

Algo+naEscola, Ambiente para Apoio ao Ensino de Programação

Natalia do V. Gouvea¹, Maria Clicia S. de Castro¹, Vera M. B. Werneck¹

¹ Programa de Pós-Graduação em Ciências Computacionais

Instituto de Matemática e Estatística

Universidade do Estado do Rio de Janeiro (UERJ)

Rio de Janeiro – RJ – Brasil

professoravalle@gmail.com, {clicia, vera}@ime.uerj.br

Abstract. *This paper defines the Algo+naEscola environment that provides the teacher, a support for the use of the computational logic through games in the classroom. The requirements of this environment were proposed through an experiment in mathematics teaching using algorithms and games. The idea was to establish the teaching of angles through the use of algorithms and computer programming. This experiment was developed during Mathematics classes with Elementary II students, and was a base for the construction of the Algo+naEscola environment.*

Resumo. *Este artigo define o ambiente Algo+naEscola que fornece ao professor, um apoio ao uso da lógica computacional através de jogos na sala de aula. Os requisitos desse ambiente foram propostos através de um experimento do ensino de Matemática utilizando algoritmos e jogos. A ideia do trabalho de campo foi sedimentar, o ensino de ângulos, algoritmos e programação de computadores. Esse experimento foi desenvolvido durante aulas de Matemática com alunos do ensino Fundamental II, sendo a base para construção do ambiente Algo+naEscola.*

1. Introdução

A utilização de jogos computacionais é considerada um forte atrativo para os alunos dos mais diversos níveis de escolaridade. Os educadores podem utilizar esses recursos para desenvolver a capacidade de executar tarefas reais, expandir conceitos e melhorar o desempenho dos alunos (Pereira, 2014; Connolly, 2012).

Os ambientes computacionais dinamizam os conteúdos e auxiliam os métodos pedagógicos relacionados ao processo de ensino e aprendizagem. Neste sentido, pensando no método pedagógico, a utilização das tecnologias com a Matemática pode gerar uma satisfação proporcionada pelo ato de jogar. Esta conciliação e satisfação podem ser alcançadas, uma vez que os jogos eletrônicos estão cada vez mais populares e de fácil acesso.

Os jogos onde se desenvolvem situações de problemas reais necessitam do aluno diversos tipos de competências, como autonomia, iniciativa e raciocínio, que são fundamentais para complementar o processo de ensino aprendizagem (Bennett *et al.*,

2008). Em particular, jogos voltados para o ensino de programação são desejáveis para que haja uma evolução considerável na construção de habilidades de raciocínio lógico e desenvolvimento de algoritmos.

A Matemática é o exemplo de uma disciplina em que o confronto entre motivar e aprender fica bem visível. Não é difícil encontrar classes em que o ensino da matemática e da lógica é apenas teórico, sem propósitos ou sem proximidade com a realidade do aluno e do mundo (Medeiros *et al.*, 2013). Baseado nestes fatos, uma questão muito pertinente é: como o aluno relaciona o estímulo por aprender com o propósito de aprender?

Tentin (2013) destaca que os professores das séries finais (Ensino Fundamental II e Ensino Médio) ainda necessitam de recursos para incluir a prática de jogos. Os professores, segundo Miskulin (2003), não possuem preparo adequado para definir como e quais ambientes computacionais utilizar.

Diante deste cenário, e após abordar “ângulos” numa aula de Matemática, para alunos do 7º ano, de uma escola particular da zona sul do Rio de Janeiro, percebeu-se que era preciso tornar o ensino deste conteúdo mais próximo da realidade deles. Desta forma, propôs-se a utilização de jogos computacionais de ensino em programação como ferramenta para auxiliar na sequência didática. Porém, a aplicação de jogos em sala de aula não é uma tarefa trivial. Ela acarreta num planejamento minucioso e na adaptação aos conteúdos ensinados.

Neste sentido, pensando no método pedagógico, propomos o ambiente Algo+naEscola. Ele surgiu com objetivo de conter espaços para trocas de experiências, com a possível ampliação dos conhecimentos sobre temas relacionados ao ensino de programação e à educação de forma geral, proporcionando diretrizes para um planejamento de aula eficaz, e desenvolvendo experimentos com jogos educacionais.

Este artigo está organizado da seguinte forma. A Seção 2 apresenta a revisão da literatura sobre jogos computacionais para ensino fundamental. A Seção 3 descreve o processo de experimentação e a aplicação de jogos para ensinar Matemática através da programação, utilizando os jogos. A Seção 4 apresenta o ambiente Algo+naEscola. Finalmente, a Seção 5 discute as conclusões e os trabalhos futuros.

2. Revisão da Literatura

Inicialmente, buscou-se na literatura meios a fim de selecionar os jogos computacionais mais adequados para um contexto escolar. Neste caso, um mapeamento da área foi desenvolvido com base no trabalho de Kitchenham (2007), definindo três questões de pesquisa: i) De que maneira os jogos computacionais devem ser utilizados nas escolas? ii) Qual público alvo está sendo focado nos trabalhos de pesquisa de jogos para ensino e programação e matemática nas escolas? iii) Quais os jogos que estão sendo empregados para ensino de algoritmo, programação e Matemática nas escolas?

Por meio desta revisão, foram selecionados 24 artigos que responderam as três questões de pesquisa. Com base na primeira e segunda questão, os trabalhos destacam as oficinas que despertam a curiosidade quanto à área tecnológica, e posteriormente, conceituam o que é algoritmo coletivamente. A proposta de atividade, para este primeiro momento, deve concentrar-se em ferramentas onde o público alvo, os alunos variando de 10 a 17 anos, esteja inserido e familiarizado.

A última questão de pesquisa busca encontrar ferramentas para serem utilizadas. Foram selecionados oito ambientes, que auxiliam o ensino de algoritmos e Matemática: Scratch, GameMaker, Lego Minstorms, LightBot, CodeMonkey, Kodu, Takkou e Tux. Com exceção do ambiente Tux, todos os jogos possibilitam a interdisciplinaridade da Matemática com a programação. Os conteúdos envolvidos nos ambientes abordam os conceitos de ângulos, que estão descritos, a seguir: i) compreender que o conceito de ângulo está associado às ideias de giro, canto (região), orientação e inclinação; ii) identificar os elementos de um ângulo; iii) perceber a existência de ângulos em diversos ambientes; iv) reconhecer o grau como unidade de medida de ângulo; v) identificar os ângulos reto, agudo, obtuso; e vi) medir e construir ângulos com o transferidor.

3. Processo de Experimentação

Em seguida a realização do mapeamento da área, neste contexto, buscou-se um método de experimento que fosse flexível. O método utilizado foi proposto por Wohlin (2012), e se baseia nas seguintes etapas: Escopo, Planejamento, Realização e por fim, Análise, Interpretação e Apresentação dos Resultados.

Na etapa de escopo foi elaborada a hipótese e o objetivo do experimento. A hipótese é sintetizada na possibilidade de tornar interdisciplinar a aula de Matemática com o ensino de programação. O objetivo é propor um modelo de planejamento, por intermédio de dinâmicas de grupo, testes individuais e coletivos. Além de questionários que potencializem o aprendizado do conteúdo de Matemática, desenvolvendo a relação entre o abstrato e o prático, estimulando as habilidades computacionais dos alunos e familiarizando-os quanto à utilização das tecnologias.

3.1. Planejamento

A escola selecionada é privada, localizada no bairro de Botafogo, no município do Rio de Janeiro. Ela contém uma infraestrutura adequada, com computadores, com acesso a Internet, para uma turma de até 30 alunos. O experimento foi realizado no segundo semestre de 2015, envolvendo alunos do 7º ano do ensino Fundamental II. Os encontros aconteceram durante as aulas de Geometria ocorridas nas sextas-feiras, dentro do horário escolar pertinente a turma selecionada. A faixa etária dos alunos variou de 12 aos 14 anos, sendo a turma composta, inicialmente, por 25 alunos.

O primeiro encontro foi organizado com uma breve apresentação do que seria realizado nas próximas semanas e definição das características dos participantes através um questionário avaliativo. Após esse encontro, caracterizado como sondagem, foi feita uma análise do questionário avaliativo. De maneira geral, observou-se que os alunos não tiveram nenhum conhecimento prévio quanto ao ensino de algoritmo, desconheciam qualquer prática relacionada à área tecnológica, mas se interessavam pelo uso de jogos. Os jogos, para os alunos, são passatempos frequentes, mas nunca utilizados em um contexto escolar. A revisão sistemática de jogos foi a base para a escolha dos jogos adequados a este grupo.

O conteúdo pertinente à turma concentra-se no ensino de ângulos. A turma estabelecida para este processo de experimentação está inserida no 7º ano do ensino Fundamental II e busca habilidade de orientação sobre outro objeto. Os jogos selecionados: Frozen e Labirinto (Partovi, 2013) possuem características similares e desenvolvem as mesmas habilidades que os jogos dos ambientes Lightbot (Yaroslavski,

2008) e CodeMonkey (Pinchover *et al.*, 2014), coletados no mapeamento da área. Entretanto, os jogos Frozen e Labirinto possuem a ferramenta de gerenciamento de alunos disponível *online* e gratuito no *site* code.org (Partovi, 2013). Este ambiente foi selecionado por fornecer ao professor um cadastro simples e gratuito para acompanhamento *online* dos alunos. Além de possibilitar a criação de turmas com quantidade ilimitadas de alunos.

Os jogos selecionados possuem graus de dificuldade diferentes e gradativos. O Frozen explora movimentos similares com os do dia a dia. Ele utiliza comandos do tipo “Vire à direita por 90 graus” e abrange não somente a linguagem de programação, mas, também, o ensino específico de ângulos, incluindo ângulos que não são tão óbvios visualmente como 5°, 30°, 36°, 60° e 100° entre outros. O jogo Labirinto utiliza uma maneira inovadora de desenvolver as funções através de blocos que são visuais. Neste jogo o aluno desenvolve conceitos de sequência, como por exemplo, o “Se”, “Se não”, “Repetir vezes” e “Repetir até que”. A seguir, estão descritos mais detalhadamente o ambiente Lightbot e o jogo Frozen.

O ambiente Lightbot é dividido em três níveis: *Basics*, *Procedures* e *Loops*. Ele possibilita uma evolução gradativa do aluno à medida que as fases são concluídas. Cada um destes níveis estão descritos a seguir:

- Nível 1: *Basics*, o aluno tem que concluir 8 fases que necessitam de comandos executados um após o outro. A Figura 1 mostra a janela relativa a fase 4 do nível 1 do ambiente Lightbot.

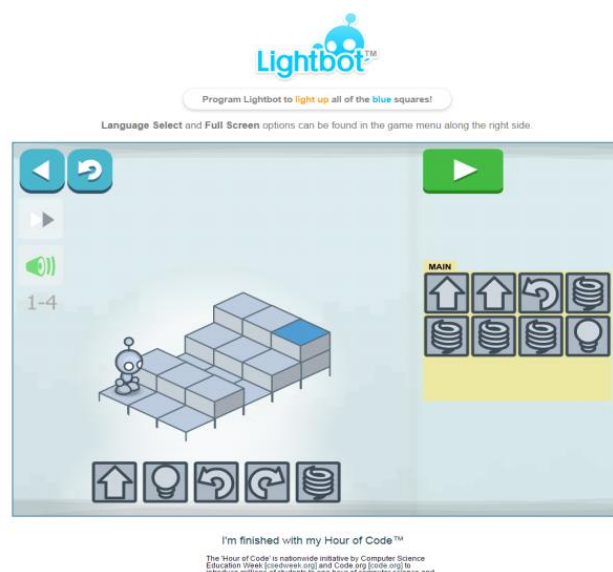


Figura 1. Fase 4 nível 1 do ambiente Lightbot.

- Nível 2: *Procedures*, neste nível o aluno deve concluir 6 fases. A partir do deste nível, como pode ser visto na Figura 2, que mostra a fase 1 do nível 2, o aluno deve utilizar blocos de códigos para a realização das fases seguintes.



Figura 2. Fase 1 Nível 2 do ambiente Lightbot.

- Nível 3: *Loops*, o aluno conclui 6 fases que utilizam, também, blocos de comandos, como pode ser visto na Figura 3, que ilustra a fase 1 deste nível. A diferença é que a partir do nível 3 são usados, especificamente, os padrões que se repetem ou *loops*.



Figura 3. Fase 1 Nível 3 do ambiente Lightbot.

As Figura 4 e Figura 5 mostram o ambiente Lightbot com alguns exemplos de códigos com comandos executados um após o outro, procedimentos e repetições.

O jogo Frozen foi desenvolvido para que os alunos programem duas personagens, com o objetivo de guiá-las até que se explore um caminho determinado pelo próprio ambiente.

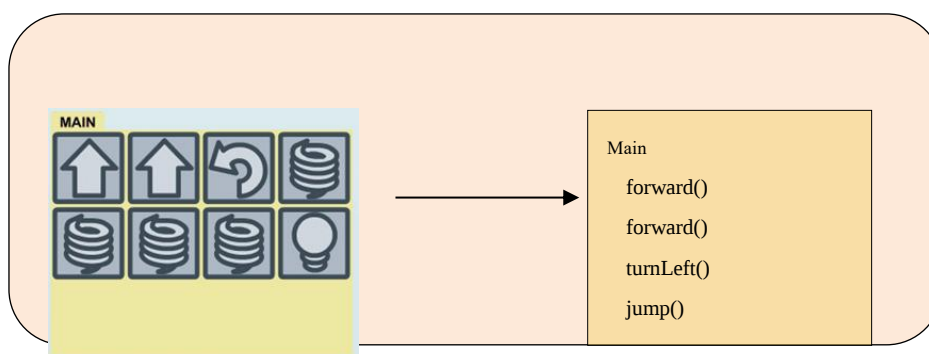


Figura 4. Comandos relativos à fase 4 nível 1 do ambiente Lightbot.

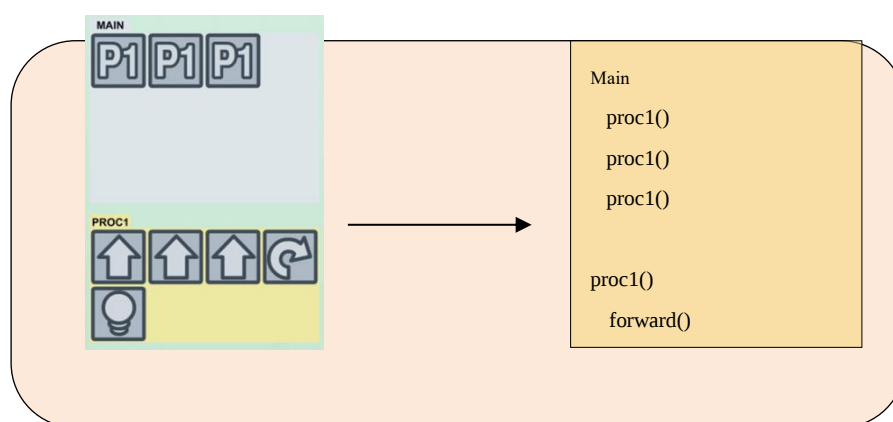


Figura 5. Comandos relativos à fase 1 nível 2 ambiente Lightbot.

Tanto neste jogo como no Lightbot, o professor tem a oportunidade de explorar movimentos similares com os do dia a dia. Contudo, o jogo Frozen possui características mais específicas, observe na Figura 6, a fase 2 deste jogo. Ele abrange tanto a linguagem de programação, quanto, também, o ensino específico de ângulos, incluindo ângulos que não são óbvios.

Para jogá-lo o aluno deve ter um conhecimento previamente adquirido, o que torna ainda mais aplicável na sala de aula, como uma forma de complementar a aula meramente exploratória.

O ambiente não possui separações em níveis. Ele é composto por 20 fases com graus de dificuldade gradativos. Nas três primeiras fases o aluno possui apenas comandos de direção e sentido, como por exemplo “Avance por 100 pixels”, “Vire à direita por 90 graus” ou “Vire à esquerda por 90 graus”. Na Figura 6 pode-se observar estes tipos de comando. Nos comandos citados o aluno pode alterar a quantidade de *pixels* que a personagem deve avançar e quantos graus Elsa pode virar. O professor, como mediador desta etapa, deve trabalhar ângulos maiores e menores que 90 graus e senso de direção.

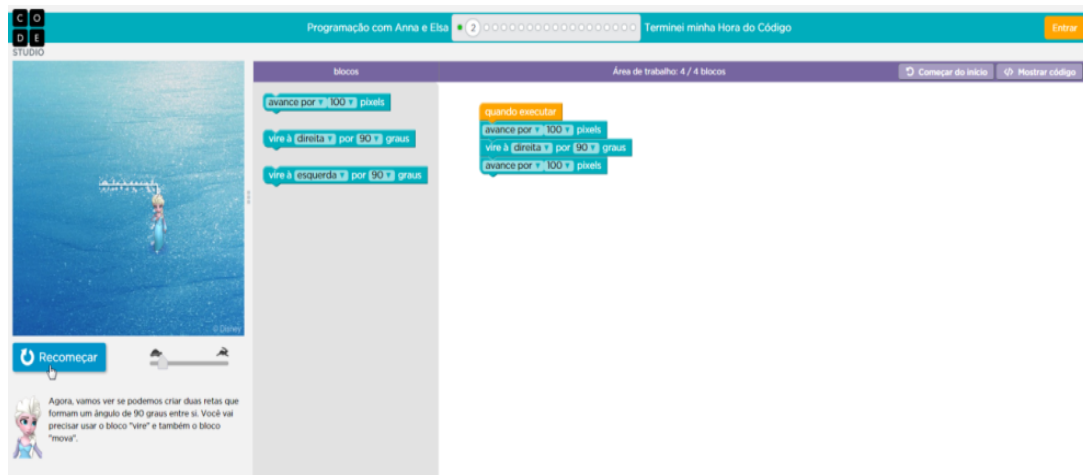


Figura 6. Fase 2 do jogo Frozen.

No término de cada fase o usuário (neste caso o aluno) recebe uma mensagem interativa, como mostrado na Figura 7. Esta mensagem diz se o usuário conseguiu ou não concluir a fase. Em caso positivo, existe a oportunidade, através dos comandos utilizados pelo ambiente, de visualizar o código relativo aos comandos escrito na linguagem de programação JavaScript. O próprio ambiente sinaliza quantas linhas de código o aluno desenvolveu.

A partir da fase 4 até a fase 8 o jogo explora as estruturas de repetição, como pode ser visto na Figura 8. Os movimentos repetidos N vezes podem ser descritos com comandos que maximizam as linhas de código. Nestas fases os alunos necessitam destas estruturas para completar suas fases utilizando linhas de código mais eficientes.



Figura 7. Mensagem referente à conclusão de cada fase.

Ao longo da fase 9 até a fase 19 o ambiente aumenta o grau de dificuldade. Ele utiliza recursos para colorir o trajeto feito pela personagem, criando, então, funções para esses movimentos. O ambiente explora diferentes ângulos (de 90° , 180° e 360°), incluído os ângulos que não são óbvios visualmente, como os ângulos de 5° , 30° , 36° , 60° e 100° entre outros.

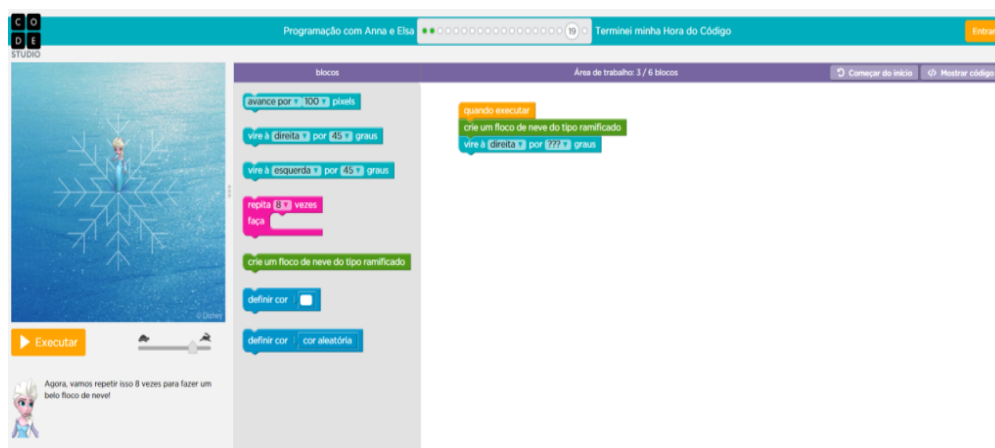


Figura 8. Estruturas de repetição do ambiente Frozen.

Na última fase, a fase 20, o aluno deve criar um movimento que use os comandos aprendidos e utilizados ao longo das 19 fases anteriores desse tutorial e simulador. Nesta etapa pode-se notar a característica relativa a modelagem. Neste momento, não existe o interesse de criar um programa com menos linhas de códigos, mas sim que explore de uma maneira interessante todos os comandos e possibilitando uma comparação de movimentos entre os alunos participantes desta atividade.

Definido os jogos que seriam utilizados, foi possível traçar um plano de execução da etapa de Realização, descrito na Tabela 1, que mostra o plano do experimento para ensino de programação.

O segundo encontro dividiu-se em dois momentos consecutivos e interligados. Foram desenvolvidas duas dinâmicas. A primeira dinâmica consistiu em montar um passo a passo para o caminho até a escola. Na segunda dinâmica, resultado mostrado na Figura 9, foi proposta a montagem do avião de papel. Nesta dinâmica, os alunos divididos em grupos, receberam uma folha A4 com instruções detalhadas para montar um avião de papel.

Tabela 1. Plano do experimento para ensino de programação.

Encontros	Planejamento
Primeiro	Aplicação de questionário avaliativo para conhecimento da realidade de cada aluno.
Segundo	Realização de dinâmica em grupo.
Terceiro e Quarto	Apresentação do ambiente <i>Code.org</i> , suas funcionalidades, organização e possibilidades. Apresentação para o aluno do ambiente <i>Labirinto</i> .
Quinto	Aplicação do questionário avaliativo sobre o ambiente <i>Labirinto</i> .
Sexto	Apresentação para o aluno do ambiente <i>Frozen</i> (realização até a fase 20).
Sétimo	Aplicação do questionário avaliativo sobre o ambiente <i>Frozen</i> .
Oitavo	Aplicação de teste escrito com questões envolvendo senso de direção sobre outro objeto.



Figura 9. Passo a passo da dinâmica do avião de papel.

3.2. Realização

A partir do terceiro encontro inicia-se a fase de Realização. O terceiro e sexto encontros consistem na apresentação dos jogos Labirinto e Frozen, respectivamente. Com o apoio do professor, os alunos obtiveram o acesso individual ao ambiente code.org.

Finalizado o cadastro, os alunos puderam explorar o ambiente, e em sequência iniciaram as tarefas a serem realizadas. A finalização das fases ocorreu ao longo da semana com o acompanhamento à distância pelo professor, utilizando a ferramenta de gerenciamento, mostrada pela **Tabela 2**. Nesta tabela, os alunos que não iniciaram as tarefas ficam com a numeração em branco, os alunos que iniciaram e não terminaram têm a numeração em amarelo, os alunos que concluíram a fase efetivamente ficam com a numeração em verde e os que utilizaram muitos blocos de comando ficam em verde mais claro.

Com base nos jogos realizados, o quinto e sétimo encontros foram destinados a realização de um questionário avaliativo. O questionário utilizado foi baseado no modelo de questionário MEEGA (Savi, 2011), que pode ser encontrado em <http://www.gqs.ufsc.br/avaliacao-de-jogos-educacionais/>. Este questionário foi adaptado para o contexto dos alunos, porque nem todos ainda estão familiarizados com números negativos.

No último encontro foi elaborada uma avaliação, baseada nas fases jogadas ao longo do experimento e que abordasse conceitos de ângulos. Para Freitas (2010), as provas devem ser criadas com alternativas problematizadas e a situação estímulo favorece a organização de ideias, dados ou informações para resolver qualquer problema. Neste caso, o aluno utiliza a situação estímulo para tentar resolver as questões propostas para ele. Todas as questões da avaliação final foram elaboradas de forma discursiva, envolvendo interpretação a partir de uma situação-estímulo que compôs o enunciado. Foi necessária a utilização de instrumentos como: régua e transferidor.

Tabela 2. Exemplo do progresso dos alunos no jogo Frozen (adaptado de code.org).

Aluno	Progresso Frozen
A1	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A2	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A3	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A4	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A5	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A6	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A7	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A8	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A9	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A10	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A11	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A12	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A13	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A14	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A15	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A16	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A17	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A18	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A19	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A20	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A21	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A22	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A23	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
A24	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Legenda	não iniciado em progresso concluído, muitos blocos concluído, perfeito

Na fase final do experimento (oitava), obteve-se 100% de frequência e os alunos necessitaram de um tempo maior para a realização desta avaliação. Além dos 50 minutos estimados para a sua realização, foi necessário aproximadamente mais 25 minutos para o seu término. O nervosismo dos alunos ao tentar resolver os problemas propostos era evidente, principalmente pelo fato do tempo estar se esgotando. Desta forma, é importante observar, através de um modo interpretativo, e analisar se houve aproveitamento com o ensino de ambientes computacionais neste cenário.

3.3. Análise, Interpretação e Apresentação dos Resultados

O sistema de avaliação pertencente às escolas, em geral, seguem o mesmo modelo de avaliação convencional. Como alternativa buscou-se a lógica Fuzzy comparada com a lógica convencional. A avaliação foi separada em três conjuntos qualitativos, *Insuficiente*, *Mediano* e *Satisfatório*. No cenário escolhido, cada aluno tem um grau de pertinência em cada um destes conjuntos. A seguir, está definida a função de pertinência desenvolvida para o problema proposto.

- Função de Pertinência triangular para o conjunto *Satisfatório*:

$$\pi(x) \begin{cases} 0, & \text{para } x \leq 5 \\ \frac{x-5}{2}, & \text{para } 5 < x \leq 7 \\ 1, & \text{para } 7 \leq x \end{cases}$$

- Função de Pertinência triangular para o conjunto *Mediano*.

$$\pi(x) \begin{cases} 0, & \text{para } x \leq 3 \\ \frac{x-3}{2}, & \text{para } 3 < x \leq 5 \\ \frac{7-x}{2}, & \text{para } 5 \leq x < 7 \\ 1, & \text{para } 7 \leq x \end{cases}$$

- Função de Pertinência triangular para o conjunto *Insuficiente*.

$$\pi(x) \begin{cases} 0, & \text{para } 5 \leq x \\ \frac{5-x}{2}, & \text{para } 3 \leq x < 5 \\ 1, & \text{para } x \leq 3 \end{cases}$$

Ao comparar pontualmente o sistema de avaliação convencional, exemplificado na Tabela 3, com a lógica Fuzzy, observa-se que 79,16% (19 alunos) possuem maior grau de pertinência em Satisfatório, enquanto que no Teste Final a porcentagem aumentou para 83% (20 alunos). No grau de pertinência em Mediano, o percentual encontrado foi de 12,5% (3 alunos) para ambos os métodos de avaliação. No sistema de avaliação convencional 8,3% (2 alunos) se concentram em Insatisfatório e 4,1% (1 aluno) no Teste Final

Tabela 3. Comparação entre avaliação convencional e a lógica Fuzzy.

Função de Pertinência	Teste Final		Avaliação Final Convencional	
	Porcentagem	Número de alunos	Porcentagem	Número de alunos
Satisfatório	83,3%	20	79,16%	19
Mediano	12,5%	3	12,5%	3
Insatisfatório	4,1%	1	8,3%	2

Ao visualizar a **Tabela 6**, que apresenta uma visão geral dos resultados obtidos pelos alunos, nota-se que houve uma melhora significativa nos indivíduos A1, A5, A10, A11, A12, A16, A17, A19, A21 e A24. Nestes casos, em especial, os alunos aumentaram o grau de pertinência para Mediano e Satisfatório.

O indivíduo A1 necessitou de uma atenção especial. A escola solicitou um acompanhamento psicológico e uma atenção em relação ao cumprimento dos deveres e trabalhos solicitados, bem como pontualidade e assiduidade.

O seu grau de pertinência até o 3º bimestre era equivalente a 1 (Insuficiente). Após o período de experimentação o aluno A1 obteve um rendimento Mediano e Satisfatório.

O aluno A10 estudava em outro colégio localizado em Caraguatatuba. Em atendimento com o responsável ele foi descrito como tímido e que apresentava dificuldade em trabalhar com os conteúdos matemáticos, pelo fato de não esclarecer suas dúvidas ao longo das aulas. O indivíduo A10 apresentava um rendimento 7,0, com grau de pertinência 1 (Satisfatório). Contudo, é importante destacar que mesmo o grau de pertinência ainda permanecendo 1 (Satisfatório), ele obteve um rendimento 9,4 na avaliação final.

O aluno A12 necessitou no ano anterior de um acompanhamento psicológico. Este aluno se encontrava muito disperso no período escolar. No período relativo ao 3º bimestre ele obteve grau de pertinência 0,75 para Mediano e 0,25 para Satisfatório. Após o período de experimentação este aluno aumentou o grau de pertinência para 0,65 (Satisfatório) e diminuiu o grau de pertinência relativo ao Mediano.

O aluno A17 está refazendo o 7º ano. A equipe pedagógica disponibilizou a informação de que ele não tem demonstrado envolvimento pelos conteúdos abordados e compromisso em relação aos trabalhos e deveres solicitados. O rendimento dele, anterior a atividade, era de 4,0 pontos referente ao 3º bimestre, com grau de pertinência 0,5 para Insuficiente e 0,5 para Mediano. Após o período de experimentação este aluno (A17) apresentou uma melhora significativa. O rendimento final foi de 9,4 e grau de pertinência 1 para Satisfatório.

Da mesma forma o aluno A19 está refazendo o 7º ano. A equipe pedagógica destacou o perfil deste aluno semelhante ao A17. Ao longo do 3º bimestre o rendimento foi 5,0 pontos com grau de pertinência 1 para Mediano. Ao longo da experimentação o aluno A19 obteve um Satisfatório, sendo seu rendimento final de 7,6 pontos e grau de pertinência 1.

Tabela 4. Avaliação dos resultados aplicado a lógica Fuzzy.

Aluno	Nota do 3º Bimestre	Grau de pertinência			Nota do Teste	Grau de pertinência		
		I	M	S		I	M	S
A1	2,5	1	0	0	6,0	0	0,5	0,5
A2	6,0	0	0,5	0,5	5,0	0	1	0
A3	10,0	0	0	1	10,0	0	0	1
A4	8,5	0	0	1	6,5	0	0,25	0,75
A5	5,6	0	0,7	0,3	8,1	0	0	1
A6	7,5	0	0	1	6,6	0	0,2	0,8
A7	10,0	0	0	1	9,4	0	0	1
A8	8,5	0	0	1	6,7	0	0,15	0,85
A9	10,0	0	0	1	9,5	0	0	1
A10	7,0	0	0	1	8,3	0	0	1
A11	6,0	0	0,5	0,5	7,3	0	0	1
A12	5,5	0	0,75	0,25	6,3	0	0,35	0,65
A13	10,0	0	0	1	7,3	0	0	1
A14	9,0	0	0	1	7,6	0	0	1
A15	9,5	0	0	1	8,7	0	0	1
A16	8,0	0	0	1	8,1	0	0	1
A17	4,0	0,5	0,5	0	9,4	0	0	1
A18	9,5	0	0	1	7,5	0	0	1
A19	5,0	0	1	0	7,6	0	0	1
A20	9,5	0	0	1	9,0	0	0	1
A21	7,4	0	0	1	8,0	0	0	1
A22	8,5	0	0	1	5,0	0	1	0
A23	10,0	0	0	1	7,6	0	0	1
A24	3,5	0,75	0,25	0	3,7	0,65	0,35	0

Insatisfatório – I

Mediano – M

Satisfatório – S

O aluno A21 apresenta um bom desempenho escolar. Referente ao 3º bimestre, ele obteve 7,4 pontos com grau de pertinência 1 para Satisfatório. Com um perfil de agitação e falta de concentração com diagnóstico de TDA-H, o A21 concluiu com uma melhora de 8,0 pontos e o grau de pertinência permaneceu em 1.

O aluno A24 veio de uma transferência externa e ao longo do ano apresentou um grande déficit de aprendizagem. Com rendimento 3,5 ponto, no 3º bimestre, ele obteve grau de pertinência 0,75 para Insatisfatório e 0,25 para Mediano. Mesmo o rendimento final sendo 3,7, o grau de pertinência para Insuficiente diminuiu e o grau de pertinência para Mediano aumentou para 0,65 e 0,35, respectivamente.

Em relação aos alunos A3, A7, A9, A13, A14 e A23, eles obtiveram rendimento Satisfatórios no 3º bimestre e posteriormente mantiveram-se no mesmo grau de pertinência respondendo positivamente ao novo método utilizado.

Os alunos A4, A6, A8 e A22 obtiveram graus de pertinências mais baixos em relação ao novo método. Contudo os alunos destacados não se encontram com nenhum grau de pertinência Insuficiente.

De uma forma geral, 66,6% dos indivíduos possuíam o maior grau de pertinência em Satisfatório, 20,9% no grau de pertinência Mediano e 12,5% no grau de pertinência Insuficiente. Após o experimento houve uma melhora significativa considerando não somente os casos individuais citados anteriormente, mas sim o total e alunos. Houve o aumento para 83,3% no grau de pertinência Satisfatório, redução para apenas 4,1% em relação ao grau de pertinência Insuficiente e 12,5% permanecendo no grau de pertinência Mediano.

4. Ambiente Algo+naEscola

A tarefa que envolve o processo de ensino aprendizagem é complexa e necessita de um comprometimento diário dos professores, cuja tentativa é acompanhar a demanda da tecnologia. Para Souza *et al.* (2016) ensinar e aprender num mundo caracterizado pelo grande fluxo de informações e diferentes tecnologias o professor deve, não somente buscar um bom material didático, mas também utilizar um método de busca e estudo.

O ambiente educativo desenvolvido e descrito nesta seção propicia ao professor um apoio e auxilia na utilização de tecnologias através de ferramentas computacionais. A descrição do ambiente está organizada da seguinte forma: descrição geral do ambiente, requisitos funcionais e não funcionais e a modelagem do ambiente.

4.1. Descrição Geral do Ambiente

O ambiente denominado Algo+naEscola tem os seguintes usuários: Administradores, Coordenadores, Professores e Visitantes. Os Administradores são responsáveis pelo gerenciamento dos usuários e das permissões concedidas para cada um deles. Os Coordenadores são responsáveis pela disponibilização de conteúdos adicionais. Além disso, são responsáveis por buscar recursos através de patrocinadores. Os Professores cadastrados realizam tarefas comuns a todos os observadores do ambiente e podem desenvolver funções como: visualizar os conteúdos disponibilizados, desenvolver planejamentos e criação de atividades e enviar mensagens.

Dentre as ferramentas de comunicação do ambiente, inicialmente a proposta se concentra no uso de um fórum de discussão. O ambiente disponibiliza planos de aulas já realizados, que têm como função o auxílio para futuros planos, e também a oportunidade de elaborar o plano de aula com o apoio de especialistas. O professor utiliza ferramentas variadas para estimular os alunos no dia a dia em sala de aula, contudo existem dúvidas quanto à utilização de algumas metodologias. O fórum de discussão auxilia o professor a questionar quais jogos e temas são adequados para uma proposta de aula. Entretanto, existem outras ferramentas como textos, questionários, dinâmicas, vídeos, apresentações, inclusive jogos computacionais que podem enriquecer os planos de aula.

Através do experimento utilizado nesta pesquisa, foram descritos requisitos funcionais e não funcionais, que estão descritos nas Tabelas 5 e 6, respectivamente.

Tabela 5. Requisitos funcionais.

#	RequisitoS	Breve descrição
RF01	Permitir o cadastro de usuário	Ao realizar o cadastro, o ambiente valida os dados pessoais do professor.
RF02	Permitir o <i>login/logout</i>	O ambiente deve autorizar o acesso do usuário cadastrado a qualquer momento.
RF03	Contatar Administrador	Permitir que o Visitante estabeleça contato com o Administrador.
RF04	Atualizar dados do professor	O professor cadastrado pode atualizar seus dados cadastrais armazenados.
RF05	Gerenciamento de perfis	O administrador pode atribuir e remover perfis de acesso dos usuários.
RF06	Busca de Planos de Atividades	Oferecer uma ferramenta de busca simplificada aos Planos de Atividades já cadastrados por tema, jogos utilizados.
RF07	Criar um Plano de programação referente a um tema/jogo	Permitir que o professor cadastrado construa um Planos de Atividades para um determinado tema ou jogo, sem utilizar planos já cadastrados.
RF08	Detalhar o Plano de atividades	Permitir que o professor informe as atividades que serão realizadas e os recursos que serão utilizados em cada encontro do planejamento.
RF09	Autenticidade	Os Planos de Aula submetidos devem ser identificados com os dados do autor.
RF10	Adaptar Planos de Atividades	Permitir que o Professor cadastrado construa seu Planos de Atividades utilizando fragmentos de planos de atividades já cadastrados.
RF11	Atualizar Planos de atividades	Permitir que o Professor cadastrado adicione e atualize as informações dos seus planos de atividades de acordo com as etapas.
RF12	Apoio à produção de atividades	O professor cadastrado pode contatar Coordenadores, Administradores e outros professores cadastrados para auxiliar a elaboração do Plano de Atividade relacionado a um tema ou jogo em questão.
RF13	Disponibilizar fórum de discussão para criação de um plano de aula	Permitir ao professor criar um fórum de discussão que possibilite e promova a colaboração entre o(s) Professor(es) que está(ão) usando o Plano de aula, Coordenadores, Administradores e Professores cadastrados.
RF14	Disponibilizar fórum de discussão para discussão de um jogo	Permitir ao professor criar um fórum de discussão que possibilite e promova a colaboração entre Coordenadores, Administradores e Professores cadastrados.
RF15	Análise de comentário	Permitir que Coordenadores e Administradores analisem o conteúdo das mensagens enviadas para o fórum, para identificar comentários inapropriados.
RF16	Notificar via <i>e-mail</i>	Quando um novo comentário for adicionado ao fórum de discussão os participantes deste fórum devem ser notificados via <i>e-mail</i> .
RF17	Informar a experiência na execução de um plano de atividade	Permitir que o professor cadastre suas observações identificadas durante a execução do plano de atividades como um todo (pontos positivos e negativos, possíveis melhorias, entre outros).
RF18	Informar a experiência na execução das atividades	Permitir que o professor cadastre suas observações sobre execução de uma atividade específica (pontos positivos e negativos, possíveis melhorias, ...).
RF19	Informar os dados de avaliação geral dos alunos	Permitir que o professor informe os parâmetros que devem ser utilizados para avaliar os alunos em cada atividade prevista no plano.
RF20	Informar os dados de avaliação dos alunos nas atividades	Permitir que o professor registre os resultados das avaliações individuais executadas em cada atividade do plano.
RF21	Fornecer dados estatísticos sobre avaliação dos alunos	O ambiente deve realizar uma análise estatística sobre os dados de avaliação dos alunos e disponibilizá-la para o professor cadastrado.
RF22	Fornecer análise fuzzy sobre avaliação dos alunos nas atividades	O ambiente deve avaliar cada aluno, com base na lógica fuzzy, utilizando todas as notas das avaliações de um plano de atividades. Apresentar um comparativo entre esta avaliação e a avaliação final do aluno informada.
RF23	Fornecer dados estatísticos gerais sobre avaliação dos alunos	O ambiente deve realizar uma análise estatística considerando todas as avaliações de um mesmo tema.

Tabela 6. Requisitos não funcionais.

Identificador	Requisito	Breve descrição
RNF01	Segurança	O ambiente deve salvaguardar as informações pessoais e prevenir acessos não autorizados.
RNF02	Fácil para usuários	O ambiente deve se organizar de forma centralizada e dinâmica. Os usuários devem localizar e acessar facilmente a informação correta com o mínimo de treinamento.
RNF03	Modelo de Plano de Aula	O ambiente deve oferecer um modelo para a produção do Plano de Aula. Este modelo se encontrará no ambiente para <i>download</i> .
RNF04	Impressão e <i>Download</i>	O ambiente deve oferecer as ferramentas de impressão e <i>download</i> dos planos de aula.
RNF05	Privacidade	Os dados dos nomes de escola, professores e alunos serão codificados e só o próprio autor terá acesso.

4.2. Modelagem do Ambiente

O ambiente permite uma ferramenta de busca inicial para os planos de atividades, onde o professor seleciona um tema para o plano de atividade. Os planos de atividades relacionados no resultado desta busca contém o mesmo tema da procura.

O professor também pode buscar um plano de atividades através de um recurso específico. Na busca inicial o usuário pode, através de um jogo computacional, encontrar outros planos de atividades que tenham utilizado o mesmo recurso. Desta forma, a busca inicial por planos de atividades já cadastrados pode ser realizada através de um tema ou um jogo. O professor pode buscar planos de atividades que auxilie a criação de um novo plano, com base nas experiências já cadastradas no ambiente.

Baseado no experimento descrito na Seção 3, o primeiro passo do plano de atividades é criar o escopo, traçando os objetivos em que os alunos terão de alcançar. Esses objetivos estão relacionados com o novo conteúdo a ser ensinado naquele momento, ou com a tentativa de recuperar um conteúdo que não foi aprendido. Deste modo, essas hipóteses formam um conjunto inicial de um determinado resultado a ser alcançado.

De uma forma geral, a proposta do ambiente se concentra na troca de experiências com o objetivo de apoiar professores a elaborar seus planos de atividades e registrar suas atividades e avaliações. A disponibilização deste ambiente fornece para o professor uma ferramenta de apoio simultâneo e flexível. O fórum auxilia para que outros professores possam trocar experiências no momento que o plano de atividade está sendo elaborado. Além de auxiliar o professor a elaborar seus planos de atividade, o ambiente também armazena as observações dos participantes do fórum. A Figura 10 apresenta um modelo de atividades, exemplificando o processo de construção do plano de atividades. A Figura 11 apresenta um modelo de classes que apresenta as ferramentas utilizadas através do ambiente, de modo que auxiliam o processo de elaboração do plano de atividades.

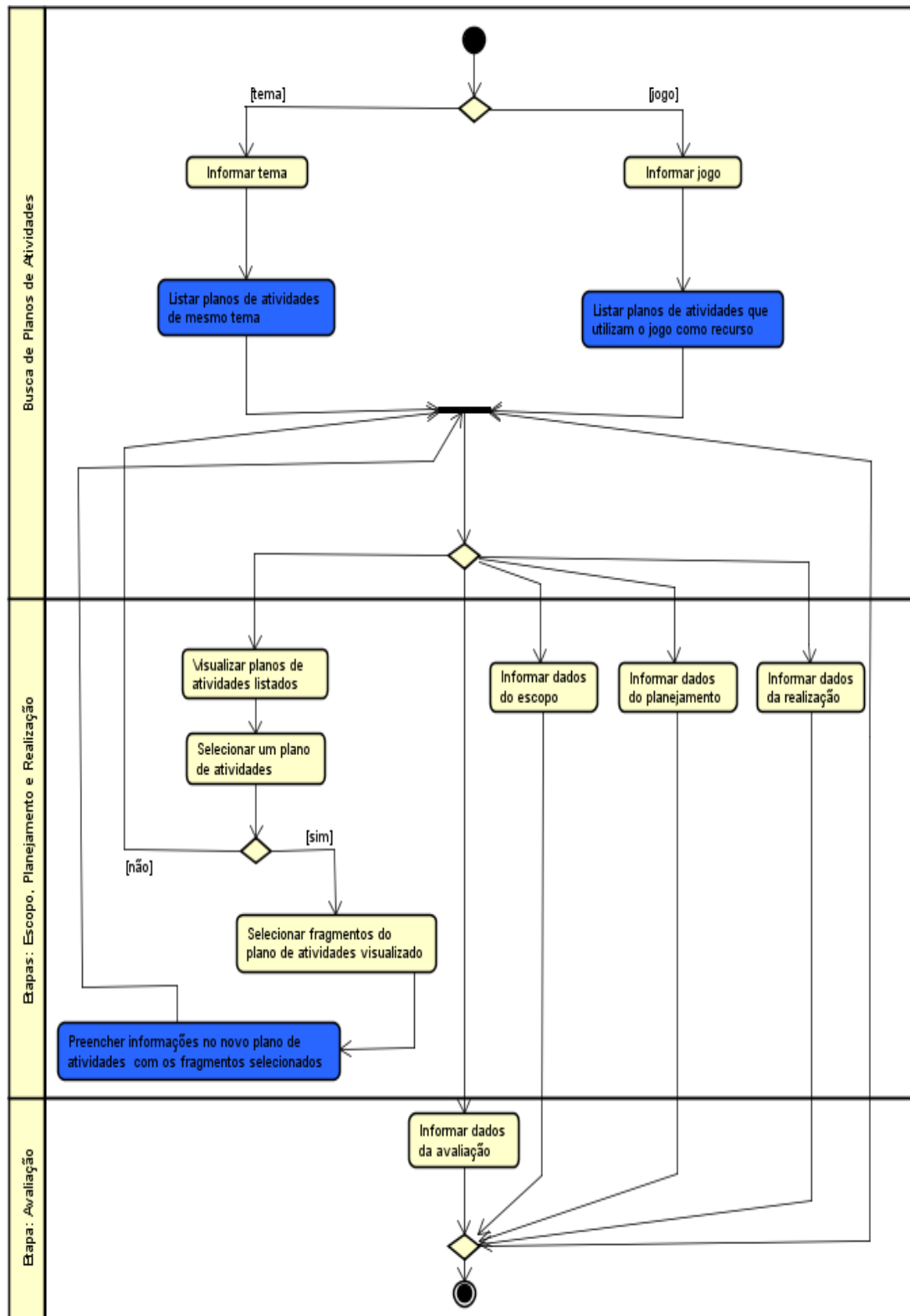


Figura 10. Modelo de Atividades criação do plano de atividades.

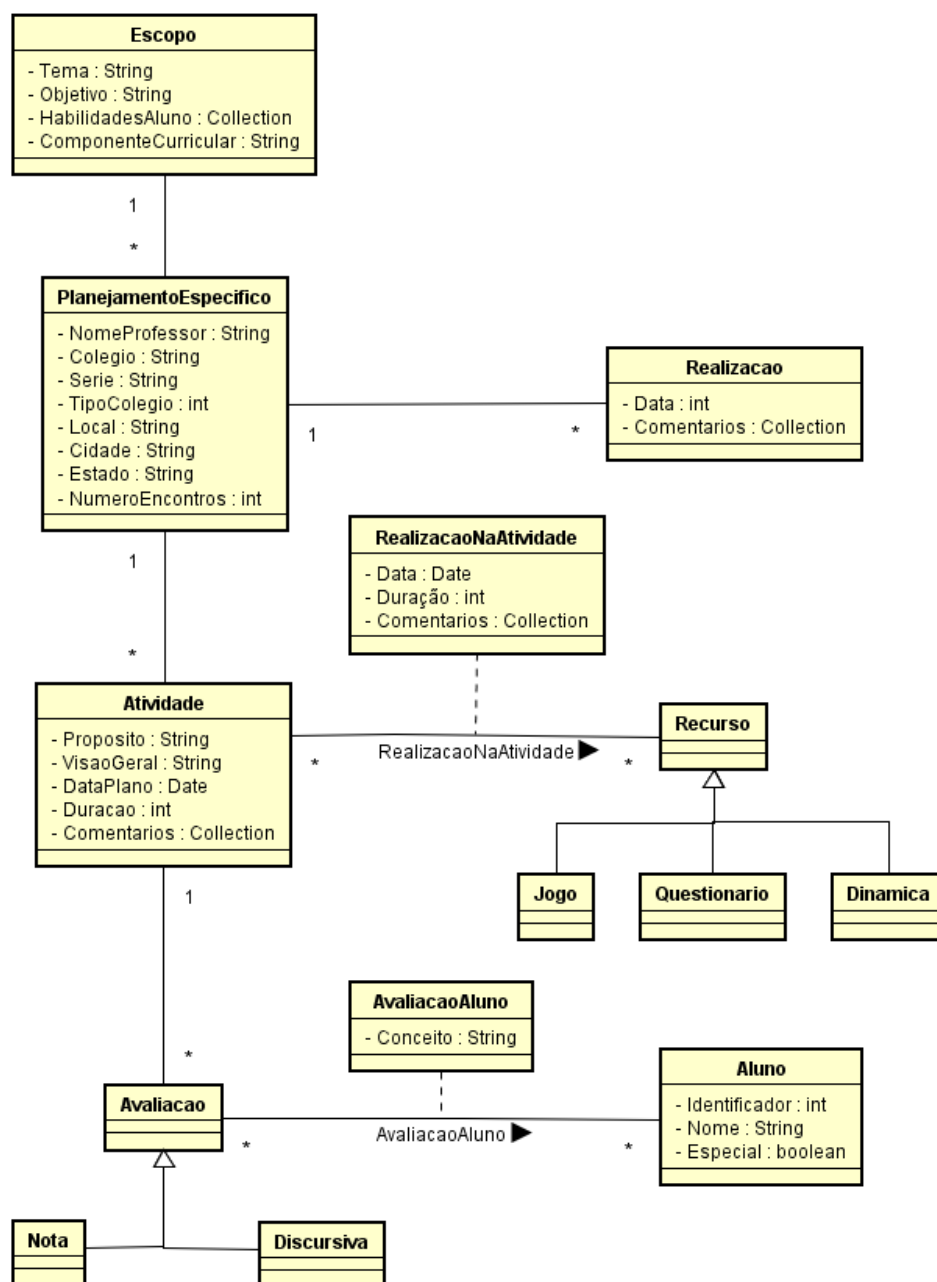


Figura 11. Modelo de classes referente à produção do plano de aula.

5. Conclusões

Este artigo teve como objetivo principal apresentar, os requisitos de um ambiente que proporcione a disponibilização de conteúdos didáticos e metodologias que envolvem o ensino de programação através do uso de jogos. Este ambiente promove a interação entre professores atuantes em diferentes níveis de ensino, desde o Ensino Fundamental até o Ensino Médio. O experimento realizado com uma turma do Ensino Fundamental II proporcionou uma contribuição na elaboração dos requisitos funcionais e não funcionais para a proposta do ambiente Algo+naEscola. Através deste experimento foi possível traçar diretrizes para um plano de atividades que tem a proposta de utilizar recursos

computacionais. Como trabalhos futuros, considera-se importante a implementação deste ambiente contendo todos os requisitos funcionais e não funcionais exemplificados nesta pesquisa, assim como todo o cenário confeccionado. É importante ressaltar que é essencial uma validação dos requisitos com um grupo de professores cadastrados para identificar possíveis ajustes nos requisitos coletados anteriormente.

Referências

- Bennett, S., Maton, K., & Kervin, L.. The 'digital natives' debate: A critical review of the evidence. *British journal of educational technology*, 39(5), 775-786. 2008.
- Freitas, C., Sacalabrin, E., Martins, V. (2010). Uma proposta de Processo Contínuo de Avaliação para Cursos de Ciência da Computação. XVIII Workshop de Educação em Informática (WEI). XXX Congresso da Sociedade Brasileira de Computação. Belo Horizonte, Minas Gerais, Brasil.
- Kitchenham, B., Charters, S. (2007) Guidelines for performing Systematic Literature Reviews in Software Engineering. Technical report - EBSE-2007-01, Software Engineering Group. School of Computer Science and Mathematics, Department of Computer Science, University of Durham, Durham, UK.
- Medeiros, T., Silva, T., Aranha, E. (2013). Ensino de programação utilizando jogos digitais: uma revisão sistemática da literatura. *RENOTE - Revista Novas Tecnologias na Educação*. Volume 11, No. 3, p. 1-10. Universidade Federal do Rio Grande do Sul. Porto Alegre, Rio Grande do Sul, Brasil.
- Partovi, H., Partovi, A. (2013). Code.org. Disponível em: <https://code.org/>.
- Pinchover, Y., Schor, J., Schor, I. (2014) CodeMonkey. Disponível em: <http://www.playcodemonkey.com>.
- Savi, R. (2011). *Avaliação de Jogos Voltados para a Disseminação do Conhecimento*. Tese de doutorado. Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento. Universidade Federal de Santa Catarina. Disponível em: http://www.gqs.ufsc.br/wp-content/uploads/2011/11/RafaelSavi_teseFinal_A5.pdf.
- Souza, L., Rodrigues, C., Leite, L. (2016). Tecnologia ou metodologia: aplicativos móveis na sala de aula. *Anais do Encontro Virtual de Documentação em Software Livre e Congresso Internacional de Linguagem e Tecnologia Online*. Junho. Volume 5, No. 1. Disponível em: http://www.periodicos.letras.ufmg.br/index.php/anais_linguagem_tecnologia/article/view/105519382.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., Wesslén, A. (2012). *Experimentation in Software Engineering*. January. Chapter 4, p. 45-54. Springer-Verlag Berlin Heidelberg.
- Yaroslavski, D. (2008). Lightbot. Disponível em: <http://www.lightbot.com>.
- Zadeh, L.A. (1965). Fuzzy sets. *Information and Control*. Volume 8(3):338-353.