

O incentivo da Língua Inglesa infantil por meio de jogo educativo

The encouragement of English Language learning for kids through educational games

Laura Costa Godinho

Graduanda do curso de Sistemas de Informação (UNIPAM).

E-mail: lauragodinho83@gmail.com

Juliana Lilis da Silva

Professora orientadora (UNIPAM).

E-mail: juliana@unipam.edu.br

Resumo: As crianças da geração atual estão cada vez mais receptivas ao meio tecnológico. Em gerações passadas, o uso de tecnologia na educação era muito questionado e o acesso muito restrito. Nos dias de hoje, a criança faz uso da tecnologia desde o primeiro ano de vida. Com isso, diversos tipos de jogos fazem parte da rotina infantil, oportunizando novos conhecimentos e aumentando as possibilidades de aprendizagem. Analisando esses fatos, foi desenvolvido um jogo educativo na modalidade infantil de até sete anos, objetivando o incentivo ao ensino da língua inglesa. O jogo permite que a aprendizagem ocorra de forma lúdica e divertida, fazendo com que o aluno se sinta motivado na busca pelo conhecimento. Para a criação do jogo, foi utilizado o SCRUM como processo de desenvolvimento, o *Blender* para modelagem de objetos e animações e o *Unity 3D* para a criação de cenários e scripts.

Palavras-chave: Língua inglesa. Jogos. Educação tecnológica. Gameificação.

Abstract: The children of the present generation are increasingly receptive to technological means. In generations past, the use of technology in education was very questioned and the very limited access. These days the child makes use of technology since the first year of life. Thus, different types of games are part of children's routine, providing opportunities for new knowledge and increasing learning opportunities. Analyzing these facts, we developed an educational game in the children's mode of up to seven years, aiming to encourage the teaching of English. The game provides with learning to occur in a playful and fun way, making students feel motivated in the pursuit of knowledge. To create the game we used the SCRUM as development process, Blender for modeling objects and animations and the Unity 3D to create scenarios and scripts.

Keywords: English language. Games. Technological education. Gamification.

1 INTRODUÇÃO

Alguns anos atrás, o mercado de jogos continha consoles para executar os jogos,

como os videogames, Playstation3, Xbox, entre outros. Contudo, no Brasil, não eram desenvolvidos jogos com essas plataformas, eram apenas para computadores, logo o desenvolvimento era para 3D ou para web.

O problema da geração passada era o alto custo de desenvolvimento, pois, quando se desenvolvem jogos 3D, são necessárias uma equipe muito capacitada e uma distribuição das unidades do jogo em forma de CD-ROM. Isso exige um custo e um risco muito alto e, conforme o custo vai se elevando, o risco também aumenta (TOYAMA, 2013).

No entanto, percebe-se que a tecnologia está evoluindo em quantidade e em curto prazo de tempo. Como consequência, houve o surgimento das ferramentas e dos dispositivos móveis como os *smartphones* e *iphones* e, positivamente, tornou-se mais fácil e barato o desenvolvimento de jogos. Assim, a evolução constante da tecnologia está impulsionando a educação para novos rumos, enfatizando a utilização de novas ferramentas e propiciando uma evolução no processo de ensino/aprendizagem (BRAGA, 2001).

Hoje em dia, existem diversos tipos de jogos, cada um com objetivos divergentes, uns para o divertimento, outros para o aprendizado ou treinamento e outros objetivam a divulgação de uma marca, enfim, jogos para diversas finalidades.

O jogo educativo é entendido como um recurso de interatividade que ensina, desenvolve e educa de forma prazerosa. Assim, ao atender às necessidades infantis, o jogo infantil torna-se uma forma adequada para a aprendizagem dos conteúdos escolares (MORCHIDA, 2011).

Nesse contexto, percebe-se que os jogos educativos estão, aos poucos, entrando no mercado, tanto para *desktop* quanto para *mobile*, e, aos poucos, estão proporcionando grandes resultados, como o aprendizado de conceitos e conteúdos, o desenvolvimento de habilidades, da criatividade e do raciocínio lógico.

Utilizar o jogo na educação infantil significa transportar para o campo do ensino-aprendizagem condições para maximizar a construção do conhecimento, introduzindo as propriedades do lúdico, do prazer, da capacidade de iniciação e ação ativa e motivadora (MORCHIDA, 2011).

Nesse contexto, foi proposto o desenvolvimento de um jogo para incentivar e auxiliar o aluno no processo de ensino-aprendizagem da língua inglesa. Para criação do jogo educacional, foram utilizadas ferramentas de modelagem e de desenvolvimento dos cenários.

2 REFERENCIAL TEÓRICO

Nesta seção, são apresentados os conceitos referentes ao processo de desenvolvimento do jogo educacional.

2.1 Computação Gráfica

Conceitos de computação gráfica podem ser apresentados a partir de extensões, ou seja, primeiro se aprende rotação, escalonamento e translação no espaço 2D e depois é feita a extensão para o espaço 3D.

Dois fatores, entretanto, foram fundamentais para o desenvolvimento da Computação Gráfica como a conhecemos hoje (TORI, 2008). Foi o desenvolvimento da tecnologia de circuitos integrados durante a década de 70 que permitiu o barateamento e a conseqüente popularização das máquinas. Houve o fim da ideia de que os fabricantes de computadores devem fornecer apenas a máquina e o sistema operacional e de que os usuários devem escrever seus próprios aplicativos.

A popularização dos aplicativos prontos e integrados (planilhas, editores de texto, editores gráficos, processadores de imagem, bancos de dados, etc.) permitiu a popularização da Computação Gráfica na medida em que possibilitaram que o usuário comum, sem conhecimento ou tempo para desenvolver aplicativos gráficos (nem sempre tão simples de serem programados), pudesse utilizar as facilidades da mesma (TORI, 2008).

2.2 Manifesto Ágil

O desenvolvimento ágil é qualquer processo de desenvolvimento criado com base nos conceitos do manifesto ágil. Ele foi assinado em 2001 e elaborado por profissionais experientes e veteranos da indústria de software. Embora cada um tivesse a sua forma de pensar e agir, todos estavam de acordo que os projetos de sucesso tinham certos princípios em comum. Esses princípios servem como diretrizes para equipes que procuram uma forma ágil de administrar seus processos de desenvolvimento (RODRIGUES, 2011).

O Manifesto Ágil apresenta oito fundamentos para serem seguidos durante o desenvolvimento (RODRIGUES, 2011):

- Qualidade do produto;
- Manutenção do mesmo ritmo de desenvolvimento;
- Simplicidade;
- Satisfação do cliente, com a entrega do produto no prazo e com qualidade;
- Entregas constantes em um período curto;
- Funcionalidades com mais prioridades implementadas;
- Abertura às mudanças constantes de requisitos, adequando o processo às novas necessidades;
- Comunicação eficaz entre os membros da equipe;
- Discussões em intervalos regulares para aperfeiçoar o processo de desenvolvimento.

Enfim, os métodos ágeis têm como sua principal vantagem a adaptabilidade às mudanças. Metodologias tradicionais tentam descobrir todas as possibilidades sem deixar espaço para descobertas durante o desenvolvimento.

2.3 Scrum

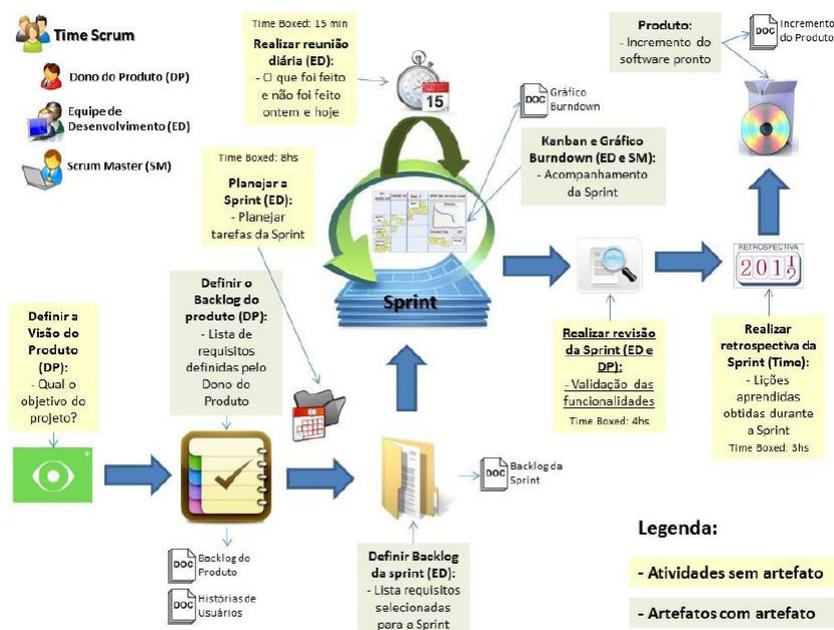
Criado no início da década de 1990 para gerência de projetos, o Scrum é um *framework* ágil indicado para projetos complexos, originalmente idealizado para projetos de desenvolvimento de software, mas funciona bem para qualquer âmbito, pois suas possibilidades de uso são infinitas (ALLIANCE, 2011).

Por ser um *framework*, essa ferramenta não diz o que fazer em cada etapa do desenvolvimento, devido ao fato de que, em projetos muito complexos, não se sabe exatamente qual decisão tomar em determinada situação, mas traz um conjunto de boas práticas para serem seguidas durante todo o processo. Dessa maneira, o Scrum mostra que é um processo extremamente adaptativo, se ajustando às mais variadas situações no decorrer do projeto para concluir o objetivo.

O Scrum segue rigorosamente o princípio KIS (*keep it simple* – mantenha simples). De acordo com seu princípio, segue o processo de desenvolvimento iterativo e incremental, no qual grandes problemas são divididos em problemas menores que podem ser resolvidos gradativamente em cada etapa até estar completamente resolvido.

As equipes Scrum apresentam como características principais a auto-organização, por decidirem qual a melhor forma de realizarem seu trabalho, em vez de serem subordinados a gerentes de projeto e a multifuncionalidade, pelo fato de apresentarem competências de concluir seu trabalho sem depender de alguém de fora do time. Esse modelo é projetado para aperfeiçoar a criatividade, a flexibilidade e a produtividade, conforme representado na Figura 1.

Figura 1 - O ciclo de vida do Framework SCRUM



Fonte: Revista Engenharia de Software Magazine, 2014

O time Scrum é composto por:

- *Product Owner*: este é o dono do produto, responsável pelo gerenciamento do *product backlog*, ou seja, definir as características do produto e o nível de prioridade de cada item.
- *Scrum Master*: tem a responsabilidade de liderar a equipe, lidar com o *product owner*, resolver os impedimentos que atrapalhem a produção e também garantir que a equipe cumpra com suas responsabilidades.

- Equipe de desenvolvimento: são os responsáveis por desenvolver e entregar um incremento no final de cada *sprint*, composta por no máximo nove e no mínimo três integrantes, não sendo incluído na contagem o *product owner* e o *scrum master*, ao menos que desempenhem alguma função no *sprint*. Os artefatos do Scrum representam o trabalho ou o valor das várias maneiras que são úteis no fornecimento de transparência e oportunidades para inspeção e adaptação. (REVISTA ENGENHARIA DE SOFTWARE MAGAZINE, 2014)

Os artefatos definidos para o Scrum são especificamente projetados para maximizar a transparência das informações-chaves necessárias para assegurar que o time Scrum tenha sucesso na entrega do incremento funcional (SCHWABER; SUTHERLAND, 2013).

Um artefato é chamado de *sprint backlog*, composto por todos os itens que irão entrar no *Sprint*, que é definido pela equipe de acordo com a complexidade da história e a prioridade do *product owner*. Esse documento permite alterações no decorrer dos *sprint*, mas somente a equipe tem autonomia para retirar ou acrescentar novos itens.

Outro artefato chamado *product backlog* é uma lista ordenada de acordo com a prioridade de cada item. Todas as funcionalidades e características que o software deverá apresentar são chamadas de *user stories* (histórias de usuário), que são compostas pelo nome de cada requisito e sua descrição de forma bem simples em nível de usuário. Cada história corresponde a um caso de uso utilizado no RUP (Rational Unified Process).

O último artefato utilizado no Scrum é o gráfico *Burndown*, que demonstra a quantidade de trabalho do *sprintbacklog* que já foi executado e a quantidade restante de maneira simples e clara.

Por ser um processo iterativo e incremental, o Scrum trabalha em ciclos em que cada iteração é chamada de *sprint*, um período de duas a quatro semanas em que a equipe de desenvolvimento trabalha para produzir um incremento a partir do *product backlog*. Durante esse processo, acontecem as reuniões de *Sprint Planning Meeting*, processo de criação do *product backlog* e definição de duração do *sprint*.

Diariamente, durante o *sprint*, acontecem as *daly scrum*, que são reuniões diárias rápidas de, no máximo, quinze minutos, nas quais cada membro da equipe informa qual tarefa concluiu ou se teve algum impedimento e qual tarefa irá efetuar no decorrer do dia. Outro acontecimento é o *Sprint Review*, que acontece sempre no final do *sprint*, apresentando o incremento a todas as partes interessadas para inspeção do resultado final do *sprint*. Como último item, há o *Sprint Retrospective*, uma reunião entre o *scrum master* e a equipe para inspecionar e discutir melhorias para os próximos *sprints*.

3 METODOLOGIA

Para o desenvolvimento do jogo, foram utilizadas as seguintes ferramentas:

- *Blender*, para modelagem de objetos 3D, gerando cenários e suas características necessárias. Trata-se de uma ferramenta *open source*.

- *Unity 3D*, essa ferramenta foi utilizada para a criação das *engines* do ambiente virtual, por meio de scripts que foram desenvolvidos a partir da linguagem JavaScript.

Como metodologia de desenvolvimento, foi utilizado o Scrum. Por ser um processo em ciclos, o Scrum passa por diversas etapas até a conclusão do produto. O ponto inicial é a construção do *product backlog*, na qual o *product owner* constrói as *user storys*. O quadro 1 mostra as *users storys* referentes ao jogo.

Quadro 1 - Histórias de Usuários

<i>User Storys</i>		
ID	Nome	Descrição
1	Definir funcionamento	Definição do funcionamento da aplicação e regras do jogo.
2	Definir tecnologia	Definição da tecnologia.
3	Definir interação	Definição do tipo de interação do jogo.
4	Preparar os objetos	Obtenção e preparação dos objetos virtuais em 3D, como criação e aplicação das texturas.
5	Construir cena	Construção das cenas do jogo.
6	Modelar personagem principal	Modelagem do boneco principal do jogo educacional.
7	Construir aplicação sobre os cenários	Aplicação de efeitos visuais sobre os cenários.
8	Construir scripts com objetos	Construção dos <i>scripts</i> envolvidos com os objetos 3D.
9	Construir scripts com cenários	Construção dos <i>scripts</i> com ações dos cenários.
10	Aplicar teste	O teste da usabilidade do sistema, interferência psicológica de uso do sistema será feito após os alunos experimentarem o jogo.

Fonte: Elaboração das autoras, 2014.

Por ser iterativo e incremental, esse processo se repete até que o produto esteja completamente pronto.

4 DESENVOLVIMENTO E RESULTADOS

As *user storys* foram desenvolvidas tanto na implementação e construção de scripts das cenas e objetos quanto nos testes direcionados na interação do personagem com os ambientes.

Na primeira fase do jogo, foram criadas as cenas (poças de água, texturas de todo o chão, céu, árvores, gramas e pedras) e os objetos (cadeiras, mesa, computador, celular, jarra e livros). Na sequência, foi determinado que o personagem será instruído por meio de mensagem na tela, cujas tarefas deverão ser executadas pela criança para dar continuidade ao jogo. Foi utilizado áudio para reproduzir o som de objetos e retornar erro ou acerto na execução das tarefas.

Na segunda fase do jogo, o participante será submetido a obstáculos até chegar a seu destino. Nessa fase, o participante terá que passar por um labirinto, eliminando os ratos que aparecem no caminho e completando a frase que também será exibida na tela, de acordo com os objetos que são encontrados no final do percurso. Os sons também serão reproduzidos para indicar se o objeto foi pegado na ordem certa.

A Figura 2 mostra a logomarca que foi criada para o jogo, possuindo duas palavras, uma referenciando o foco do jogo, educação, e a outra o tema, ensino do inglês. O símbolo que fica entre as palavras remete a um livro de ensino-aprendizagem.

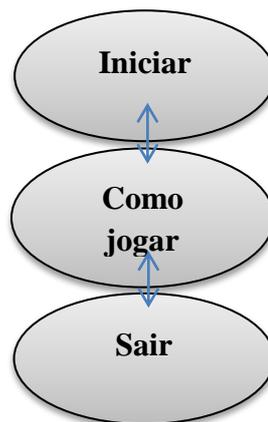
Figura 2 – Logomarca do jogo educativo



Fonte: Elaboração das autoras, 2014.

A Figura 3 indica o protótipo da tela inicial do jogo, com as três opções: “como jogar”, que indica as instruções do jogo, “iniciar” e “sair do jogo”.

Figura 3 – Protótipo do menu principal



Fonte: Elaboração das autoras, 2014.

A Figura 4 destaca um dos códigos feitos na linguagem javascript, o qual possui duas funções com objetivos diferentes: a função *Start()*, que é executada apenas na inicialização do jogo, das variáveis e scripts, e a função *Update()*, que é executada

durante todo o jogo e por várias vezes. Conforme o código abaixo, na primeira função, inicializou-se uma variável e, na segunda função, o método *loadedLevel()* foi implementando para chamar a próxima fase.

Figura 4 – Implementação de funções

```

1 #pragma strict
2
3 static var ponto: int;
4
5 function Start ()
6 {
7     ponto=0;
8 }
9
10 function Update ()
11 {
12     if (ponto==40)
13     {
14         Application.loadedLevel("FASE 2");
15     }
16 }
17
18 }

```

Fonte: Dados do trabalho, 2014.

A Figura 5 destaca o menu principal que é constituído da logomarca do jogo e de três botões, os quais contemplam as atividades iniciais que podem ser exploradas pelo usuário/criança.

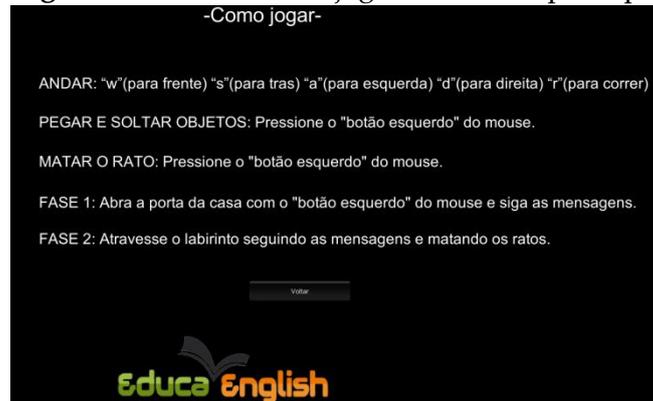
Figura 5 – Menu principal



Fonte: Dados do trabalho, 2014.

A Figura 6 representa a tela que será apresentada caso o usuário, no menu principal, escolha a opção “Como jogar”. O objetivo dessa tela é informar ao usuário as instruções que ele deve executar para realizar as tarefas propostas no jogo.

Figura 6 – Botão “Como jogar” do menu principal
-Como jogar-



Fonte: Dados do trabalho, 2014.

A Figura 7 mostra o personagem criado para esse jogo. O personagem será um cão, o qual foi escolhido por possibilitar a utilização do sistema tanto por meninas quanto por meninos, pois chama a atenção de ambos.

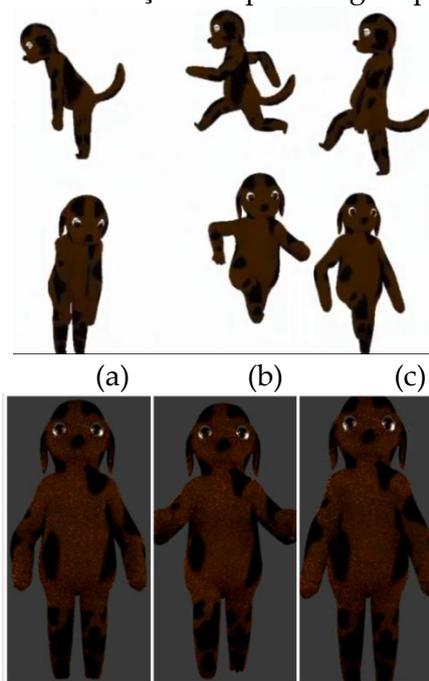
Figura 7 - Personagem principal



Fonte: Dados do trabalho, 2014.

A Figura 8 mostra as animações do personagem. Na Figura 8(a), o personagem está parado, abaixando para pegar algum objeto, na Figura 8(b) ele está correndo e a Figura 8(c) descreve um dos movimentos que pode ser utilizado para eliminar o rato. Todas as animações foram modeladas no *Blender* e adaptadas a cada tipo de cena do jogo.

Figura 8 – Animações do personagem principal



Fonte: Dados do trabalho, 2014.

A Figura 9 destaca mais alguns objetos do jogo (caixa, jarra, relógio) e um rato, que será o obstáculo e a diversão para a criança. Ela terá que eliminá-lo para dar continuidade à fase. Optou-se por objetos do dia-a-dia, pois acredita-se que esses são mais fáceis de serem lembrados pelo usuário/aluno.

Figura 9 – Objetos diversos



Fonte: Dados do trabalho, 2014.

A Figura 10 mostra o cenário principal e os objetos contidos no mesmo. Esses objetos deverão, no decorrer do jogo, ser traduzidos para a língua inglesa. Caso o usuário/criança não consiga efetuar a tradução, ele não passará para a fase seguinte.

Figura 10 – Cenário e objetos da primeira fase



Fonte: Dados do trabalho, 2014.

A Figura 11 mostra um dos cenários modelados para a primeira fase, no qual pode-se observar que as árvores são ajustadas de acordo com a cena, ou seja, assim que se definiu o cenário, foram identificados quais objetos seriam mais adequados para compor o ambiente.

Figura 11 – Árvores da primeira fase



Fonte: Dados do trabalho, 2014.

A Figura 12 mostra outro cenário composto por céu, montanhas e coqueiros, o qual também faz parte da primeira fase do jogo. Todavia, o céu é alterado para amanhecer, entardecer e anoitecer a cada dois minutos, com o objetivo de dar mais realidade às cenas. Na Figura 12, é exibido o estado entardecer.

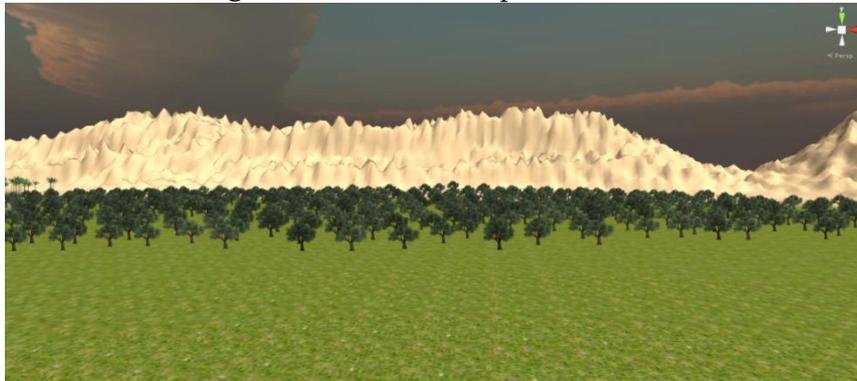
Figura 12 – Cenário da primeira fase



Fonte: Dados do trabalho, 2014.

Na Figura 13, pode-se observar outro cenário, o qual é constituído por montanhas, gramas e árvores, sendo todos desenvolvidos no *Unity 3D*.

Figura 13 - Cenário da primeira fase



Fonte: Dados do trabalho, 2014.

A Figura 14 mostra o personagem juntamente com o cenário da primeira fase. Nessa fase, o objetivo é fazer o personagem se aproximar de cada objeto que está dentro da casa. Ao aproximar-se do objeto, uma mensagem será exibida com a tarefa a ser cumprida. Em seguida, o usuário/criança deverá pegar um objeto, se aproximar de cinco caixas, identificadas com a palavra traduzida para o inglês, e colocar o objeto dentro da caixa correta. Ao colocar o objeto na caixa, um som será emitido para retornar ao usuário/criança se a opção escolhida está correta ou incorreta. Ao identificar a qual caixa pertence cada objeto, automaticamente, o jogador passará para a próxima fase.

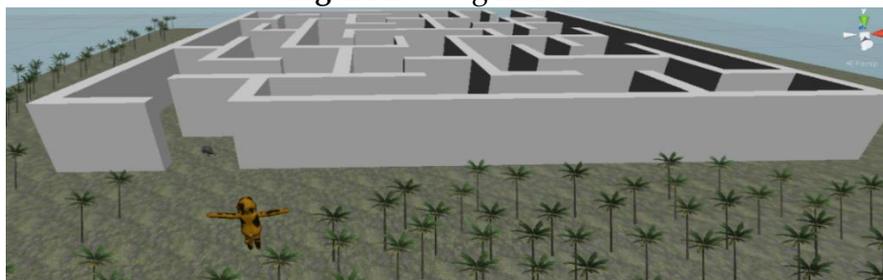
Figura 14 – Primeira fase



Fonte: Dados do trabalho, 2014.

A Figura 15 mostra o segundo cenário do jogo, o personagem principal é um rato. O objetivo é que o personagem percorra todo o labirinto, com o apoio do mapa, e elimine todos os ratos que aparecerem no decorrer do labirinto. No final do labirinto, serão encontrados dois objetos que deverão ser colocados na ordem correta, completando, corretamente, a frase que será exibida na tela. Assim, ao terminar de percorrer o labirinto, o jogo é finalizado.

Figura 15 – Segunda fase



Fonte: Dados do trabalho, 2014.

A Figura 16 mostra a tela que aparece quando o personagem perde a vida, ou seja, é derrotado pelos três ratos dentro do labirinto. Caso o usuário/criança queira jogar novamente, basta clicar no botão “Jogar Novamente”.

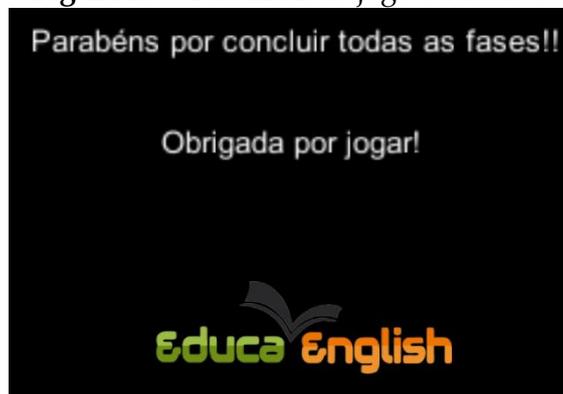
Figura 16 – Tela da segunda fase



Fonte: Dados do trabalho, 2014.

A Figura 17 mostra a tela de conclusão do jogo, indicando que o usuário/criança conseguiu alcançar todos os objetivos propostos nas duas fases do jogo – aprender a língua estrangeira de uma maneira diferente e motivadora.

Figura 17 – Tela final do jogo educativo



Fonte: Dados do trabalho, 2014.

Conforme apresentado nas imagens, pretendeu-se criar cenários com uma boa modelagem, que pudessem ser de fácil interação pelo usuário e que possibilitassem um resultado satisfatório na aprendizagem da língua inglesa.

5 CONCLUSÃO

O jogo teve o objetivo de incentivar o ensino-aprendizagem da língua inglesa para o público infantil, especificamente para crianças de até sete anos de idade, de uma forma diferenciada, e proporcionando a aprendizagem lúdica e efetiva para essa faixa etária.

O jogo foi desenvolvido em duas fases, sendo que, na primeira, o foco estava na aprendizagem e descoberta de todo o ambiente e, na segunda, o foco estava na aprendizagem e na diversão.

Compreende-se, por meio deste estudo, que os jogos educativos estão cada vez mais presentes na educação das crianças, com atividades que estão de acordo com a faixa etária das mesmas.

Como projeto futuro, pretende-se testar o jogo na escola English Studio, com a finalidade de verificar a usabilidade e a interação da criança ao utilizá-lo. Outro projeto seria a migração do jogo para aplicativos *mobile* em sistema operacional *Android* e o desenvolvimento de novas fases e atividades que estejam dentro do contexto do jogo.

REFERÊNCIAS

ALLIANCE, Scrum. *Scrum Is an Innovative Approach to Getting Work Done*. Disponível em: <http://www.scrumalliance.org/pages/what_is_scrum>. Acesso em: 17 mar. 2014.

BRAGA, Mariluci. Realidade Virtual e Educação. *Revista de Biologia e Ciências da Terra*. 2001. Disponível em:

<<http://eduep.uepb.edu.br/rbct/sumarios/pdf/realidadevirtual.pdf>>. Acesso em: 11 nov. 2014.

MORCHIDA, Tizuko. O jogo e a educação infantil. In: _____. *Jogo, brinquedo, brincadeira e a educação*. São Paulo: Cortez, 2011, págs. 15-48.

REVISTA ENGENHARIA DE SOFTWARE MAGAZINE 54. *Scrum Master*. Disponível em: <<http://www.devmedia.com.br/revista-engenharia-de-software-magazine-54/26601>>. Acesso em: 17 mar. 2014.

RODRIGUES, Luiza. *O que é desenvolvimento ágil*. Disponível em: <<http://blog.myscrumhalf.com/2011/05/faq-scrum-o-que-e-desenvolvimento-agil/>>. Acesso em: 13 mai. 2011.

SCHWABER, Ken; SUTHERLAND, Jeff. *Guia do Scrum*. Disponível em: <<http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum%20Guide%20-%20Portuguese%20BR.pdf#zoom=100>> Acesso em: out. 2011.

TOYAMA, André. *Visão geral do desenvolvimento de jogos*. Disponível em: <<http://www.infoq.com/br/interviews/visao-desenvolvimento-jogos/>>. Acesso em: 07 out. 2013.

TORI, Romero. *Computação gráfica, jogos e mundos virtuais na educação*. Disponível em: <<http://www.arquiamigos.org.br/seminario3d/pdf/tori-jogosmundosvirtuais.pdf>>. Acesso em: 06 nov. 2011.