if/else) e estrutura de repetição (while e for).

1.2.2 OBJETIVOS ESPECÍFICOS

- Identificar os conteúdos da disciplina de Introdução à Programação que os alunos têm maiores dificuldades.
- Realizar, na literatura, uma busca de objetos de aprendizagem utilizados para ensino-aprendizagem da disciplina de Introdução à Programação.
- Desenvolver o protótipo do objeto de aprendizagem.
- Realizar uma avaliação sobre os aspectos pedagógicos do OA.

1.3 METODOLOGIA

Visando atingir os objetivos propostos neste trabalho, inicialmente realizou-se uma pesquisa bibliográfica visando identificar quais os motivos que levam ao alto índice de desistência e reprovação na disciplina de Introdução à Programação. Em seguida, realizou-se uma busca por objetos de aprendizagem utilizados para ensino e/ou aprendizagem da disciplina de Introdução à Programação e que abordavam os conteúdos citados. Em consequência da análise dos objetos de aprendizagem identificados e do levantamento de dados com os professores, iniciou-se o levantamento de requisitos do objeto de aprendizagem aqui proposto. Em seguida, passou-se para a prototipação, implementação e por último, a avaliação do objeto.

1.4 ESTRUTURA DO TRABALHO

O trabalho divide-se em mais cinco capítulos, organizados da seguinte forma: o Capítulo 2, denominado Fundamentação Teórica, apresenta uma análise da literatura citando as dificuldades relacionadas à disciplina de Introdução à Programação, a utilização dos objetos de aprendizagem como recurso para auxílio ao processo de ensino-aprendizagem e gamificação. O Capítulo 3, denominado Trabalhos Relacionados, expõe objetos de aprendizagem para ensino e/ou aprendizagem de programação para iniciantes. O Capítulo 4 descreve sobre a concepção do objeto de aprendizagem. O Capítulo 5, intitulado Avaliação, descreve uma análise da ferramenta proposta segundo critérios pré-definidos. Por fim, o Capítulo 6 apresenta as considerações e propostas de trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Nesse capítulo, serão discutidas as dificuldades no ensino introdutório de programação (Seção 2.1), a utilização de Objetos de Aprendizagem como ferramenta de auxílio para o ensino da disciplina de Introdução à Programação (Seção 2.2), e por último, conceitos de gamificação (seção 2.3) utilizados para colaborar com os alunos para superar suas dificuldades em relação à disciplina.

2.1 DIFICULDADES NO ENSINO DE PROGRAMAÇÃO

Nos cursos da área de TI é comum nos depararmos com disciplinas que requerem um pouco mais de conhecimento abstrato e capacidade de interpretação. As disciplinas iniciais dos cursos dessa área necessitam de atenção e dedicação, tornado assim os primeiros anos do curso importantes e decisivos para a motivação e permanência dos alunos nos mesmos. A disciplina de Introdução à Programação é um exemplo desse caso.

Observando ementas dessa disciplina em diferentes instituições de ensino, é comum encontrar conteúdos como construção de algoritmos e conceitos básicos de um programa como, variáveis, operadores matemáticos, lógicos e relacionais e, expressões. Além disso são discutidas também as estruturas de controle: atribuição, seleção e repetição. Alguns cursos abordam também subprogramas (funções e procedimentos), introdução a dados estruturados (vetores, matrizes e registros) e tratamento de arquivos.

Diversos trabalhos abordam as preocupações com relação ao ensino e aprendizagem de programação (Jenkins, 2002), (Lahtinen e Ala-Mutka, 2005), (Sloane e Linn, 1988) (Branco Neto e Schuvartz, 2007), entre outros. Esses estudos são motivados pela importância dos conceitos iniciais de programação, visto que disciplinas avançadas dependem fortemente dessa fase inicial do curso. Fazer com que o estudante compreenda e desenvolva habilidades de abstração destes conceitos iniciais é de grande importância, pois tem relação direta com o desempenho do aluno no decorrer do curso.

O ponto de origem dos problemas é complexo, pois envolve comportamentos e percepções de alunos e professores. No tocante à perspectiva do aluno, lista-se a seguir um conjunto de habilidades requeridas que necessitam ser desenvolvidas por eles durante a disciplina:

a) Abstrair conceitos e representá-los através de códigos: Jenkins (2002) afirma que uma das dificuldades dos alunos é o baixo nível de abstração e a falta de competências de resolução de problemas. É necessário que haja a compreensão do problema, pois muitas vezes os alunos passam para a fase de codificação, sem compreenderem completamente o que é necessário para resolução do problema. Um dos motivos listados pelo autor é o fato dos alunos se sentirem ansiosos para começar a codificar uma solução.

b) Expressar comandos em nova linguagem: Lahtinen e Ala-Mutka (2005) afirmam que os alunos compreendem conceitos e estruturas que compõem a linguagem, porém não sabem como utilizá-los corretamente. Eles possuem dificuldades quando realizam atividades, pois não conseguem colocar em prática, na linguagem de programação, os conceitos teóricos aprendidos.

c) Interpretação das questões e dos problemas: Segundo Gomes, Henriques e Mendes (2008), as deficiências relacionadas à expressão em língua materna, interpretação de textos e enunciados são fatores que influenciam na aprendizagem. A leitura do enunciado da questão e a interpretação inicial são os primeiros passos para se começar a elaborar um algoritmo para resolvê-la. Os autores ainda citam que hábitos de estudo e pesquisa e, especialmente, com formação básica deficitária no que tange a conteúdos de Matemática são os principais empecilhos para elaboração dos algoritmos.

d) Conhecimento matemático e raciocínio lógico pouco desenvolvido: Sloane e Linn (1988) consideram como exigências essenciais para aprender programação a capacidade de resolução de problemas e conceitos de matemática envolvidos no processo. Programar requer uma boa base de conhecimentos matemáticos e de resolução de problemas.

e) Manter os estudos e acompanhamento de atividades práticas: Assistir às aulas e estudar um texto do livro não são atividades suficientes para aprender programação. Programar exige um intenso trabalho e horas de prática de desenvolvimento e resolução de questões. Os alunos, muitas vezes, negligenciam práticas fora do horário de sala de aula. Assis (2014) discute que uma das causas de reprovações e desistências em disciplinas de programação é o fato do aluno não realizar estudos além do tempo da aula. Outra dificuldade encontrada por parte do professor está no ritmo de aprendizagem de cada aluno. Cada aluno apresenta um ritmo de aprendizagem diferente, que dificulta a condução de um curso na área, pois, não necessariamente, o andamento da disciplina é acompanhado do ritmo de assimilação de cada aluno.

2.2 OBJETO DE APRENDIZAGEM

Mediante a popularização das Tecnologias da Informação, houve uma crescente aceleração no desenvolvimento do processo de ensino e aprendizagem, contribuindo para um avanço na elaboração de materiais didáticos que fazem uso do som, cor e movimento, proporcionando ao usuário uma maior interatividade. Este tipo de recurso didático pode ser classificado como um objeto de aprendizagem (OA).

Existem diversas definições na literatura sobre o conceito de OA. Conforme Wiley (2000, p.3 apud Macêdo, 2007, p. 330), um OA é qualquer recurso digital que possa ser reutilizado para o suporte ao ensino.

Em outra definição, o termo objeto de aprendizagem é utilizado como granular e reutilizável pedaço de informação independente de mídia (Muzio, 2001). Tarouco, Fabre e Tamusiunas (2003) definem como qualquer recurso suplementar ao processo de aprendizagem, que possa ser reusado para apoiar a aprendizagem.

Spinelli (2007 p. 7) afirma que:

Um objeto virtual de aprendizagem é um recurso digital reutilizável que auxilie na aprendizagem de algum conceito e, ao mesmo tempo, estimule o desenvolvimento de capacidades pessoais, como, por exemplo, imaginação e criatividade. Dessa forma, um objeto virtual de aprendizagem pode ainda compor um percurso didático, envolvendo um conjunto de atividades, focalizando apenas determinado aspecto do conteúdo envolvido, ou formando, com exclusividade, a metodologia adotada para determinado trabalho.

Na literatura, diversos autores relatam a importância do uso de objetos de aprendizagem. Para Fagundes e Taquary (2009), tais objetos podem ser utilizados como recurso de aprendizagem antes, durante ou ao fim dos conteúdos preparados pelo professor. Os autores ressaltam que, quando esses objetos forem utilizados, deve-se questionar como e quando utilizá-los para que eles possam se enquadrar adequadamente no plano de ensino do docente.

Singh (2001) define a estrutura que um objeto de aprendizagem deve possuir. O autor lista que, no mínimo, o OA deve possuir objetivo, conteúdo instrucional ou pedagógico e, prática e *feedback* para que este recurso se diferencie de outras tecnologias aplicadas à educação. A seguir são listados brevemente a definição de cada item:

- Objetivo: sua finalidade é demonstrar ao aluno o que pode ser aprendido a partir do estudo desse objeto de aprendizagem, além dos conceitos necessários para um bom aproveitamento do conteúdo;
- Conteúdo Instrucional ou pedagógico: parte que apresenta todo o conteúdo necessário para que, ao término, o aluno possa atingir os objetivos definidos;
- Prática e *Feedback*: a cada final de utilização, julga-se necessário que o aluno registre a interação com o objeto para a produção do conhecimento; isto é, confirmam-se se as hipóteses ou opções do aluno estão corretas ou são dadas orientações para ele continuar buscando novas respostas.

Com isso, entende-se que os objetos de aprendizagem são recursos capazes de proporcionar, mediante a combinação de diferentes mídias digitais, situações de aprendizagem em que o educador assuma o caráter de mediador e o aluno o caráter de sujeito ativo dentro do processo de ensino aprendizagem.

2.3 GAMIFICAÇÃO

Existem várias técnicas capazes de estimular os alunos no processo de ensino e aprendizagem, e uma delas é Gamificação. Um dos primeiros autores a definir o termo de gamificação foram Deterding e Khaled(2011), que o definiram como sendo o uso de mecanismo e artifícios, que normalmente são utilizados em jogos, em processos que não são jogos. Gamificar um processo significa dar a ele características e dinâmicas que os jogos utilizam, em outras atividades que não são jogos. Não se trata de construir um jogo e sim fazer com que processos de diversos segmentos das atividades, apresentem-se de uma forma jogável, com a possibilidade de que as pessoas que executam tais atividades sintam as mesmas sensações que os jogos provocam nos seus usuários. A proposta da Gamificação é fazer com que esse potencial dos jogos seja aproveitado por qualquer área que se utilize da forma de pensar dos games. Segundo Dubois e Tamburreli (2013), a gamificação surgiu como sendo uma estratégia de marketing para aumentar o compromisso com o cliente, e, devido a sua eficácia, ligeiramente se espalhou para os diversos domínios, como na educação, sistemas bancários, gerenciadores de tarefas, sistemas colaborativos, entre outros.

Para Lands e Bédard (2011) há diversas técnicas e métodos utilizados pelos jogos que podem se caracterizar como um processo de gamificação. Contudo, os mesmos afirmam que se faz necessário que os métodos estejam associados e adaptados ao trabalho proposto,

para que a finalidade do processo não perca sentido. Os autores listam os principais artifícios e técnicas para incluir gamificação:

- Níveis: As atividades são separadas por níveis. Trata-se de uma forma de mostrar ao jogador um reflexo de sua evolução.
- Rankings: Uma forma de aumentar a competitividade entre os jogadores e fazer com que estejam sempre buscando o melhor lugar.
- Barra de Progresso: estímulo visual para que o jogador veja como está se saindo quanto ao seu objetivo final.
- Moeda Virtual: O uso de recompensas é sempre estimulante. O acúmulo de moeda virtual pode ser trocado depois, e de preferência, para serem trocados por objetos virtuais.
- Sistemas de pontos: pontuações diferentes para atividades diferentes fazem o jogador perceber que alguns objetivos podem ser mais fáceis e outros mais importantes.
- Desafios entre os usuários: faz o jogador se comparar a outro e buscar meios para vencê-lo.
- Bônus: premiação extra para o usuário que completa uma série de desafios.
- Combos: pontuações extras para o usuário que consegue acertos.
- Colaboração em comunidade: desafios a serem resolvidos em colaboração com outros jogadores, aumentando a interatividade.
- Contagem regressiva: aumenta a pressão sobre o jogador fazendo com que ele se envolva mais no jogo. Isso pode gerar maior entusiasmo ao jogo e estimulá-lo a concluir o objetivo.
- Descoberta: itens ou informações escondidas despertam a curiosidade do jogador.
- Significado épico: alimenta a imaginação e a criatividade do jogador.
- *Gameplay* Infinito: o jogador sempre recomeça de onde parou.
- Loteria: ações que envolvem sorte , onde o jogador tem a oportunidade de obter pontos extras
- Pontuação por tempo: Mostram ao jogador a importância da agilidade para cumprir tarefas.

2.4 AVALIAÇÃO DE OBJETOS DE APRENDIZAGEM

Esta seção apresenta a metodologia para avaliação de objetos de aprendizagem LORI (*Learning Object Review Instrument*) (2002).

2.4.1 Proposta de avaliação LORI

A metodologia LORI (*Learning Object Review Instrument*) tem como objetivo avaliar a qualidade de um objeto de aprendizagem a partir de uma avaliação heurística. Um aspecto importante desta metodologia é a análise feita para identificar se o objeto de aprendizagem atende aos requisitos que se propõe. Os critérios utilizados nesta metodologia avaliam nove aspectos. Para cada item, a qualidade é avaliada em uma escala que possui cinco níveis, sendo o nível 1 o mais baixo e o nível 5 o mais alto, podendo ser avaliado como NA (não aplicável) quando o item não é relevante para o objeto de aprendizagem ou caso as pessoas que estejam realizando a avaliação não se sintam qualificadas para julgar este critério. A seguir são apresentados os critérios:

- **Qualidade do conteúdo:** a metodologia busca identificar se os conteúdos não têm erros e enfatizam conceitos-chaves e ideias significantes com um nível apropriado de detalhes.
- **Alinhamento dos objetivos de aprendizagem:** o nível das atividades, conteúdos e avaliações de aprendizagem contidos no objeto de aprendizagem devem estar alinhados com os objetivos pretendidos e de acordo com o público-alvo.
- **Feedback e adaptação:** o objeto de aprendizagem possui mensagens instrutivas ou atividades de acordo com as necessidades ou características específicas do aluno.
- **Motivação:** o conteúdo do OA é motivador e relevante para os objetivos e interesses do aluno. O objeto de aprendizagem busca relacionar as atividades com aspectos da vida real, interatividade ou desafios em forma de jogo. O *feedback* compara o desempenho do aluno aos critérios, mostra as consequências naturais do desempenho, e explica como pode o desempenho ser melhorado.
- **Design da Apresentação:** a forma como o objeto de aprendizagem está estruturado (texto legível, gráficos, conteúdos e informações) permite ao usuário uma aprendizagem eficaz.
- **Usabilidade:** facilidade de navegação, previsibilidade da interface do usuário, e qualidade das funções de ajuda da interface.
- **Acessibilidade:** facilidade do acesso independente de plataforma.

- **Reusabilidade:** capacidade de usar em contextos de aprendizagem variados e com alunos com diferentes níveis de conhecimento.
- **Conformidade com os padrões:** o objeto de aprendizagem adere a todos os padrões e especificações internacionais relevantes, por exemplo, padrões dos Metadados de Objetos de Aprendizagem IEEE.

Conforme Gladcheff et al (2011), a avaliação de um OA deve ser feita com metodologias específicas, pois a incorporação desses aplicativos só se justifica quando possibilita um avanço qualitativo nos processos de ensino-aprendizagem. Propomos a utilização dos critérios da metodologia apresentada anteriormente para avaliação do objeto de aprendizagem proposto, LORI (2002).

Tabela 1 Critérios de LORI, adaptado de LORI (2002)

Qualidade do Conteúdo: A metodologia busca identificar se os conteúdos não têm erros e enfatizam conceitos-chaves e ideias significantes com um nível apropriado de detalhes	1	2	3	4	5	NA
Alinhamento do Objetivo da Aprendizagem: O nível das atividades, conteúdos e avaliações de aprendizagem contidos no objeto de aprendizagem devem estar alinhados com os objetivos pretendidos e de acordo com o público-alvo.	1	2	3	4	5	NA
Feedback e Adaptação: O objeto de aprendizagem possui mensagens instrutivas ou atividades de acordo com as necessidades ou características específicas do aluno	1	2	3	4	5	NA
Motivação: o conteúdo do OA é motivador e relevante para os objetivos e interesses do aluno	1	2	3	4	5	NA
Design da Apresentação: A forma como o objeto de aprendizagem está estruturado (texto legível, gráficos, conteúdos e informações) permite ao usuário uma aprendizagem eficaz.	1	2	3	4	5	NA
Usabilidade: Facilidade de navegação, previsibilidade da interface do usuário, e qualidade das funções de ajuda da interface	1	2	3	4	5	NA
Acessibilidade: Facilidade do acesso independente de plataforma	1	2	3	4	5	NA
Reusabilidade: Capacidade de usar em contextos de aprendizagem variados e com alunos com diferentes níveis de conhecimento.	1	2	3	4	5	NA
Aderência a padrões: O objeto de aprendizagem adere a todos os padrões e especificações internacionais relevantes, por exemplo, padrões dos Metadados de Objetos de Aprendizagem IEEE.	1	2	3	4	5	NA

Reatengui, Boff e Finco (2010) afirmam que, um dos primeiros passos na avaliação de um objeto de aprendizagem é identificar sua abordagem epistemológica. Visando cumprir este requisito, utilizaremos as questões propostas por Reatengui, Boff e Finco (2010), para identificar características construtivistas no objeto de aprendizagem. Para que o OA seja considerado construtivista deve possuir resposta positiva para grande parte das seguintes questões:

- Propõe situações-problema que envolvam a formulação de hipóteses, a investigação e/ou a comparação?
- Apresenta recursos (como exercícios, alternativas de navegação) que favorecem a capacidade de elaboração a partir da ação e reflexão?
- Apresenta diferentes caminhos para solucionar um determinado problema?

- Possibilita o registro e a consulta às ações desenvolvidas, permitindo que o aluno reveja e retome seu processo de construção do conhecimento?
- Instiga a procura de outras informações em diferentes fontes de pesquisa?

2.5 CONSIDERAÇÕES FINAIS

Neste capítulo foram identificadas as dificuldades no ensino introdutório de programação como, a dificuldade dos alunos em desenvolverem o próprio raciocínio lógico, quando estão acostumados a decorar os conteúdos das aulas, além da falta de familiarização com a construção de algoritmos e o entendimento da lógica de programação que exigem um nível de abstração para os quais os alunos ainda não estão acostumados. Visando propor uma solução para estes problemas, pode-se utilizar Objetos de Aprendizagem como ferramenta de auxílio para o ensino da disciplina de Introdução à Programação.

Para tornar os Objetos de Aprendizagem mais atrativos para os alunos, é necessário identificar características essenciais que podem ser aliadas ao conceito de gamificação que pode oferecer vários benefícios aos alunos devido a utilização de mecânicas dos jogos em processos que não são jogos, proporcionando assim um fator atrativo como, por exemplo: pontos, recompensas, premiações, entre outros. A utilização de aspectos da gamificação em objetos de aprendizagem tem como objetivo motivar os alunos a aprender e estimular sua produtividade. Por fim, julgou-se importante também conhecer mecanismos de avaliar os OA.

3. TRABALHOS RELACIONADOS

Buscando solucionar parte das dificuldades relacionadas ao ensino–aprendizagem de programação, diversos trabalhos têm sido desenvolvidos na academia. No levantamento inicial sobre objeto de aprendizagem, observamos que poucos são aqueles que partiram de uma elaboração levando-se em consideração aspectos técnicos, pedagógicos e com elementos de gamificação.

Nas seções a seguir serão conhecidas e discutidas algumas destas ferramentas.

3.1. Progame

O progame [Dantas, 2011] é um jogo onde o foco é a edição e manutenção de algoritmos pré-definidos. Inicialmente o jogador deve interagir com os objetos que fazem parte da programação como, por exemplo, pequenos erros de código, acessórios, bits e trechos de código (instruções) (Figura 4). O jogador deve passar por diversos desafios até chegar ao desafio final deparando-se com uma situação-problema onde o aluno tenta solucionar um algoritmo que terá ligação com um elemento do nível, como por exemplo: resolver um algoritmo para abrir a porta de uma sala do tesouro (Figura 5).

Figura 4 Fase do Progame



Figura 5 Desafio do Programe

```
1 END;  
2 READLN(numero);  
3 END.  
4 WRITELN(maior);  
5 READLN(numero);  
6 THEN maior:= numero;  
7 VAR  
8 numero, maior, cont: integer;  
9 IF (numero > maior)  
10 BEGIN  
11  
12     maior:= numero;  
13 PROGRAM Maior100;  
14 BEGIN  
15     FOR cont:=2 TO 100 DO
```

Podemos observar que o programe não trata de um jogo para iniciantes, uma vez que exige do aluno um conhecimento prévio de uma linguagem para solucionar o algoritmo e que ele não oferece ao aluno aspectos motivacionais e *feedback*, entre outros elementos de gamificação que o objeto de aprendizagem proposto neste trabalho possui.

3.2 Entrando pelo Cano

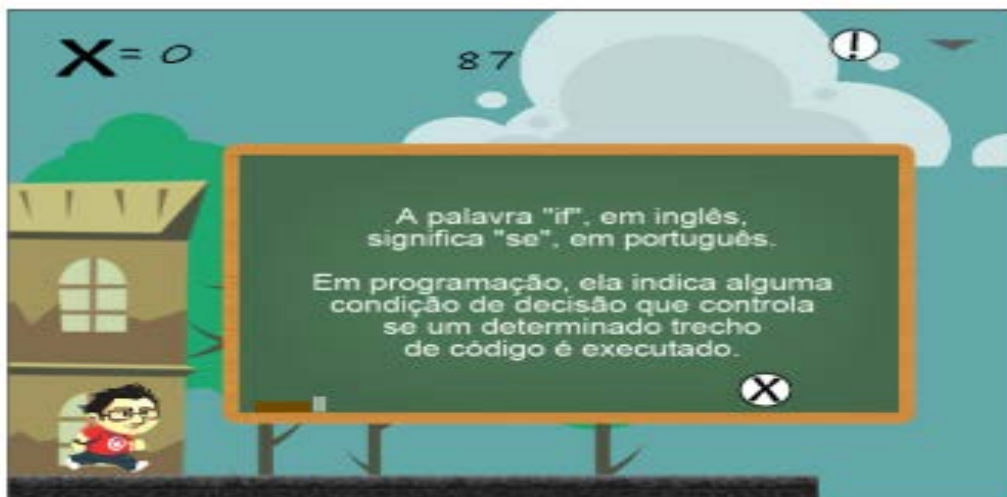
Entrando pelo cano [Scaico et al., 2012] trata-se de um jogo web, onde são propostos aos usuários blocos de desafios com diferentes níveis, onde o usuário tem a opção de escolher qual desafio irá solucionar. O jogo é executado através do Construct 2D 2. O jogador tem acesso ao game através de um navegador web, onde tem acesso aos desafios disponíveis. O usuário é quem controla o ritmo de instruções que deseja receber e a complexidade dos desafios que deseja resolver. Os principais desafios tratam da resolução de algoritmos com elementos da linguagem de programação python, onde o aluno ao final de cada desafio recebe um relatório de desempenho, contendo informações relacionadas à pontuação, classificação do jogador em relação aos outros jogadores e informações de ajuda.



3.3 Klouro

Klouro [Silva et al., 2014] é um objeto de aprendizagem do tipo jogo onde o principal objetivo é praticar conceitos relacionados a operadores e expressões lógicas (Figura 6). O usuário é submetido a uma missão onde ele deve completá-la através de elementos que são propostos ao longo dos desafios, como ultrapassar elementos do jogo, como buracos, piso móvel, cujo objetivo é fazer com que o usuário entenda o algoritmo que é expressado através da missão proposta por cada fase. No decorrer de cada missão, caso o usuário tenha alguma dúvida, pode recorrer a opção de ajuda. O jogo foi desenvolvido usando Construct 2D 2. O jogador tem acesso ao game através de um navegador web.

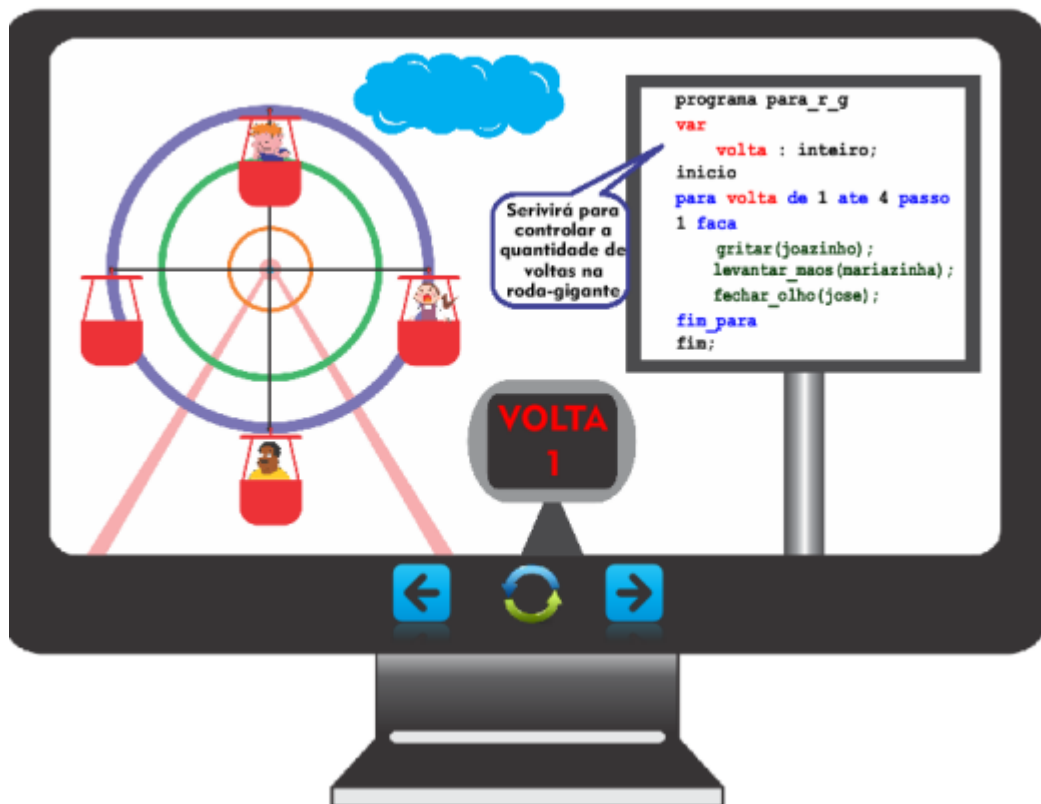
Figura 6 Tela do Objeto de Aprendizagem Klouro



3.4 Monitor

Monitor [Saraiva et al., 2010] é um objeto de aprendizagem do tipo animação (Figura 7). Um quiz cujo objetivo principal é a utilização de uma pseudo-linguagem de programação para apresentação dos conceitos e dos exemplos de Introdução à Programação. O software utilizado para a construção do OA foi o Adobe Flash Player CS6, os assuntos abordados no OA referem-se aos conceitos de Portugol e algoritmo, tipos de variáveis, tipos de operadores e estrutura de repetição. O objeto de aprendizagem apresenta exercícios para prática dos alunos, com perguntas e correção automática.

Figura 7 Tela do Objeto de Aprendizagem Monitor



4.Trivia

Este capítulo apresenta a proposta do objeto de aprendizagem planejado para auxiliar a prática de conceitos introdutórios de programação. Na Seção 4.1 é apresentado o planejamento do Trivia. Na seção 4.2 é descrito o desenvolvimento do Trivia, juntamente com os protótipos de telas.

4.1 Planejamento

O objeto proposto, chamado Trivia, pode ser utilizado como uma ferramenta educacional. Ele atende aos requisitos de Singh (2001), que cita as características mínimas que um OA deve possuir. O objeto de aprendizagem Trivia possui objetivo, conteúdo instrucional/pedagógico e, prática e *feedback*. No Trivia estas características são apresentadas da seguinte maneira:

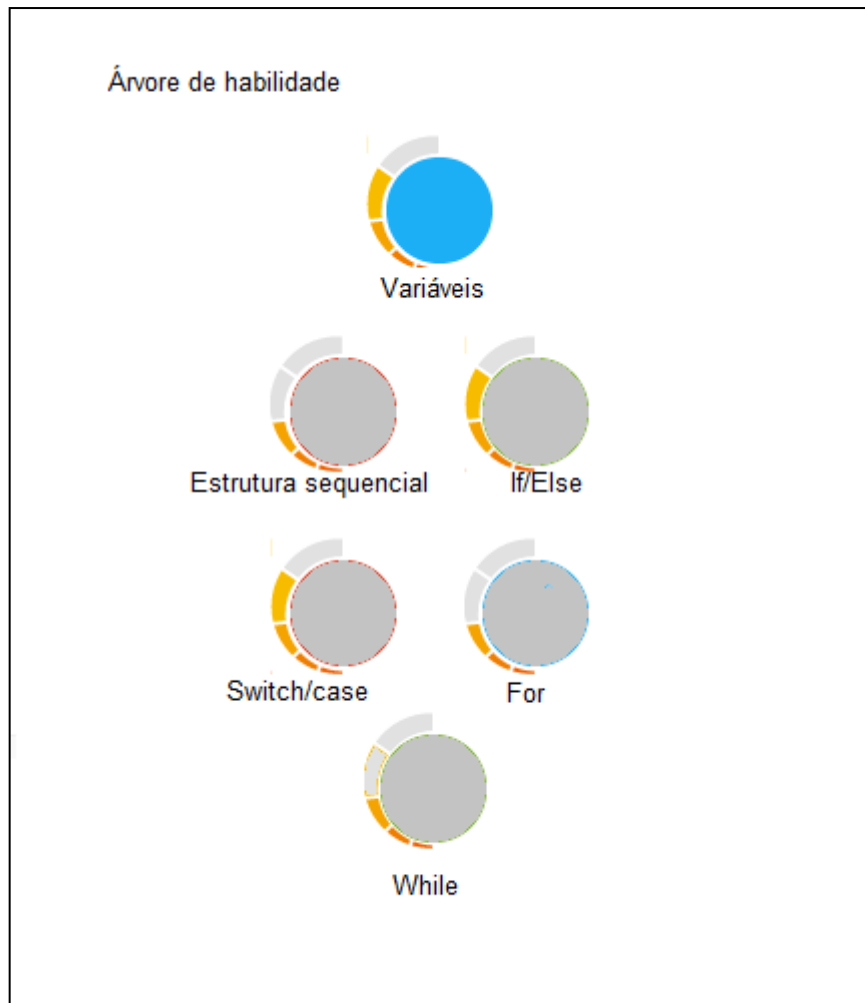
- Objetivo: auxiliar o aluno a praticar conceitos relacionados à programação para iniciantes em disciplinas de Introdução à Programação na linguagem Python.

- Conteúdo Instrucional: os conteúdos abordados são: variáveis, operadores matemáticos, operadores lógicos, operadores relacionais, estrutura de seleção (if, if/else) e estrutura de repetição (while e for). Esses assuntos foram escolhidos por serem considerados essenciais para o conteúdo de Introdução à Programação. Segundo levantamento de dados junto professores que lecionaram ou lecionam a disciplina.

- Prática e *feedback*: A ferramenta oferece lições de apoio ao ensino dos assuntos do conteúdo instrucional. Cada lição consiste em um conjunto de perguntas (estilo quiz) a respeito de um assunto do conteúdo instrucional. Assim, o aluno poderá treinar cada um dos assuntos à medida que eles forem sendo abordados na disciplina. Cada lição apresenta ao aluno mensagens de *feedback* dependendo de sua ação.

O Trivia, além de suas características de OA, também apresenta elementos de gamificação embutidos em seus requisitos e design. Trivia possui uma árvore de habilidades na qual os usuários vão progredindo ao completar as lições. A Figura 8 mostra a árvore de habilidades dos conteúdos instrucionais disponíveis: variáveis, estrutura sequencial, If/Else, Switch/case, for, while.

Figura 8 Árvore de habilidades do Trivia



As habilidades, no início, estarão bloqueadas e serão desbloqueadas à medida que o aluno for respondendo todas as lições propostas por cada fase. Inicialmente, a habilidade desbloqueada será o conteúdo relacionado a variáveis, as demais serão desbloqueadas gradativamente, à medida que o aluno for progredindo nas lições.

Os usuários terão acesso à primeira fase do Trivia que contém 10 questões relacionadas ao conteúdo de variáveis. Os usuários ganham 10 pontos para cada resposta correta, podendo somar até 100 pontos por lição. As lições abordam conceitos sobre determinado assunto e são consideradas completas quando os usuários solucionam todas as atividades associadas a ela. Os usuários começam com quatro estrela-bônus nas primeiras lições e três nas lições posteriores. Cada vez que o usuário comete um erro, como forma de penalizá-lo, ele perde 5 pontos da pontuação acumulada na lição e uma estrela é perdida. Quando o usuário perde todas as estrelas durante a lição, ele deve recomeçá-la. O Trivia

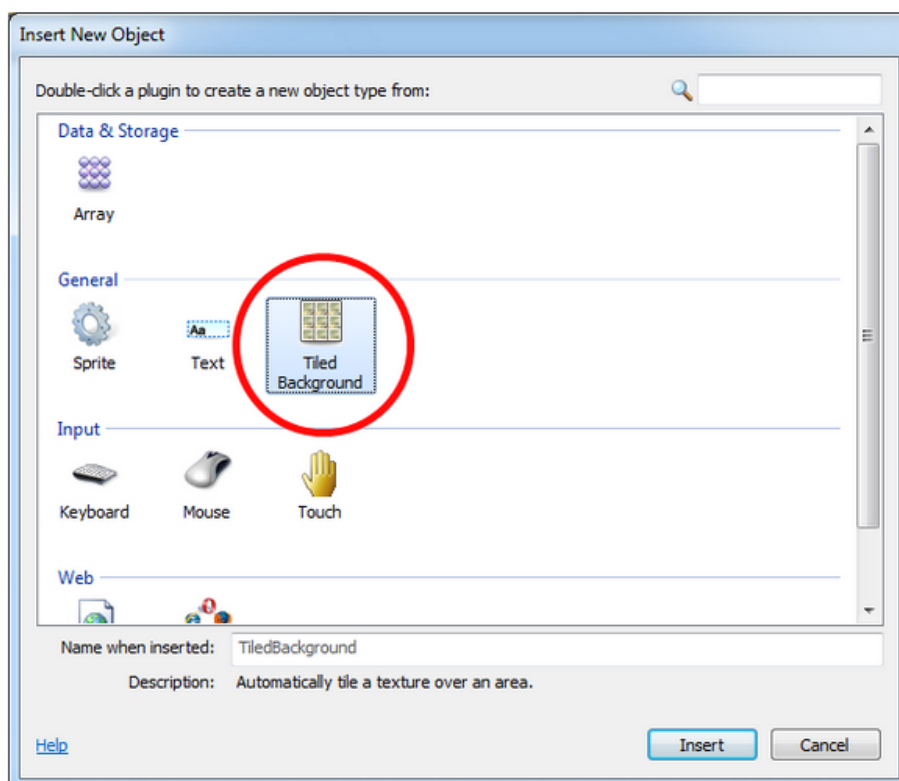
armazena as questões em que os usuários cometeram mais erros e que tipos de erros foram cometidos. Depois de identificadas essas questões, o Trivia fornece o *feedback* de quais conceitos precisam ser reforçados.

Na próxima seção são apresentados alguns dos recursos do Construct 2D que foram utilizados para o desenvolvimento do Trivia.

4.2 Recursos do Construct 2D

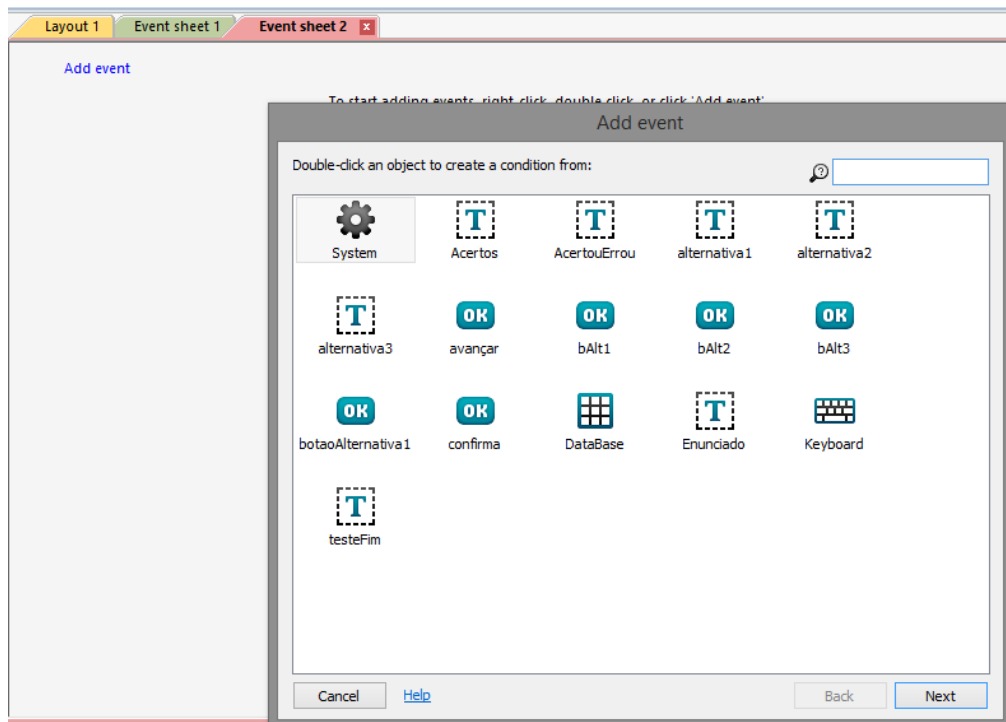
O protótipo inicial de Trivia está sendo desenvolvido usando o Construct 2D (Scirra, 2012). O Construct 2D é uma ferramenta para fazer jogos 2D em diferentes plataformas, permitindo adicionar o jogo desenvolvido a Chrome Store ou ao Facebook. Não é necessário que o usuário tenha conhecimento específico em linguagem de programação, pois a ferramenta possui vários elementos prontos que são disponibilizados para a criação do jogo. A ferramenta permite que cada elemento da tela possa ser associado a um conjunto de eventos. Além disso, o Construct 2D contém vários plug-ins e comportamentos.

O Construct 2D apresenta a opção de criação de telas na própria ferramenta. Para utilizar este recurso é necessário clicar duas vezes em um espaço vazio dentro do layout para inserir um novo objeto. Pode-se, também, clicar com o botão direito e selecionar a opção *Insert new object* (inserir novo objeto). Uma vez que aparecer a caixa do *Insert new object*, deve-se clicar duas vezes no objeto *Tiled Background* para inseri-los. Como mostra a Figura 9.



Para adicionarmos eventos ao projeto que está sendo desenvolvido, clicamos em *Event sheet* (folha de eventos) para mudar para o editor da folha de eventos. O Construct 2D irá executar todos os eventos que estiverem na folha de eventos. Para adicionar um evento clicamos em *Add event* na folha de eventos e escolhemos um dos objetos que foram adicionados no *Layout*. Em seguida, os possíveis eventos a serem adicionados são exibidos. Há diversos eventos para cada Figura 10.

Figura 10 Tela *Add Event* do Construct 2D



De modo geral, todos os recursos do Construct 2D que foram utilizados no desenvolvimento do Trivia foram adicionados através das abas de *layout* e o *event sheet* que são utilizadas para elaborar o visual do projeto e adicionar comportamentos, respectivamente.

A próxima seção apresenta o detalhamento do OA Trivia.

4.2 Descrição do Trivia

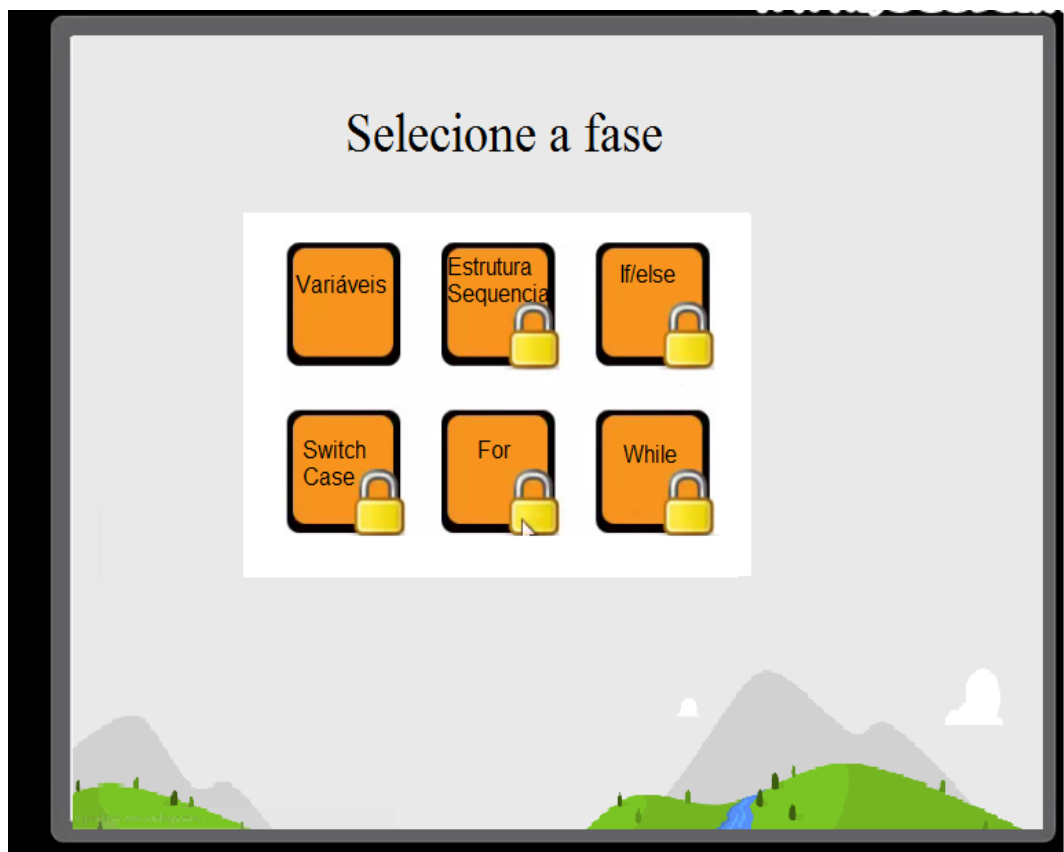
Ao iniciar o Trivia é apresentada ao usuário a tela mostrada na Figura 11. Nela, o usuário tem as seguintes opções: i) botão “Sobre a ferramenta” que apresenta, superficialmente, os assuntos abordados e aspectos de como manusear a ferramenta; ii) botão “Níveis” que indica os possíveis níveis de cada lição proposta; e por último iii) o botão de iniciar a ferramenta.

Figura 11 Tela inicial do Trivia



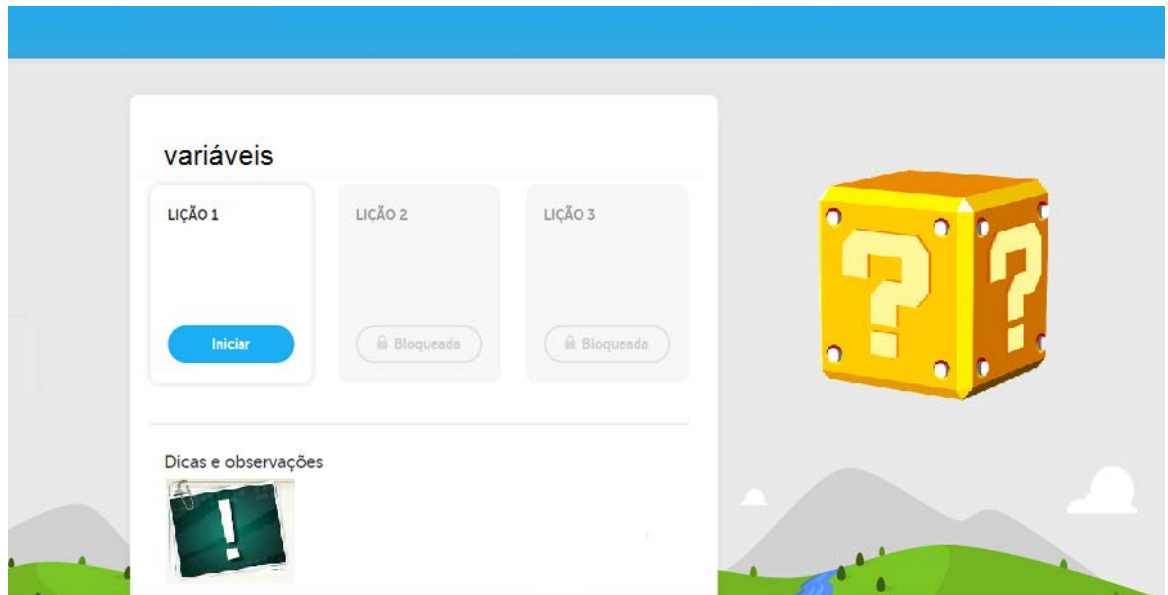
Ao clicar em iniciar, o usuário terá a opção do menu, onde deverá começar a exercitar suas habilidades relacionadas ao conceito de variáveis. Esta lição inicialmente estará desbloqueada, e as demais lições serão desbloqueadas de acordo com o desempenho do usuário, como mostra a Figura 12.

Figura 12 Menu do Trivia



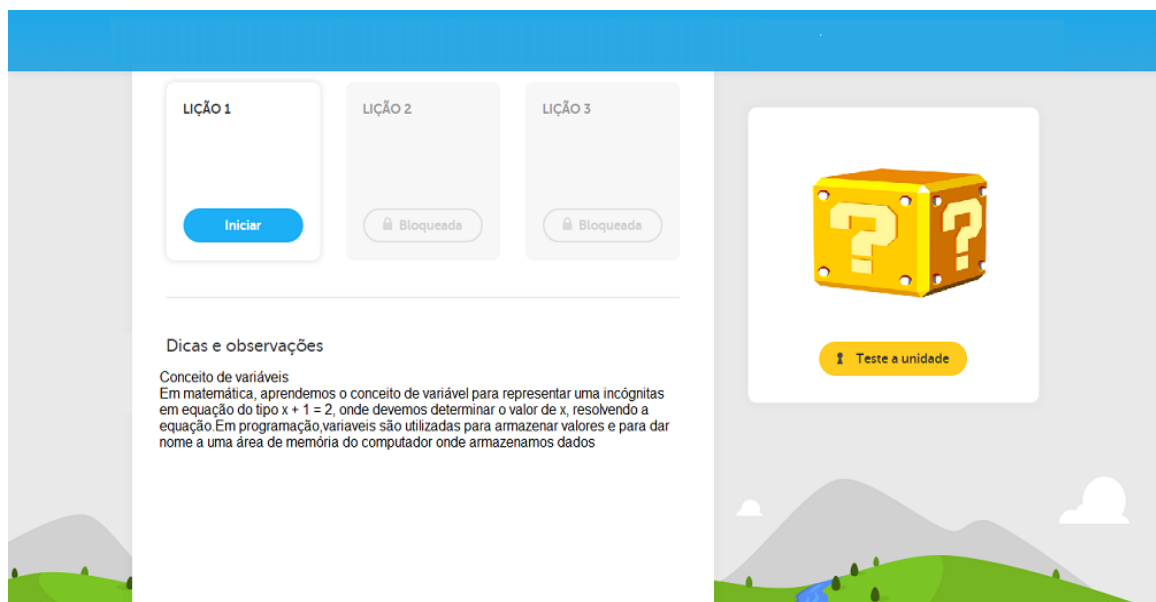
Ao clicar no botão iniciar, o usuário inicia seu teste de conhecimentos e acúmulo de pontos e, em seguida, desbloqueia as outras lições e seus futuros desafios, como mostra a Figura 13.

Figura 13 Tela de lições relacionadas ao conteúdo variáveis



Ao clicar no botão iniciar do lado superior esquerdo, o mesmo estará azul o que simboliza que a fase estará desbloqueada. Para cada lição o usuário recebe uma breve explicação do assunto escolhido como mostra a Figura 14.

Figura 14 Tela de lições com explicação do Trivia



Ao iniciar a lição, serão propostas 10 questões simbolizadas na parte superior pela barra cinza e a medida que o usuário responde cada questão corretamente o usuário irá receber dez pontos e a barra ficará verde, indicando progresso. Caso o usuário responda todas as questões corretas poderá obter pontuação máxima de 100 pontos, O usuário terá a opção de errar 4 vezes no primeiro desafio e 3 nos posteriores. Para isso o usuário receberá 4 estrelas no primeiro desafio e 3 nos demais. Essas estrelas do lado superior simbolizam bônus, bônus estes que deverão ser utilizados, caso o usuário responda a questão incorretamente, É necessário que o usuário responda as 10 questões propostas por cada desafio, para que desbloqueie a próxima fase de questões, Cada vez que o usuário comete um erro, como forma de penalizá-lo, ele perde 5 pontos da pontuação acumulada na lição e uma estrela é perdida. Quando o usuário perde todas as estrelas durante a lição ele deve recomeçá-la.

Na Figura 15 é mostrada uma das possíveis questões da ferramenta da fase inicial em forma de pergunta direta. O usuário deverá selecionar a resposta correspondente à pergunta, e para isto o mesmo poderá clicar selecionando a resposta desejada. Uma vez que a resposta esteja correta o usuário receberá 10 pontos e um *feedback* positivo da questão.

Figura 15 Tela de acerto do Trivia

The screenshot shows a trivia question interface. At the top, there is a blue header bar. Below it, a progress bar consists of 10 segments, with the first 5 being green and the last 5 being grey. To the right of the progress bar are the buttons "Desistir" and "Dicas & observações", followed by four star icons (red, yellow, blue, green). The current score is displayed as "score:50". The question text is "Selecione a sequencia correta para cada tipo de variável". The question content is displayed in a box: "mensagem --> 'O que há? Velhinho'", "n --> 17", and "pi --> 3,14". To the right of the question box are four radio button options: "inteiro, String, float", "String, float, inteiro", "String, inteiro, float", and "float, String, inteiro". The third option, "String, inteiro, float", is selected and has a green checkmark next to it, with the text "Resposta correta!" in green. At the bottom, there are two buttons: "Pular" (grey) and "Verificar" (green).

Figura 16 Tela de erro do Trivia



Neste protótipo de tela (Figura 16) é mostrado o erro do aluno, quando o usuário seleciona a resposta incorreta, ele recebe um *feedback* do erro e uma indicação de qual seria a resposta correta. Cada erro cometido pelo usuário o fará perder 5 pontos e uma estrela-bônus, as quais se encontram no canto superior e simbolizam vidas que o usuário recebe a cada desafio. No primeiro desafio o usuário receberá quatro estrelas-bônus e nas posteriores apenas três. Isso significa que a cada erro o usuário tem uma vida a menos, caso o usuário erre mais de quatro vezes no primeiro desafio ele é obrigado a recomeçá-lo.

A mesma sequência prossegue para as demais lições do Trivia, como também para as demais fases do objeto de aprendizagem. Ao final, o usuário receberá um relatório com seu desempenho, mostrando as fases com acertos e erros .

Figura 17 Relatório do desempenho do usuário no Trivia



4.3. Considerações finais

Neste capítulo foi apresentado o protótipo do Trivia, um objeto de aprendizagem desenvolvido com características de gamificação para auxiliar o ensino da disciplina de Introdução à Programação. Este OA possui 2 lições que tratam de conteúdos relacionados ao conteúdo de variáveis. Cada lição possui 10 desafios, o usuário deve solucionar os problemas para conseguir avançar de nível, ou seja, passar para uma nova lição. Quando o usuário comete um erro ele perde uma das três estrelas que possui, caso perca todas terá que fazer a lição novamente do início.

O próximo capítulo apresenta a avaliação inicial do Trivia, onde foi utilizada duas diferentes metodologias, de LORI (2002) e Reatengui, Boff e Finco (2010).

5. AVALIAÇÃO

Sabendo da importância que o objeto de aprendizagem traz para professor e alunos, realizou-se a avaliação do OA proposto utilizando critérios de duas metodologias, a proposta por Reatengui, Boff e Finco (2010) e a LORI (2002), ambas apresentadas no capítulo de fundamentação teórica. A avaliação foi realizada pela própria autora do trabalho.

Segundo a abordagem LORI, os critérios a serem analisados são organizados em uma tabela (Tabela 2).

Tabela 2 Avaliação do Trivia em relação a LORI (2002)

Qualidade do Conteúdo: A metodologia busca identificar se os conteúdos não têm erros e enfatizam conceitos-chaves e ideias significantes com um nível apropriado de detalhes	1	2	3	4	5	NA
			x			
Alinhamento do Objetivo da Aprendizagem: O nível das atividades, conteúdos e avaliações de aprendizagem contidos no objeto de aprendizagem devem estar alinhados com os objetivos pretendidos e de acordo com o público-alvo.	1	2	3	4	5	NA
			x			
Feedback e Adaptação: O objeto de aprendizagem possui mensagens instrutivas ou atividades de acordo com as necessidades ou características específicas do aluno	1	2	3	4	5	NA
			x			
Motivação: o conteúdo do OA é motivador e relevante para os objetivos e interesses do aluno	1	2	3	4	5	NA
Design da Apresentação: A forma como o objeto de aprendizagem está estruturado (texto legível, gráficos, conteúdos e informações) permite ao usuário uma aprendizagem eficaz.	1	2	3	4	5	NA
			x			
Usabilidade: Facilidade de navegação, previsibilidade da interface do usuário, e qualidade das funções de ajuda da interface	1	2	3	4	5	NA
			x			
Acessibilidade: Facilidade do acesso independente de plataforma	1	2	3	4	5	NA
				x		
Reusabilidade: Capacidade de usar em contextos de aprendizagem variados e com alunos com diferentes níveis de conhecimento.	1	2	3	4	5	NA
		x				
Aderência a padrões: O objeto de aprendizagem adere a todos os padrões e especificações internacionais relevantes, por exemplo, padrões dos Metadados de Objetos de Aprendizagem IEEE.	1	2	3	4	5	NA
						x

No aspecto de qualidade de conteúdos, O Trivia apresenta conteúdos de acordo com a necessidade do usuário em questão, focando em assuntos relevantes para o objetivo do objeto. Com isso o Trivia recebeu nível 3

Para selecionar os conteúdos utilizados no Trivia, realizamos uma entrevista com professores que ministraram a disciplina, por isso, os conteúdos presentes no jogo estão alinhados com os conteúdos da disciplina. Assim o Trivia recebeu pontuação máxima neste item.

No aspecto de usabilidade, o Trivia obteve nível 3, pois não foi realizado nenhuma avaliação com usuários para verificar aspectos de usabilidade.

Com relação ao aspecto de feedback o Trivia obteve nível 3, pois identifica qual o erro e mostra para o aluno o correto, usando feedback direto de cada questão.

No Aspecto de motivação, O Trivia obteve nível 3, pois alinha conteúdos da disciplina com dinâmicas de jogos como pontuação. Não foi realizada nenhuma avaliação com usuários para verificar aspecto motivacional do Trivia

No aspecto de Design, o Trivia obteve nível 4, pois apresenta questões diretas com informações claras.

No aspecto de usabilidade os botões de ajuda estão presentes em cada desafio , o Trivia obteve nível 3, pois ainda não foi avaliado por alunos

No aspecto de Acessibilidade o trivia obteve nível 4, por ter sido feito inicialmente como auxílio do construct que nos da liberdade de rodar em qualquer sistema operacional

No aspecto de reusabilidade o Trivia obteve nível 2, por se tratar conceitos ligados apenas a disciplina de introdução a programação.

No aspecto de aderência de padrões, o Trivia obteve nível 2, pois não se adere a todos os padrões internacionais .

Em relação a Reatengui, Boff e Finco (2010), o Trivia foi avaliado segundo abordagem comportamentalista, da seguinte maneira (Tabela 3).

Tabela 3 Avaliação do Trivia em relação a Reatengui, Boff e Finco (2010)

Questões	Perspectiva Trivia
Apresenta informações em seções breves?	A ferramenta foi projetada de forma simples, de modo que as telas apresentam informações claras e breves
Fornecer recompensa para respostas corretas?	Para cada fase concluída o usuário receberá pontuação, o que corresponde a responder corretamente, o usuário pode passar para a próxima fase.
Só permite seguir para outro nível se obtiver resposta esperada do aprendiz?	Para que o usuário passe para a próxima fase é necessário que ele tenha concluído o desafio. O que implica em responder todas as questões
Obriga o aluno, no caso de erros, a retornar ao ponto anterior	A ferramenta oferece a opção de a qualquer momento o usuário repetir a fase, mas caso ele cometa três erros, é necessário reiniciar a fase.
Propõe situações-problema que envolvam a formulação de hipóteses, a investigação e/ou a comparação?	A ferramenta não oferece situações problemas, por se tratar de um quiz
Apresenta recursos (como exercícios, alternativas de navegação) que favorecem a capacidade de elaboração a partir da ação e reflexão?	Em cada tela haverá um botão de ajuda, onde o usuário terá acesso a dicas da atual pergunta que o usuário está respondendo

Em relação aos aspectos técnicos definidos por Reatengui, Boff e Finco (2010), o Trivia foi avaliado da seguinte maneira (Tabela 4).

Tabela 4 Avaliação do Trivia em relação a Reatengui, Boff e Finco (2010)

Questões	Perspectiva do Trivia
É isento de erros?	O protótipo inicial do Trivia foi testado pelos desenvolvedores, e até então não apresentam erros, podendo ocorrer nas demais fases do Trivia.
O objeto de aprendizagem pode ser utilizado em computadores com configurações diversas, das mais simples até as mais sofisticadas?	O Trivia está sendo desenvolvido no Construct 2, uma ferramenta que permite que o OA possa ser exportado para diferentes plataformas
O objeto pode ser utilizado em computadores com diferentes sistemas operacionais (ex. Linux, Windows, MacOS)?	Por ser um OA que também estará disponível na Web, o Trivia pode ser utilizado em qualquer sistema operacional, desde que o computador possua conexão com a Internet.
As imagens são empregadas para ilustrar conceitos e explicações ao invés de apenas decorar as páginas?	As imagens que foram empregadas em cada fase do trivia estão sendo utilizadas para explicar ou exemplificar algo, o que o torna significativo. Exemplos: botões, textos e outros
O número de imagens apresentadas em cada página é adequado, considerando-se que a presença excessiva de imagens pode gerar sobrecarga cognitiva - terminando por prejudicar os processos de aprendizagem?	As imagens foram definidas de modo que não sobrecarregassem as telas do Trivia, projetadas de forma simples e direta,
Há contraste suficiente entre fontes e fundo de tela, facilitando a leitura dos textos?	Os textos foram projetados de forma que o usuário possa identificar facilmente os elementos textuais dos demais elementos da tela.
As fontes utilizadas apresentam tamanho adequado, ou permitem que sejam aumentadas/diminuídas de acordo com a necessidade de cada usuário?	Os textos contidos no Trivia procuram de forma clara e breve, levar em consideração a cor, tamanho e posicionamento da fonte, para que o usuário não tenha dificuldades em ler os conteúdos.
Os ícones que dão acesso a outras páginas e funções do objeto são facilmente compreensíveis?	Os ícones utilizados no Trivia foram projetados para que sejam intuitivos. Como por exemplo, o botão de dicas e observações possui uma sinal de exclamação.
Os recursos interativos empregados vão além da seleção de links e botões para avançar ou recuar na apresentação dos conteúdos?	Não se aplica
O objeto de aprendizagem emprega recursos gráficos que melhoram o aspecto estético da interface, tornando mais agradável sua utilização?	A interface gráfica do Trivia está sendo projetada de forma que seja atraente ao público-alvo.

6. CONCLUSÃO

O objetivo deste trabalho foi apresentar a proposta de um objeto de aprendizagem que possa auxiliar alunos e professores no processo de ensino-aprendizagem de conteúdos específicos da disciplina de Introdução à Programação. Os conteúdos instrucionais foram selecionados de acordo com os ministrados na disciplina.

O principal objetivo deste trabalho foi apresentar a proposta de um objeto de aprendizagem que possa auxiliar alunos e professores no processo de ensino-aprendizagem de conteúdos específicos da disciplina de Introdução à Programação. Os conteúdos instrucionais foram selecionados de acordo com os assuntos ministrados na disciplina. Uma vez que foram identificados os conteúdos que os alunos demonstravam ter maior dificuldade começamos a trabalhar com os conteúdos introdutórios como conceito de variáveis e estrutura sequencial. Após realizar a revisão literária, se pôde perceber que o atual trabalho se difere dos demais, pois contém elementos de gamificação embutidos em sua mecânica, com perguntas com alternativas. Isso faz com que seja um ponto positivo, pois é uma forma de motivar cada um a aprender com uma experiência nova.

O protótipo desenvolvido até o momento apresenta-se de forma diferenciada pois relaciona dinâmicas de jogos e conteúdos, buscando atingir seu objetivo maior que é exercitar de forma motivadora conteúdos iniciais de programação. Outro aspecto positivo do OA em questão que ele possui uma estrutura com perguntas que possui alternativas . A ferramenta também pode ser usada como arcabouço para conteúdos diferentes do que ela foi implementado nesse trabalho. Podendo ser instanciada para outros conteúdos da linha de programação que desejem se beneficiar da estrutura do OA com elementos de gamificação em forma de quiz.

Sugestões de Trabalhos Futuros

Como trabalhos futuros, pretende-se finalizar a versão com as demais fases do Trivia, fases que tratem de assuntos como estrutura sequencial, *if* e *else*, *switch case* entre outros assuntos. Nessa mesma versão, será implementada a autenticação com *login* e senha para que os usuários tenham acesso a uma conta. Serão explorados elementos de gamificação que não foram contemplados, como construir *ranking* para que haja competição entre os usuários, bonificação de medalhas para que haja mais incentivo, melhorar o feedback, com

aspectos como: indicar links diretos para referências de qualidade sobre o conteúdo que precisa ser melhor reforçado pelo aluno.

Pretende-se projetar no OA uma parte para o professor, para que eles possam acompanhar o progresso dos alunos e ver quem e quais conceitos precisam ser reforçados. E por fim se pretende realizar uma avaliação com alunos e professores para saber a perspectiva do público que irá de fato utilizar a ferramenta.

Referências Bibliográficas

Almeida, E. S., Costa, E. B., Braga, J. D. H., Silva, K. S., Paes, R. B. E Almeida, A. A. M. (2002). AMBAP: Um Ambiente de Apoio ao Aprendizado de Programação. In X Workshop sobre Educação em Computação, Florianópolis. Anais do WEI 2002/ SBC 2002.

Araújo, F.V; Falkembach, G.A.M Resultados da comparação na aprendizagem de Algoritmos, através de um Ambiente Adaptado ao Aprendizado, entre um Curso Técnico e um curso de Graduação. In: I Congresso Sul Catarinense de Computação. Criciúma/SC. Brasil (2004).

Aureliano, V. C. O., & Tedesco, P. C. D. A. R.. Ensino-aprendizagem de Programação para Iniciantes: uma Revisão Sistemática da Literatura focada no SBIE e WIE. In Anais do Simpósio Brasileiro de Informática na Educação ,Rio de Janeiro (2012).

Branco, W. C. N; Schuvartzm, A. A.. Ferramenta Computacional de Apoio ao Processo de Ensino-Aprendizagem dos Fundamentos de Programação de Computadores. (2007,p.520-528).

Byrne, P.; Lyons, G. The effect of student attributes on success in programming. In ACM SIGCSE Bulletin (2001), vol. 33, n. 3, p. 49-52.

Dubois D. J and Tamburrelli G.. 2013. Understanding Gamification mechanisms for software development. In Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering (ESEC/FSE 2013).

Scaico P.; Lopes, D.; Melo, L.; Azevedo, M.; Neto, M.; Oliveira, A.; Junior, J.; Labanca, M.; Scaico, A. Um jogo para o ensino de programação em Python baseado na taxonomia de Bloom. In Anais do XXXII Congresso da Sociedade Brasileira de Computação. Curitiba-PR, 2012.

Duarte, A. N. As disciplinas que mais retém alunos em um curso de computação. Disponível em: <<http://alexandre.ci.ufpb.br/retencao-cc-2/>>. Acesso em: 02 nov. 2014.

INEP. Disponível em: <<http://www.inep.gov.br/>>". Acesso em: 21 nov. 2014.

Disponível em: <<http://blog.dito.com.br/2013/11/o-que-e-gamification-como-a-logica-de-jogos-pode-alavancar-o-engajamento-dos-seus-usuarios/>>. Acesso em 30 de out. 2014.

Jenkins, T. On the difficulty of learning to program. In Proceedings of 3rd Annual LTSN_ICS Conference (Loughborough University, United Kingdom, August 27-29, 2002). The Higher Education Academy, p.53-58, 2002.

Jesus, A. N.; Lopes, D. L.; Perin F. R.; Cantão, J. M.; Pimentel, E. P. Objetos de Aprendizagem no ensino de Lógica de Programação. Informática Aplicada, v. 3, n. 2, p. 37-42, 2007.

Júnior, J. C. R. P. e Rapkiewicz, C. E.. “O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura”. In III Workshop de Educação em Computação e Informática do estado de Minas Gerais (WEIMIG’ 2004). Belo Horizonte, MG, Brasil, 2004.

Lahtinen, E.; Ala-Mutka, K.; Järvinen, H.-M. (2005) A Study of the Difficulties of Novice Programmers. Proceedings do 10th ITiCSE, Monte de Caparica, Portugal, p.14-18

Mercado Brasileiro de Software: panorama e tendências, 2014 = Brazilian Software Market: scenario and trends, 2014 [versão para o inglês: Anselmo Gentile] - 1ª. ed. - São Paulo: ABES -Associação Brasileira das Empresas de Software, 2014

Nesbit, J.; Belfer, K.; Leacock, T. Learning Object Review Instrument (LORI). Disponível em <<http://www.transplantedgoose.net/gradstudies/educ892/LORI1.5.pdf>> Acesso em: 07 de novembro de 2014.

Neves, D. E.; Santos, L. G. N. O.; Santana, R. C.; Ishitani, L. Avaliação de jogos sérios casuais usando o método GameFlow. Revista Brasileira de Computação Aplicada v. 6, nº 1. P. 45-59. 2014. <http://www.upf.br/seer/index.php/rbca/article/view/3244/2552>

Rapkiewicz, C. et al. Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais. CINTED-UFRGS: Novas Tecnologias na Educação. 2006. Homepage: <http://www.cinted.ufrgs.br/renote/dez2006/artigosrenote/25157.pdf>

Rodrigues, M. Experiências positivas para o ensino de algoritmos. IV Escola Regional de Computação Bahia-Sergipe. Feira de Santana, 2004. Homepage: <http://www.uefs.br/erbase2004/documentos/weibase/Weibase2004Artigo001.pdf>

Saraiva, I. B.; Netto, C. M. Monitor: um conjunto de objetos de aprendizagem para apoio ao ensino de programação de computadores. In: Congresso da Sociedade Brasileira De Computação, 30., 2010, Belo Horizonte. Anais do XVIII Workshop sobre Educação em Computação. Belo Horizonte: SBC, 2010.

Silva, M.A.A; Dantas, A. KLouro: Um jogo educacional para motivar alunos iniciantes em programação. Simpósio Brasileiro de Informática Na Educação. Anais do XXV Simpósio Brasileiro De Informática Na Educação, 2014.

Sloane, K. D. and Linn, M. C. (1988). Instructional Conditions in Pascal Programming Classes. In R. E. Mayer (ed.), Teaching and Learning Computer Programming, p.207-235. Hillsdale, NJ: Lawrence Erlbaum Associates.

Walter Spinelli, Disponível em www.lapef.fe.usp.br/rived/textoscomplementares/textoImodulo5.pdf. Acesso em 18 de out. 2014