

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

GILSON TRAJANO SOARES COSTA

Imersão Total: um RPG educativo para apoiar a aprendizagem dos conceitos básicos de programação

RIO TINTO-PB
2016

GILSON TRAJANO SOARES COSTA

Imersão Total: um RPG educativo para apoiar a aprendizagem dos conceitos básicos de programação

Monografia apresentada ao Curso de Licenciatura em Ciências da Computação do Centro de Ciências Aplicadas e Educação (CCAE), Campus IV da Universidade Federal da Paraíba, para obtenção do título de Licenciado.

Orientadora: Profa. Vanessa Farias Dantas.

RIO TINTO-PB
2016

C837i Costa, Gilson Trajano Soares.

Imersão Total: um RPG educativo para apoiar a aprendizagem dos conceitos básicos de programação. / Gilson Trajano Soares Costa. – Rio Tinto: [s.n.], 2016.
71 f. : il.

Orientador (a): Profa. Msc. Vanessa Farias Dantas.
Monografia (Graduação) – UFPB/CCAE.

1. Programação. 2. Jogos educativos. 3. Computação - ensino e aprendizagem.

UFPB/BS-CCAE

CDU: 004:37+51(043.2)

GILSON TRAJANO SOARES COSTA

Imersão Total: um RPG educativo para apoiar a aprendizagem dos conceitos básicos de programação

Trabalho de Conclusão de Curso submetido ao Curso de Licenciatura em Ciência da Computação da Universidade Federal da Paraíba, Campus IV, como parte dos requisitos necessários para obtenção do grau de LICENCIADO EM CIÊNCIA DA COMPUTAÇÃO.

Assinatura do autor: _____

APROVADO EM: _____ / _____ / _____

Orientadora: Profa. MSc. Vanessa Farias Dantas
Universidade Federal da Paraíba-Campus IV

Profa. Dra. Thereza Patrícia Pereira Padilha
Universidade Federal da Paraíba-Campus IV

Profa.MSc. Jarbele Cássia da Silva
Universidade Federal da Paraíba-Campus IV

RIO TINTO-PB
2016

AGRADECIMENTOS

Primeiramente a Deus que permitiu que tudo isso acontecesse, ao longo de minha vida, e não somente nestes anos como universitário, mas que em todos os momentos é o maior mestre que alguém pode conhecer.

Agradeço a minha orientadora pelo suporte, por suas correções e incentivos. Em especial à Glazy Rodrigues, minha noiva, uma pessoa iluminada que esteve comigo compartilhando essa nova fase, me encorajando e ajudando a finalizar mais esse passo em minha vida.

Meus agradecimentos aos meus amigos e colegas que fizeram parte da minha formação e que me ajudaram nos meus últimos passos.

E a todos que direta ou indiretamente fizeram parte da minha formação, o meu muito obrigado.

RESUMO

Com as dificuldades apresentadas pelos alunos nas disciplinas de programação, faz-se necessário que haja uma iniciativa que busque estimular os alunos na prática de programação, para que, desse modo, possam desenvolver suas competências. Para tanto, é necessária a resolução de muitas listas de exercícios, o que pode leva o aluno à desmotivação, uma vez que esta atividade é monótona e repetitiva. Tem-se a necessidade da criação de um material que motive e estimule o aluno na sua prática diária de programação e, para tanto, foi utilizado o jogo de RPG (*Role-Playing game*) educativo como um recurso para permitir aos jogadores a efetuação de suas atividades e estudos de forma lúdica, reforçando e complementando a aprendizagem dos conceitos básicos de programação Java como tipo de variáveis, estrutura condicional e laço de repetição. O uso do jogo pode motivar os alunos para o estudo, pois estarão concretizando uma atividade prazerosa enquanto praticam os conceitos aprendidos. O presente trabalho expõe o projeto de um jogo educativo com o objetivo de motivar os alunos no estudo dos assuntos iniciais de programação Java para complementar os estudos.

Palavras-chave: ensino de programação, jogos educativos, jogos digitais, RPG educativo, software educacional.

ABSTRACT

The difficulties presented by the students in programming disciplines, it is necessary that there is an initiative that seeks to encourage students in the practice of programming, so thus can develop their skills. Therefore, it is necessary to solve many lists of exercises, which can lead the student to demotivation, since this activity is monotonous and repetitive. There is the need to create a material that motivate and encourage students in their daily practice schedule and, therefore, we used the RPG (role-playing game) education as a feature to allow players to effectuation its activities and studies in a playful manner, strengthening and complementing the learning of the basic concepts of Java programming as type variables, conditional structure and repeating loop. The use of the game can motivate students to study, because they will be realizing a pleasurable activity while practicing the concepts learned. This work presents the design of an educational game with the aim to motivate students in the study of the initial subjects of Java programming to complement their studies.

Keyword: programming education, educational games, digital games, educational RPG, educational software.

LISTA DE FIGURAS

Figura 1 HUD do jogo Imersão Total	21
Figura 2 Introdução do jogo	25
Figura 3 Desafio das chaves.....	26
Figura 4 Desafio dos cogumelos	27
Figura 5 Desafio dos caminhos	29
Figura 6 Desafio abrir o portão	30
Figura 7 Desafio sair da cidade Condicional.....	30
Figura 8 Desafio monstro do loop.....	32

LISTA DE TABELAS

Tabela 1 Vantagens e desvantagens dos jogos (Grando, 2001)	17
Tabela 2 Escopo do jogo Imersão Total.....	24
Tabela 3 Fase 01: Cidade Inicial	28
Tabela 4 Fase 02: Cidade Condisional.....	31
Tabela 5 Fase 03: Masmorra	33

LISTA DE SIGLAS

RPG	<i>Role-Playing game</i>
NPC	<i>Non Player Character</i>
XP	<i>Experience Points</i>
XAS	<i>Xiderowg Action System</i>
HUD	<i>Heads-Up Display</i>
HP	<i>Health Point</i>
CT	<i>Combat Time</i>
SP	Pontos Mágicos
LV	Nível do jogador
MMORPG	<i>Massively Multiplayer Online Role-Playing Game</i>

SUMÁRIO

1. INTRODUÇÃO	12
1.1. MOTIVAÇÃO	12
1.2. OBJETIVOS	12
1.2.1. Objetivo Geral	12
1.2.2. Objetivos Específicos	13
1.3. METODOLOGIA	13
1.4. ESTRUTURA DO TRABALHO	13
2. FUNDAMENTAÇÃO TEÓRICA	14
2.1. Ensino-aprendizagem de Programação	14
2.2. Ferramentas para o Ensino-aprendizagem de Programação.....	14
2.2.1. Ferramentas que auxiliam o ensino de Java	15
2.3. Jogos Educativos para o Ensino-aprendizagem de Programação.....	16
2.4. Jogos de RPG Educativo	18
2.5. Trabalhos Relacionados	19
3. DESENVOLVIMENTO DO JOGO IMERSÃO TOTAL	21
3.1. Ferramentas e Materiais Utilizados no Desenvolvimento.....	21
3.2. O Desenvolvimento do Jogo	22
3.3. Características Educativas e Motivacionais	22
4. IMERSÃO TOTAL.....	24
4.1. Descrição geral do jogo Imersão Total.....	24
4.1.1. Descrição da Primeira Fase	25
4.1.2. Descrição da Segunda Fase	29
4.1.3. Descrição da Terceira Fase.....	31
4.2. Avaliação do Jogo Imersão Total	33
5. CONCLUSÃO	40
6. REFERÊNCIAS	41
7. APÊNDICE A – Documento de game design do jogo Imersão Total	44
8. APÊNDICE B – Roteiro do jogo Imersão Total	65
9. APÊNDICE C – Questionário utilizado na avaliação do jogo Imersão Total	70

1. INTRODUÇÃO

Este capítulo apresenta a motivação do trabalho, o seu objetivo geral, bem como os seus objetivos específicos, uma breve descrição da metodologia empregada e a estrutura deste trabalho que é uma pesquisa bibliográfica sobre as dificuldades no ensino de programação e os jogos e ferramentas usadas para sanar tais dificuldades, para dar-se inicio ao desenvolvimento de um jogo de RPG para apoia a aprendizagem de programação Java.

1.1. MOTIVAÇÃO

A disciplina de programação tem por objetivo desenvolver a capacidade dos alunos de analisar e resolver problemas na forma de algoritmos. De acordo com Borges (2000), essa disciplina costuma ter altos índices de evasão e reprovação, sendo um dos gargalos existentes nos cursos de exatas, dificultando ou impedindo a continuidade dos alunos no curso.

Tal índice de evasão existe, uma vez que muitos alunos não conseguem desenvolver o raciocínio lógico necessário para o posterior desenvolvimento de programas. Há também a dificuldade para estudar, uma vez que estão acostumados com disciplinas que apenas exigem memorização, e quando se deparam com os conceitos de programação que necessitam um entendimento mais profundo para a posterior aplicação na codificação, sentem uma baixa motivação e autoestima.

Vale ressaltar que o modelo de apresentação do conteúdo utilizado pelos professores pode levar à desmotivação dos alunos, e a exigência de muitas horas de dedicação na resolução de listas de exercícios para aprender programação pode ser pouco estimulante. Diante dos fatos, o presente trabalho apresenta um jogo, chamado Imersão Total, para apoiar a aprendizagem de programação Java em seus conceitos mais básicos como tipo de variável, condicional e laços de repetição.

1.2. OBJETIVOS

A seguir é apresentado o objetivo geral, bem como os objetivos específicos.

1.2.1. Objetivo Geral

Desenvolver o protótipo de um jogo de RPG educativo para apoiar a aprendizagem dos conceitos básicos de Java, como tipos de variáveis, operações com variáveis, comando condicional e laços de repetição.

1.2.2. Objetivos Específicos

- I. Realizar uma pesquisa sobre as dificuldades no ensino de programação.
- II. Identificar as vantagens e desvantagens da utilização das ferramentas no ensino de programação.
- III. Identificar as vantagens e desvantagens da utilização de jogos eletrônicos no ensino de programação.
- IV. Elaborar um documento de *game design* do jogo.
- V. Desenvolver o protótipo do jogo Imersão Total.

1.3. METODOLOGIA

O presente trabalho iniciou-se com uma pesquisa bibliográfica referente às dificuldades no ensino-aprendizagem de programação. Depois, partiu-se para o levantamento bibliográfico das ferramentas e jogos que buscam sanar tais dificuldades. Em seguida, pesquisaram-se características do RPG para o ensino. Então, a partir da análise que foi realizada a partir das dificuldades apresentadas pelos autores encontrados na literatura.

E a partir das ferramentas e dos jogos educativos disponíveis foi analisado como eram criadas e apresentadas os desafios aos jogadores para a aprendizagem de programação, para que assim fosse idealizado um jogo de RPG educativo para apoiar a aprendizagem de programação Java, os quais foram registrados em um documento de *game design* que é uma metodologia utilizada para o desenvolvimento de jogos que será utilizado como um roteiro na criação do jogo Imersão Total. Após finalizar o protótipo do jogo ele foi aplicado com 7 alunos para que eles avaliassem o jogo através de um questionário.

1.4. ESTRUTURA DO TRABALHO

Este trabalho encontra-se organizado da seguinte forma:

O capítulo 2 aborda a questão da fundamentação teórica sendo apresentadas as dificuldades no ensino de programação, as ferramentas e jogos que buscam resolver essas dificuldades e as características dos jogos de RPG educativo.

No capítulo 3 são expostas as ferramentas para o desenvolvimento e as características educativas do jogo proposto Imersão Total. Já no capítulo 4 são mostradas as fases do jogo, fazendo-se referência ao documento de game design juntamente com as telas do jogo. Por fim, no capítulo 5, é apresentada a conclusão do presente trabalho.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, será apresentado a fundamentação teórica do trabalho sobre as dificuldades no ensino-aprendizagem de programação, ferramentas e jogos utilizados para diminuir tais dificuldades, dando destaque aos jogos de *Role playing game* (RPG). Sendo apresentados trabalhos relacionados ao apoio-aprendizagem de Java.

2.1. Ensino-aprendizagem de Programação

Segundo Borges (2000), o modo de ensino tradicional não consegue facilmente motivar os alunos a se interessar pela disciplina de programação, uma vez que o conteúdo é apresentado de forma expositiva, com alguns exemplos, e depois foca-se na solução de problemas com listas de exercícios.

Essas listas muitas vezes usam de forma exagerada a repetição de problemas semelhantes com enunciados textuais diferentes, tornando o processo de ensino e aprendizagem monótono e cansativo (RAPKIEWICZ et. al, 2006). Essa forma de abordar o conteúdo de programação não contribui para motivar os alunos.

Outro problema que Rapkiewicz et. al, (2006) destacam é a correção do algoritmo feito pelo aluno, em que muitas vezes é utilizado o teste de mesa, que consiste em o aluno acompanhar o código passo-a-passo para verificar se o mesmo está correto. Essa é uma atividade desmotivante, e que ainda introduz novos problemas, uma vez que o aluno tem dificuldades em realizar a tarefa.

2.2. Ferramentas para o Ensino-aprendizagem de Programação

Diante das dificuldades apresentadas, foram empregados esforços para desenvolver ferramentas de apoio ao ensino de programação buscando reduzir os desafios expostos. Tem-se várias abordagens para a elaboração de ferramentas como os interpretadores de pseudocódigo, aplicativos para a geração e interpretação de fluxogramas, animações de algoritmos e ferramentas de desenvolvimento como o Eclipse.

Barbosa e Fernandes (2011) classificam as ferramentas em quatro categorias diferentes, que são:

- **Ferramentas de linguagem:** voltadas a alguma linguagem de programação específica, como Java, ou a uma nova linguagem criada com o propósito de melhorar o ensino de algoritmos, como uma linguagem em português.

Enquadram-se nesta categoria as ferramentas JavaTool (MOTA et. al, 2008) e JEliot (MORENO et. al, 2004).

- **Ferramentas de simulação:** são ferramentas que, de algum modo, simulam o comportamento de algoritmos, seja via depuração de código ou através de fluxogramas. Nessa categoria tem se o VisuAlg (SOUZA, 2009) que aborda a depuração do conteúdo das variáveis na execução.
- **Ferramentas de interação com algoritmos:** não necessariamente são ferramentas voltadas ao ensino de algoritmos, mas fazem constantes referências aos seus conceitos, ou seja, apresentam uma série de instruções que os alunos podem usar para alcançar um objetivo. Como, por exemplo, organizar os comandos necessários para fazer um robô andar como ocorre no Robomind (BENITTI et. al, 2009).
- **Jogos aplicados ao ensino de algoritmos:** são jogos que possuem geralmente, de forma indireta, conceitos de algoritmos incorporados na interação do jogador, ou seja, utilizam conceitos de programação, como variáveis e condições, que o jogador precisa usar para completar o desafio. Nesse estilo, tem se o Light-Bot (LIGHTBOT, 2016), que apresenta *puzzles* que o jogador tem que levar um boneco de um ponto para outro utilizando comandos como frente, pular, etc.

Mas ferramentas como VisuAlg (SOUZA, 2009) e Portugol IDE (MANSO et. al, 2004) que utilizam pseudocódigo apresentam algumas desvantagens, como; a falta de padronização dada a sua natureza não estruturada, é um tanto difícil de padronizá-lo, logo, um programador pode não entender a lógica de um programa escrito por outra pessoa (HALSTEAD, 2007).

Tem-se ainda a programação visual como Scratch (MÉLO et. al, 2011) e iVProg (KAMIYA e BRANDÃO, 2009) que minimizam os problemas relacionados à sintaxe, uma vez que permitem ao aluno arrastar os componentes existentes para a área de codificação, sem a necessidade de digitação dos comandos no código e impedindo a colocação de componentes em locais errados, evitando vários erros comuns de sintaxe.

2.2.1. Ferramentas que auxiliam o ensino de Java

Ferramentas como o JEliot (MORENO et. al, 2004) e o JavaTool (MOTA et. al, 2008), facilitam o entendimento das listas de exercícios, pois apresentam várias formas de acompanhamento do programa, ajudando no raciocínio lógico. Permitem a execução do código passo-a-passo com animações, levando o aluno a entender como funciona a execução no computador e como é feita a sequência de passos, ajudando-o no entendimento de um teste de mesa.

2.3. Jogos Educativos para o Ensino-aprendizagem de Programação

Vários autores defendem o uso de jogos educativos para o ensino-aprendizagem (KISHIMOTO 2003; PASSERINO 1998; ALBUQUERQUE e ANTONIO 2009). Kishimoto (2003) destaca que o jogo educativo possui duas funções: a lúdica, por proporcionar diversão, e a educativa, pois através dela podem-se ensinar vários assuntos aos alunos. Deve-se buscar manter essas duas funções em equilíbrio para que assim o jogo não perca suas características lúdicas e educativas.

Os jogos educativos além de auxiliarem os alunos, apresentam um aprendizado eficiente, já que a memorização e a compreensão acontecem porque há uma necessidade prática no jogo que exige certo assunto, como um desafio que será mais facilmente superado depois de certo aprendizado (ALBUQUERQUE e ANTONIO 2009).

Entretanto, a utilização de jogos educativos implica em vantagens e desvantagens que são destacadas por Grando (2001), conforme apresentado na Tabela 1.

Tabela 1 Vantagens e desvantagens dos jogos (Grando, 2001)

Vantagens	Desvantagens
<ul style="list-style-type: none"> • Fixação de conceitos já aprendidos de uma forma motivadora para o aluno; • Introdução e desenvolvimento de conceitos de difícil compreensão; • Desenvolvimento de estratégias de resolução de problemas (desafios dos jogos); • Aprender a tomar decisões e saber avaliá-las; • Significação para conceitos aparentemente incompreensíveis; • Propicia o relacionamento de diferentes disciplinas (interdisciplinaridade); • O jogo requer a participação ativa do aluno na construção do seu próprio conhecimento; • A utilização dos jogos é um fator de motivação para os alunos; • Favorece o desenvolvimento da criatividade, de senso crítico, da participação, da competição “sadia”, da observação, das várias formas de uso da linguagem e do resgate do prazer em aprender; • As atividades com jogos podem ser utilizadas para reforçar ou recuperar habilidades de que os alunos necessitem. Útil no trabalho com alunos de diferentes níveis. 	<ul style="list-style-type: none"> • Quando os jogos são mal utilizados, existe o perigo de dar ao jogo um caráter puramente aleatório, tornando-se um “apêndice” em sala de aula. Os alunos jogam e se sentem motivados apenas pelo jogo, sem saber por que jogam; • O tempo gasto com as atividades de jogo em sala de aula é maior e, se o professor não estiver preparado, pode existir um sacrifício de outros conteúdos pela falta de tempo; • As falsas concepções de que devem ensinar todos os conceitos através dos jogos. Então, as aulas, em geral, transformam-se em verdadeiros cassinos, também sem sentido algum para o aluno; • A perda de “ludicidade” do jogo pela interferência constante do professor, destruindo a essência do jogo; • A coerção do professor, exigindo que o aluno jogue, mesmo que ele não queira, destruindo a voluntariedade pertencente à natureza do jogo; • A dificuldade de acesso e disponibilidade de materiais e recursos sobre o uso de jogos no ensino, que possam vir a subsidiar o trabalho docente.

Outro aspecto é em relação ao *feedback*, um elemento essencial em qualquer jogo educativo que, segundo Maike et. al, (2011), trata-se da maneira como o jogo responde aos

erros do jogador e um bom *feedback* deve levar o jogador a entender por que errou, ou direcioná-lo para o caminho correto, em vez de simplesmente revelar qual é a resposta certa.

Essa falha ocorre em muitos jogos educativos. Por exemplo, o jogo Vila Python (OLIVEIRA, 2015) apenas apresenta para o aluno uma pergunta, como desafio, e depois mostra um conjunto de possíveis respostas, cabendo ao jogador escolher a correta entre elas, não fornecendo mais nenhuma informação do problema, e deixando o jogo repetitivo, pois está utilizando apenas perguntas para apresentar o conteúdo ao jogador.

Esse tipo de abordagem não deixa de ser válida para verificar se o aluno tem conhecimento do assunto, mas apenas utilizar esse elemento limita o jogo, e o *feedback* para o aluno é mínimo. Entretanto, poderia ser apresentado algo além da pergunta, como uma explicação sobre o porquê da resposta, sendo essa explicação inserida no contexto do jogo.

O Takkou (BARBOSA e FERNANDES, 2011), o Robozzle (ROBOZZLE, 2016) e o Light-Bot (LIGHTBOT, 2016) são jogos lúdicos, voltados à prática de algoritmos para alunos iniciantes. Neles o aluno tem que usar uma lista de instruções, que quando for executada realizará uma tarefa representada por um *puzzle*.

Porém, os jogos não apresentam uma contextualização para seus objetivos, pois, segundo Clua e Bittencourt (2004), para um jogo ter um forte poder educativo, os desafios e as histórias bem contadas são fatores críticos de sucesso. Inclusive, o Light-Bot (LIGHTBOT, 2016) não utiliza uma linguagem textual, apenas imagens que representam as instruções.

Além dessas desvantagens citadas, Bittencourt e Giraffa (2003) informam que a maioria dos jogos educativos que propõe o auxílio no ensino de programação consistem em uma coleção de enigmas desconexos entre si, não criando uma sensação de imersão, e tratando o jogador como um estudante, o que torna o jogo desinteressante e pode provocar desistências.

2.4. Jogos de RPG Educativo

Os Role Playing Games (RPGs) podem ser encontrados em dois formatos diferentes: o de mesa e o digital. No RPG de mesa, existe o mestre que narra e controla as ações dos NPC's (personagens controlados pelo mestre ou personagens não jogáveis), como os inimigos ou outros personagens que o jogador encontra também descrevendo o local em que ele está. Já no

RPG digital, as ações são realizadas pelo computador, o que pode limitar as ações do jogador, já que não é possível programar todas as ações possíveis.

O presente trabalho propõe um RPG desenvolvido em formato digital. Dentre as vantagens no uso do RPG digital em relação ao de mesa estão o melhor visual gráfico, a possibilidade de maior interação entre o jogador e o jogo, e a utilização de recursos como imagens, animações e sons que permitem uma maior imersão do jogador. E claro a possibilidade de jogar sozinho que não é possível em um RPG de mesa.

O objetivo de um jogo de RPG é explicado por Bittencourt e Giraffa (2003):

[...] representar papéis em um mundo imaginário, construindo uma estória coletivamente em torno de uma mesa com um grupo de pessoas. Nesta modalidade digital, o RPG continua sendo esta representação de papéis, este jogo de faz-de-conta e vivenciar mundos imaginários [...].

É interessante acrescentar nos jogos educativos princípios narrativos, e dentre os estilos de jogo mais comuns (ação, tiro, estratégia e o de plataforma), o RPG tem como foco principal a narrativa que instiga o jogador-aluno a participar do jogo, mascarando o conteúdo educacional e tornando-o transparente (MAIKE *et. al*, 2011). A estória é contada através das *quests* (missões) que o jogador deve realizar, tendo sempre associada uma recompensa que o estimule a explorar o mundo do jogo.

2.5. Trabalhos Relacionados

Entre os jogos para o apoio e aprendizagem de Java, tem-se o Robocode (O'KELLY, e GIBSON, 2006) e o Furtbot (VAHLDICK et. al, 2008), que consistem em programar um robô para competir contra outros em uma batalha. O jogador cria sua estratégia, podendo usar princípios de inteligência artificial para o comportamento de seu robô.

O jogo apresenta um conceito motivador para o aluno, pois o mesmo está controlando seu próprio robô no jogo. Entretanto, esse jogo não é apropriado para alunos iniciantes, uma vez que requer um conhecimento prévio do assunto, sendo mais indicado para alunos com mais experiência.

Tem-se ainda o Code Hunt (CODEHUNT, 2016) que consiste em uma coleção de desafios em que o jogador tem que corrigir o erro do código, feito em Java, para que passe de nível. É apresentado ao jogador um código, uma lista de entrada, a saída do programa e o que

era esperado na saída do programa. A partir desses dados, o jogador tem que corrigir o jogo para que a saída seja igual ao que era esperado e assim ir para níveis mais difíceis.

Fica evidente que o jogo Code Hunt (CODEHUNT, 2016) não apresenta nenhum elemento motivacional, como um enredo ou variações de desafios, para o jogador continuar utilizando, e não passa de uma coleção de exercícios com testes predefinidos.

3. DESENVOLVIMENTO DO JOGO IMERSÃO TOTAL

Este capítulo consiste no detalhamento do desenvolvimento tais como as suas características educativas e motivacionais do jogo.

3.1. Ferramentas e Materiais Utilizados no Desenvolvimento

Para desenvolver o protótipo do jogo, foi utilizada a ferramenta RPG Maker Vx Ace por permitir a criação de jogos de RPG e possuir uma base de dados completa para a produção de jogos como imagens de cenários, personagens, sons e animações, além de possuir uma comunidade de participantes ativa.

A ferramenta RPG Maker Vx Ace foi utilizada por apresentar características como:

- Um desenvolvimento orientado a eventos: a cada elemento da tela pode-se associar um conjunto de eventos;
- Disponível em português: a ferramenta encontra-se toda em português do Brasil;
- Acesso ao código fonte: permite acesso e modificação do código utilizado para o desenvolvimento do jogo;
- Materiais gratuitos: possui uma comunidade ativa que disponibiliza gratuitamente vários recursos para o desenvolvimento.

No desenvolvimento do jogo, foi usado o script *XAS Hero Edition ACE*¹, criado por Xiderowng (MogHunter). Esse script é um sistema complexo, que permite o desenvolvimento de lutas em tempo real além de uma HUD (*Heads-Up Display*), que se encontra na Figura 1, apresentando as informações do estado do jogador como seu CT (*Combat Time*), LV (nível), HP (pontos de vida), SP (pontos mágicos), item, Skill (habilidade do jogador) e quantidade de habilidades, Wep (arma equipada do jogador), Shd (escudo equipado do jogador), e quantidade de moedas que o jogador possui.



Figura 1 HUD do jogo Imersão Total

¹ XAS hero Edition ACE: <https://atelierrgss.wordpress.com/rgss3-xiderowng-action-system-xas/>

Também foi utilizado o script *Basic Mouse System*², que permite usar o *mouse* no jogo e, por fim, o *script* para pular a tela de título³ em que permitiu a criação do menu principal de forma personalizada. Todos os *scripts* usados suprem limitações do RPG Maker Vx Ace e permitem uma maior personalização na criação do jogo. Por fim, foi empregado o editor de imagem Gimp para a confecção de algumas imagens como as da tela de título e a da seleção de personagem.

Foram comprados na *Steam* (STEAM, 2014), dois pacotes de imagens de personagens (o pacote *Dark Hero Character* e o *Fantasy Hero Character*), e um pacote de áudio (o *The Adventurer's Journey III*), e a utilização de outros pacotes de áudio⁴.

3.2. O Desenvolvimento do Jogo

O jogo Imersão Total foi desenvolvido utilizando-se o RPG Maker Vx Ace da *Enterbrain* (ENTERBRAIN, 2015) que é um aplicativo que permite a criação de jogos para a plataforma *Windows* sem que seja necessário codificar em alguma linguagem de programação propriamente dita.

O jogo foi dividido em três fases, conforme os assuntos abordados no ensino introdutório de Java. Na Cidade Inicial (primeira fase), são abordados os conteúdos de tipos e operações com variáveis. A Cidade Condisional (segunda fase) trabalha a estrutura condicional, e a terceira fase (Masmorra), aborda o laço de repetição. O jogador pode voltar para fases já finalizadas, não tendo que refazer os desafios, mas tendo que lutar novamente com os inimigos que estejam na área.

3.3. Características Educativas e Motivacionais

O jogo utiliza elementos como a possibilidade do jogador escolher o sexo e o nome do personagem principal, oferece recompensas para estimular o jogador a continuar, como habilidades novas (Ex: uso de bolas de fogo, corte congelante, bola de fogo gigante), e usa elementos do cotidiano, como um sistema de dia e noite e clima, para desafiar o jogador.

Cada fase do RPG contém músicas em seu background, para que seja mais agradável a permanência no jogo até o final. Foi utilizado um sistema de batalha ativa em que o jogador

² Basic mouse system: <http://daimoniuoustails.weebly.com/basic-mouse-system.html>

³ Pular tela de título: <http://centrorpg.com/index.php?topic=628.0>

⁴ Arquivo de áudio: <http://forums.rpgmakerweb.com/index.php?/topic/17312-free-darrens-commercially-free-music/>

pode lutar corpo-a-corpo com seus inimigos, podendo usar suas habilidades para ataques especiais. O jogador é apresentado a uma diversidade de cenários como cidades, florestas e a masmorra, com cada área dando ao jogador uma variedade de ambientes para explorar.

O jogo tem como elementos diferenciais as explicações contextualizadas, uma variedade de estilos de desafios como questões de múltipla escolha, e os desafios em que o jogador realizará uma tarefa proposta pela interpretação de um código. Com as explicações usadas como parte do tratamento do erro do jogador nos desafios, em vez de apenas penalizar o jogador, o erro será o gatilho para inserir as explicações, podendo ser usadas de três formas dentro do enredo do jogo:

- Erro por parte do jogador: por exemplo, quando o jogador tiver que trazer um item do tipo inteiro, mas pegar um do tipo *float*, seus companheiros vão explicar por que esse item não é o correto.
- Dúvida apresentada pelo personagem do jogador: diante de um acontecimento, o personagem do jogador pode ficar com uma dúvida e perguntar aos seus amigos para que eles expliquem a ele. Por exemplo, no desafio que trata de uma pergunta à explicação, foi inserido um comentário no final, utilizando-se a justificativa de que o personagem do jogador não entendeu o que ocorreu.
- O personagem do jogador é indagado se entendeu: cabe ao jogador aceitar ou não a explicação, pois como se trata de um RPG o jogador tem certa liberdade. Nesse caso, a explicação pode ser inserida no meio ou no final de um desafio que consiste de uma interpretação do código.

Serão utilizadas também analogias com os acontecimentos do jogo para apresentar conceitos de programação como laço de repetição e condicional. Por exemplo, tem se um monstro que se não for derrotado em um tempo limite, retorna o jogador para um ponto no mapa, ou seja, o jogador fica em um *loop* se não derrotar o monstro. Através de diálogos dos personagens, será apresentado o conceito de laço de repetição e o famoso erro de lógica do *loop* infinito.

4. IMERSÃO TOTAL

Neste capítulo serão apresentadas as fases do jogo sendo mostrada uma descrição geral do jogo, seguida pela descrição de cada fase com seu enredo, desafios e quais assuntos são abordados em cada uma. E por fim a avaliação dos usuários sobre o jogo Imersão Total.

4.1. Descrição geral do jogo Imersão Total

As informações gerais do jogo estão contidas no documento de Game Design (apêndice A). Para uma visão geral do jogo, tem se a Tabela 2 que apresenta o escopo do jogo.

Tabela 2 Escopo do jogo Imersão Total

Título	Imersão Total
Plataforma	PC-(Windows)
Jogadores	Apenas um jogador
Gênero	Role-Playing Game (RPG)
High Concept	O personagem principal do jogo está participando da olimpíada de programação de 2116, que utiliza a mais nova tecnologia de imersão para os games permitindo que os usuários entrem em mundos virtuais com suas mentes. Mas o jogo acaba sendo invadido por um hacker que prende os competidores dentro do jogo. E agora o jogador precisa encontrar o Hacker e derrotá-lo para libertar todos os seus companheiros dessa prisão virtual que o game se tornou.
Objetivo	O objetivo do game é que o jogador utilize seus conhecimentos em Java para superar os desafios e derrotar o hacker que prendeu os competidores dentro do jogo.
Recursos	Mundo do game – É um mundo virtual que se passa na idade média com cidades, florestas e masmorras, com habitantes e monstros. Jogador – É interpretado por um avatar que explora o mundo do jogo, enfrentando monstros e realizando as <i>quest</i> (desafios). Ao realizar as <i>quests</i> , o jogador ganha itens que poderão ajudá-lo no decorrer do jogo. O jogador poderá salvar o status do jogo apenas quando possuir certo item (a barraca) ou se estiver na hospedaria (INN) e pagar por uma noite.

O enredo do jogo conta a história de uma olimpíada de programação que busca encontrar o melhor programador, mas a competição é atacada, e o jogador tem que descobrir uma forma de sair do jogo. Ele resolve se unir a outros dois competidores, que juntos irão buscar uma forma de encontrar e derrotar o hacker. Para isso, será preciso entender as pistas e resolver os desafios deixados pelo hacker, pois ele acha que é o melhor programador e que os competidores jamais conseguirão resolver seus desafios.

4.1.1. Descrição da Primeira Fase

A introdução do jogo consiste de um dialogo em que o jogador fala sobre está preso em um jogo e a formação da sua equipe para buscar uma forma de escaparem apresentando o objetivo principal do jogo. A Figura 2 ilustra uma tela onde o personagem principal conversa com seus colegas, e esses afirmam que vão encontrar uma forma de sair do jogo.



Figura 2 Introdução do jogo

Nesse momento, o jogador pode explorar a Cidade Inicial, tendo dois desafios que podem ser realizados para sair da cidade. Um consiste em uma pergunta, mas para respondê-la o jogador precisa ter 30 moedas no mínimo, o que leva ao segundo desafio, que é abrir um baú em outro ponto do cenário.

Logo depois de conseguir abrir o baú, o hacker aparece para os competidores, mas deixar uma pista para encontrá-lo que se trata de uma caixa, mas que para conseguir abri-la é preciso da chave correta, uma chave do tipo inteira como mostra a Figura 3 que apresenta a caixa e logo a cima as possíveis chaves que podem abri-las que representam variáveis do tipo inteira, *string* e *float*. Se o jogador escolher a chave errada não vai ter nenhuma penalidade só uma explicação de porque aquela chave não é a correta.



Figura 3 Desafio das chaves

Em posse do item que estava dentro da caixa, o jogador descobrirá que precisa ir para a cidade Condisional, e para isso terá que responder a pergunta do guerreiro que guarda a saída. Ao sair da cidade, o personagem principal apresentará uma dúvida sobre a resposta do desafio, momento em que seus companheiros explicam, tirando assim a possível dúvida do jogador.

Uma vez fora da cidade, o jogador enfrentará vários monstros, para chegar à próxima cidade. No caminho, será abordado por um senhor que precisa de ajuda, cabendo ao jogador aceitar ou não o pedido. O desafio é uma busca por cogumelos que, para saber a quantidade certa, o jogador tem que interpretar o código que é apresentado na Figura 4.



Figura 4 Desafio dos cogumelos

Finaliza-se, assim, a primeira fase do jogo, mas a cidade ainda pode ser acessada pelo jogador se assim ele desejar, para comprar item, armas ou passar uma noite na estalagem para salvar o jogo. Os conteúdos trabalhados nessa fase são: tipo de variáveis inteira *float*, *string* e *boolean*; e avaliação de uma operação com variáveis inteiras.

Os conteúdos foram apresentados de duas formas: uma que consiste na busca do item correto, ou seja, encontrar o tipo correto de variável que o item representa, e a segunda para responder uma pergunta sobre operações com variáveis do tipo inteiro. Na Tabela 3, tem-se a descrição da primeira fase, apresentando o conteúdo educacional, o contexto do desafio, como o desafio é representado no jogo, e o código que será utilizado.

Tabela 3 Fase 01: Cidade Inicial

Conteúdo educacional abordado	Contexto do jogo	Como é apresentado o desafio no jogo	Código usado no desafio
Tipos de variáveis.	Desafio abrir caixa: O jogador tem que encontrar a caixa sobre a qual o <i>hacker</i> falou, mas para conseguir abrir a caixa será preciso usar a chave correta, ou seja, do tipo indicado	Os tipos de variável são as chaves, e uma delas pode abrir a caixa, mas para isso ela tem que ser do tipo indicado.	chave = 3; chave = 4.6; chave = "3";
Tipo de variável boolean.	Desafio abrir baú: Para conseguir abrir o baú, será necessário adicionar nele o valor verdadeiro, um True.	O jogador tem que encontrar um item True e usá-lo para abrir o baú.	true e false
Operações com variáveis do tipo int e diferenças em relação ao tipo float, reforçando o que foi apresentado no desafio abrir baú.	Desafio saída da cidade Inicial: Para sair da cidade, será preciso responder uma pergunta.	Uma pergunta de múltipla escolha com explicações ao final do desafio em um diálogo entre os personagens do jogo.	Qual é o valor exibido na tela? int valor = 1-25/2; System.out.println(valor); Escolhas: -11, 11, -11.5 e 11.5
Interpretação de um código para saber o valor final da variável cogumelo. Em que é abordado operações com variáveis.	Desafio busca por cogumelos: Encontrar 8 cogumelos para que o senhor possa fazer um remédio para a sua esposa.	Teste de mesa e diálogos sobre as características do código, dando ao jogador pistas sobre o resultado.	Int cogumelos = 4; cogumelos += 4;

4.1.2. Descrição da Segunda Fase

Seguindo para a cidade, o personagem principal encontrará uma bifurcação, que pode ser vista na Figura 5, mas para seguir um dos caminhos, é preciso ter um item. Um caminho está fechado por pedras, e para passar ele tem que explodir as rochas. Já o outro caminho passa por um rio, e para isso ele precisa de um gancho para atravessar. Diante disso, os seus companheiros apresentam o conceito de estrutura condicional.

Os itens para transpor os obstáculos podem ser comprados na Cidade Inicial (Fase 1), ou encontrados ao explorar a floresta.



Figura 5 Desafio dos caminhos

Ao passar, por esse desafio o jogador tem que chegar à entrada da Cidade Condisional, mas, para ter acesso à cidade, ele precisa encontrar a combinação certa das alavancas, que está representada na Figura 6, e para isso tem que compreender o código que está na placa de entrada da cidade. Já para sair da Cidade Condisional, o jogador tem que responder uma pergunta que é um teste de mesa que apresenta a diferença entre usar if e else if, que pode ser vista na Figura 7. Conteúdos trabalhados: estrutura condicional if, else if e else; diferença entre if e else if e estrutura de if dentro de if.



Figura 6 Desafio abrir o portão

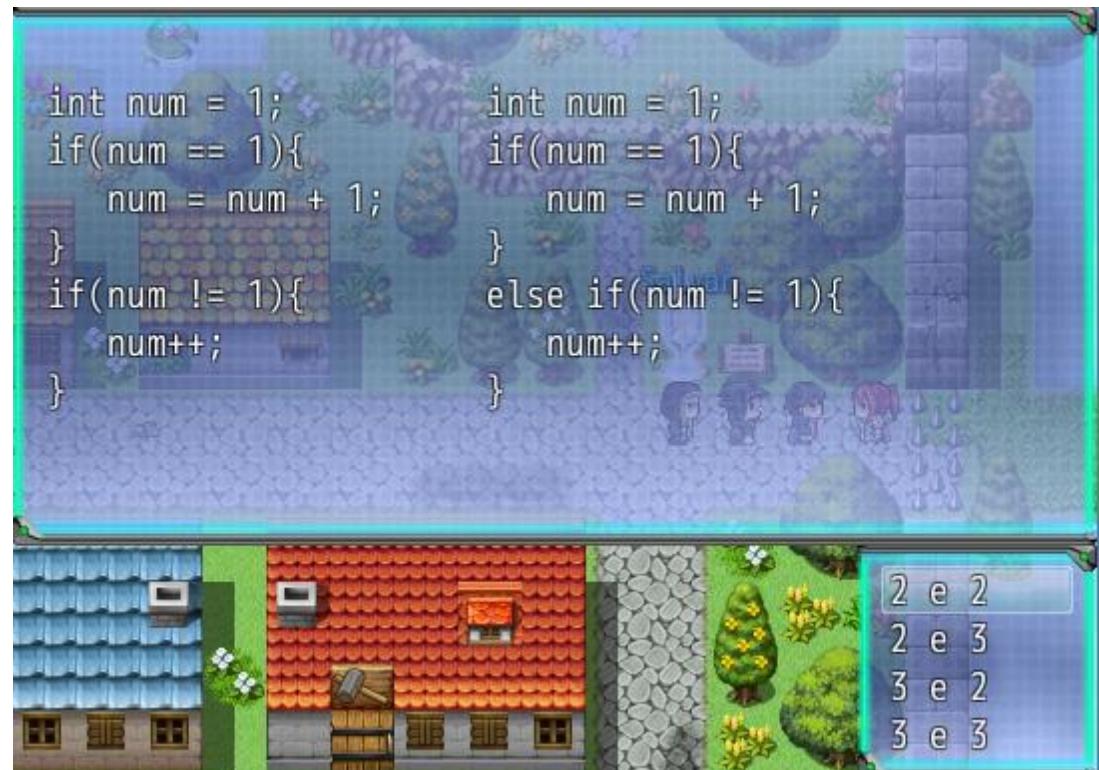


Figura 7 Desafio sair da cidade Condisional

Os conteúdos foram trabalhados de duas formas: uma que consistia na apresentação do conceito da estrutura condicional utilizando analogias, e a segunda através de questões problemas para o jogador responder. Na Tabela 4 tem se os desafios que o jogador terá que

superar para conseguir chegar à próxima cidade, e posteriormente sair dela para ir à masmorra.

Tabela 4 Fase 02: Cidade Condisional

Conteúdo educacional abordado	Contexto do jogo	Como é apresentado o desafio no jogo	Código usado no desafio
Nenhum.	Derrotar os monstros.	Monstros que atacam o jogador.	
Estrutura condicional if. Apresentando a estrutura de if dentro de um if.	Desafio entrada da cidade Condisional: Para entrar na cidade é preciso abrir o portão que está fechado para os monstros não invadirem.	Ao tentar abrir o portão é apresentado uma mensagem e é preciso entender o código para manipular as alavancas para deixar igual ao que é apresentado no código.	if(alavancaVermelha.equals("Cima")){ if(alavancaVerde.equals("Baixo")){ if(alavancaAzul.equals("Cima")){ abrirPortão();
Estrutura condicional e a diferença na utilização de if e else if .	Desafio sair da cidade Condisional: Para sair da cidade é preciso responder a uma pergunta da guarda.	Uma pergunta de múltipla escolhas sobre a comparação entre o resultado final da variável num em dois códigos diferentes.	Qual é o valor da variável num no final de cada código? <pre>int num = 1; int num = 1; if(num == 1){ if(num == 1){ num= num + 1; num= num + 1; } } if(num != 1){ else if(num != 1){ num= num + 1; num= num + 1; } }</pre>

4.1.3. Descrição da Terceira Fase

No caminho para a masmorra, o jogador se depara com mais uma armadilha do hacker: um monstro com uma habilidade peculiar que consiste de enviar o jogador para um ponto do mapa, deixando-o em um *loop*, pois esse monstro está no único caminho para a masmorra, e se não for derrotado em um intervalo de tempo ele manterá o jogador preso nessa área. Na Figura 8 é apresentado o encontro do jogador com o monstro.



Figura 8 Desafio monstro do loop

Através desse fenômeno é apresentado o conceito de laço de repetição juntamente com os diálogos dos companheiros sobre o ocorrido. Ao sair do *loop* o jogador chegará à entrada da masmorra. E para acessar a área mais profunda é preciso usar uns pesos para abrir uma passagem e para isso o jogador tem que buscar três sacos para usar como pesos e abrir a passagem, para apresentar o conceito do laço de repetição FOR.

O jogador encontra o hacker, mas ele acaba fugindo para outro plano no jogo e para chegar lá seus companheiros utilizam o acesso emergencial que o local possibilita, mas para conseguir isso eles têm seus avatares desativados temporariamente, cabendo ao jogador derrotar o hacker sozinho. Os conteúdos trabalhados na terceira fase são: laço de repetição; diferença entre while e for.

Os desafios encontram-se na Tabela 5, e percebe-se que não tem a coluna “código usado no desafio”, pois nessa parte os desafios são analogias, que apresentam os conceitos do laço de repetição através dos diálogos dos personagens e situações que são apresentados ao jogador.

Tabela 5 Fase 03: Masmorra

Conteúdo educacional abordado	Contexto do jogo	Como é apresentado o desafio no jogo
Conceito de <i>loop</i> infinito na utilização de laços de repetição e apresentando o conceito do while.	Desafio fantasma em loop: O jogador é transportado para a saída da cidade Condicional se não derrotar um fantasma.	O jogador ficar em um <i>loop</i> só podendo sair dele se derrotar um fantasma. Fazendo uma analogia com o conceito de laço de repetição que é explicado em um diálogo dos personagens.
Laço de repetição for.	Desafio busca por pesos: Para conseguir acessar uma área da masmorra é preciso destravar uma porta, mas para isso terá que colocar uma quantidade de sacos em um local para que o peso libere a trava abrindo a porta.	É apresentando uma situação problema, em que a sua solução representa o conceito do laço de repetição for. E através dessa representação seus companheiros abordarão o assunto no jogo.
Nenhum.	Derrotar os monstros.	Monstros que atacam o jogador.

4.2. Avaliação do Jogo Imersão Total

Para avaliar o jogo, foram convidados 7 alunos do curso de Licenciatura em Ciência da Computação que já possuíam conhecimento da linguagem Java, não tendo um foco em uma turma específica sendo esses voluntários. O jogo Imersão Total foi disponibilizado no *Dropbox*⁵, para que os alunos-avaliadores fizessem o download do mesmo.

Foi utilizado um questionário contendo 12 questões, sendo que 10 eram formadas por uma escolha entre cinco números em que 1 representava o discordo fortemente, e o cinco, o concordo fortemente. As outras duas perguntas pediam para o jogador citar pontos fortes e apresentar sugestões de melhorias em relação ao jogo. O questionário completo utilizado na avaliação encontra-se no apêndice C.

Não foi apenas avaliado o conteúdo acadêmico, mas também características do gênero do jogo, pois é um ponto importante já que se o aluno não gosta do gênero RPG, sua experiência vai ser negativa com o jogo educativo. No gráfico 1 de interesse pelo gênero do jogo RPG, tem-se essa relação do gosto dos alunos em relação ao jogo.

⁵ Jogo Imersão Total: <https://www.dropbox.com/sh/0n947jubraqg6qq/AACk9U3iWvqfPcizBNz7spaJa?dl=0>

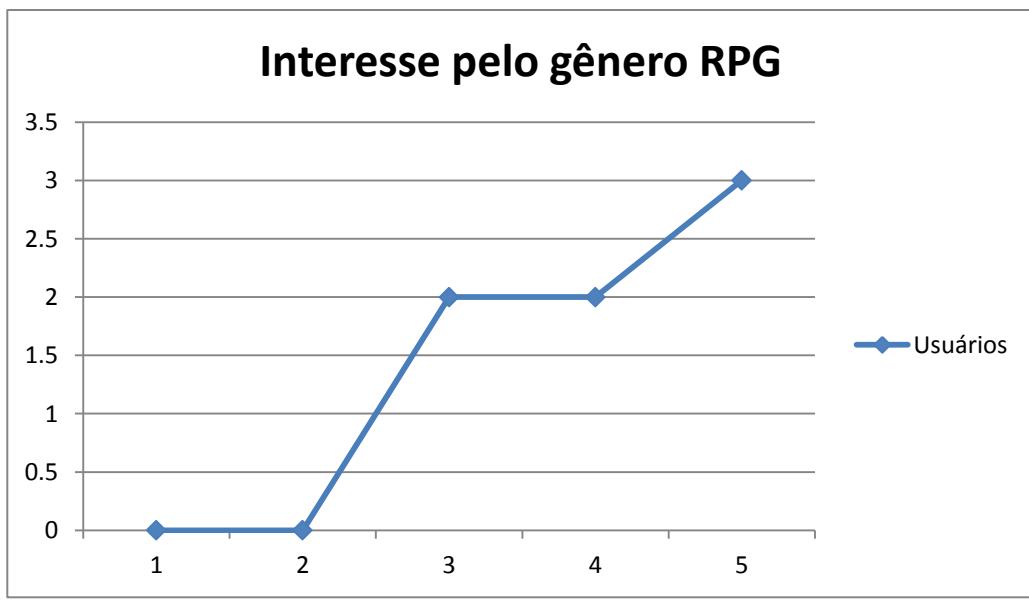


Gráfico 1 Interesse pelo gênero RPG. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

Tem-se indícios de que o jogo foi bem aceito, mas vale ressaltar que mesmo apresentando bons resultados, alunos que não gostem do gênero, não se sentirão motivados a continuar jogando como foi destacado por um dos alunos-avaliadores que destacou “*Um ponto negativo para o jogo estaria no desagrado por parte de um grupo de pessoas que não gostassem desse tipo de jogo*”. Pois o jogo de RPG demanda uma alta carga de paciência, tempo e motivação em continuar jogando.

Outro ponto que foi destacado pelos alunos-avaliadores foi que “*Jogos com uma dinâmica mais objetiva, e que demandem menor tempo de realização de conquistas poderiam promover de forma mais eficaz os propósitos de aprendizagem*”. Nesse ponto, é válido destacar que o jogo de RPG não necessariamente tem que ser finalizado de uma única vez, pois pode ser salvo e continuar-se jogando em outro momento.

Além que é possível adicionar mais elementos para ter um menor tempo na realização da conquista do jogador. Os avaliadores pediram mais desafios no jogo. Esse tipo de jogo dá mais liberdade ao jogador, fato que não ocorre em outros jogos educativos.

Dante do que foi apresentado percebe-se que o jogo de RPG tem suas limitações, mas seus pontos fortes suprem suas deficiências.

Em relação ao design e interesse que o jogo despertou nos alunos-avaliadores foram bem positivos, como foi expresso “*O que eu achei legal foram as músicas, e o sistema de dia e noite*” e “*O design ficou ótimo. Tanto a interface quanto os personagens ficaram muito bem feitos*” pode-se ver nos Gráficos de Design do jogo e Interesse inicial pelo jogo.



Gráfico 2 Sobre o design do jogo. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

Entretanto o jogo apresentou ponto negativo como foi destacado pelos alunos-avaliadores “*Os mapas às vezes estão tão cheios de plantas que você não consegue ver as saídas, aconselho limpar um pouco os cenários*”, mas essa dificuldade faz parte do contexto do jogo onde o jogador tem que explorar esse mundo que lhe é apresentado.

O interesse foi criado pelos elementos que os próprios alunos-avaliadores citaram como atrativos que são “*O jogo apresenta surpresas que fazem o jogo ficar mais atrativo*”, “*ambientação e enredo que podem capturar a atenção de jogadores amantes desse tipo de jogo*”, “*jogabilidade simples e fácil interação com o jogo*”, “*música*”, “*sistema de dia e noite e a inclusão de assuntos educacionais relacionados à computação*”.

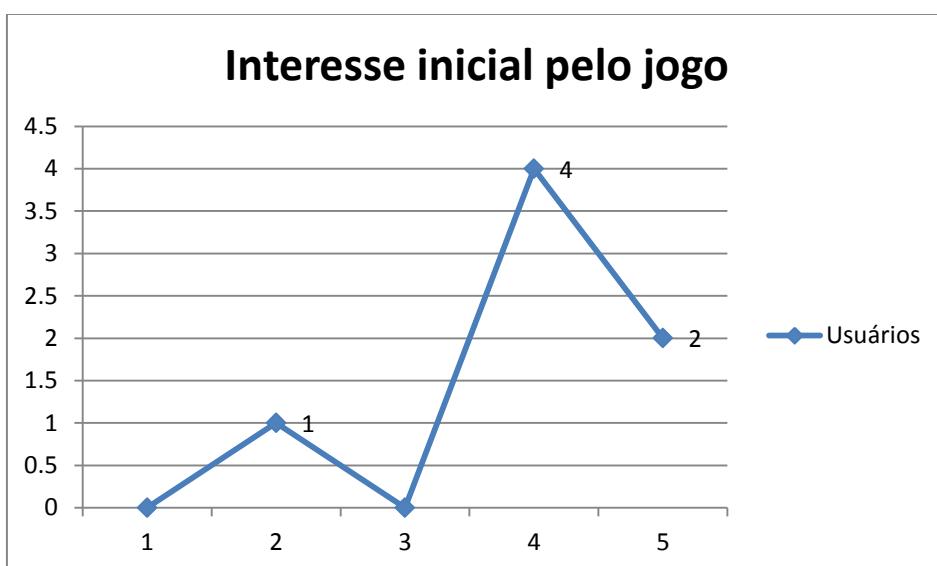


Gráfico 3 Sobre o interesse inicial pelo jogo. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

Outro elemento que contribuem para o interesse dos alunos é a variedade de desafios abordados no jogo como se vê nos gráficos de variação das atividades e evolução do ritmo do jogo para não criar uma atividade monótona.

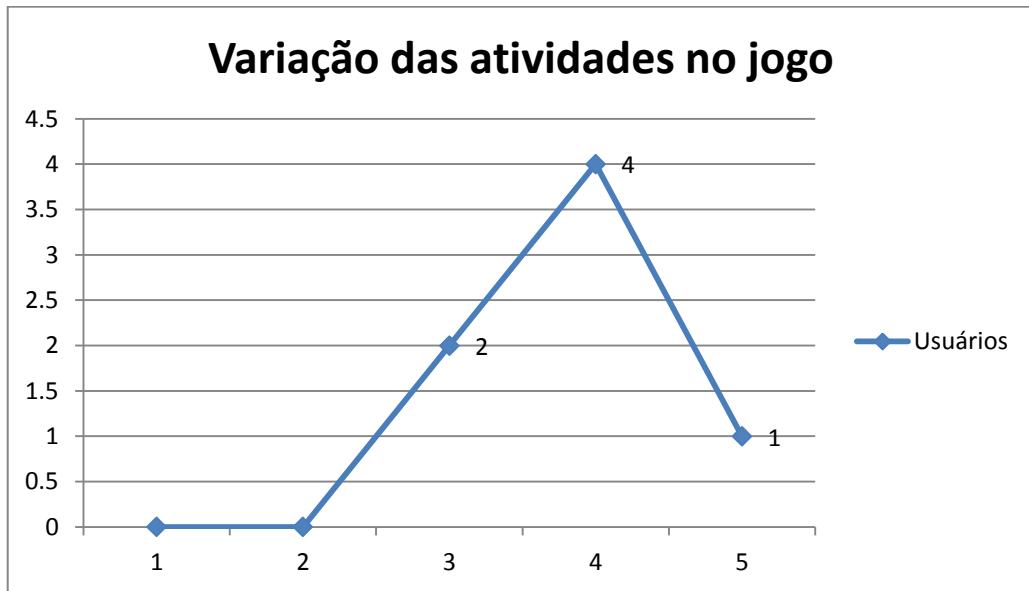


Gráfico 4 Sobre a variedade de desafios no jogo. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

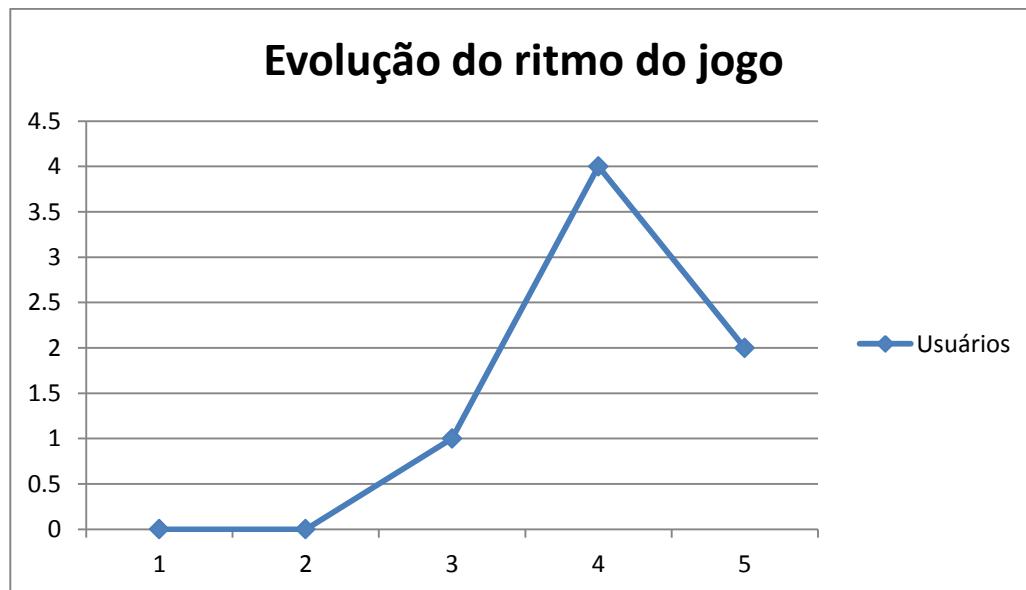


Gráfico 5 Sobre a evolução dos desafios no jogo. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

Ao verificar os gráficos tem se indícios que o jogo apresentou uma atividade imersiva e motivadora. Dessa forma um critério muito importante foi alcançar o equilíbrio entre o lúdico e o educativo. Isso se repetiu ao observar outros gráficos em que foi perguntado se eles se divertiram no jogo e se o recomendaria para seus colegas.



Gráfico 6 Sobre o divertimento na utilização do jogo. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

Foi exposto pelos alunos-avaliadores que o jogo seria recomendado para alunos iniciantes na computação como foi expresso “é uma forma de exercitar o conteúdo da disciplina e instiga-los a continuarem no curso e mostrando que eles podem fazer algo parecido”. Vale ressaltar que os alunos-avaliadores não sabiam qual era o público alvo do jogo.

Para avaliar os elementos educativos do jogo foram perguntadas aos alunos-avaliadores quais as contribuições do jogo para a disciplina e se a aprendizagem gerada na utilização do jogo foi eficiente. Tem-se esses valores nos gráficos de contribuição do jogo para a disciplina de programação e eficiência para a aprendizagem:

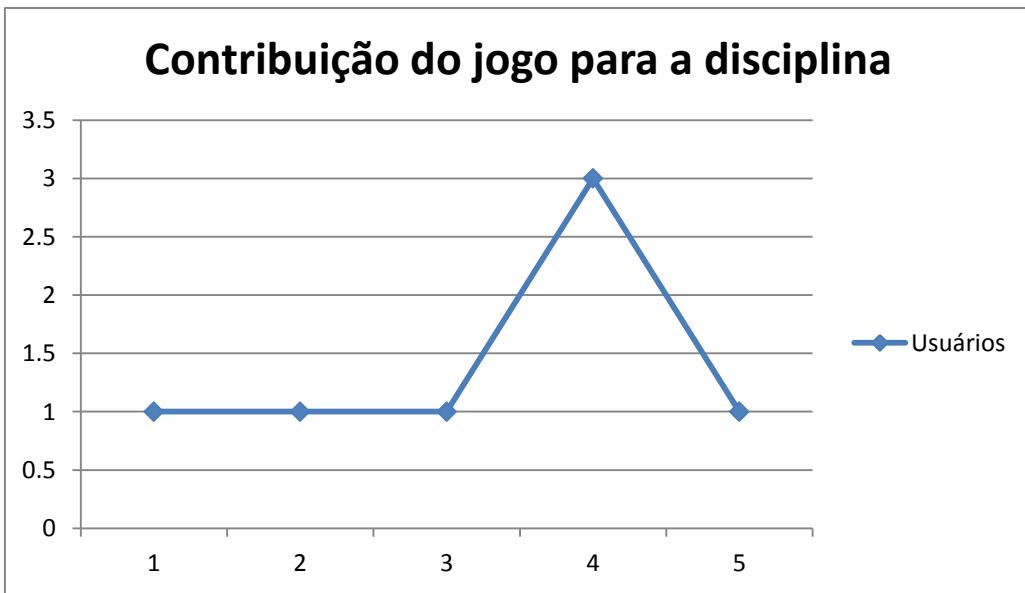


Gráfico 7 Sobre a contribuição do jogo para a disciplina de programação que utiliza-se Java. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

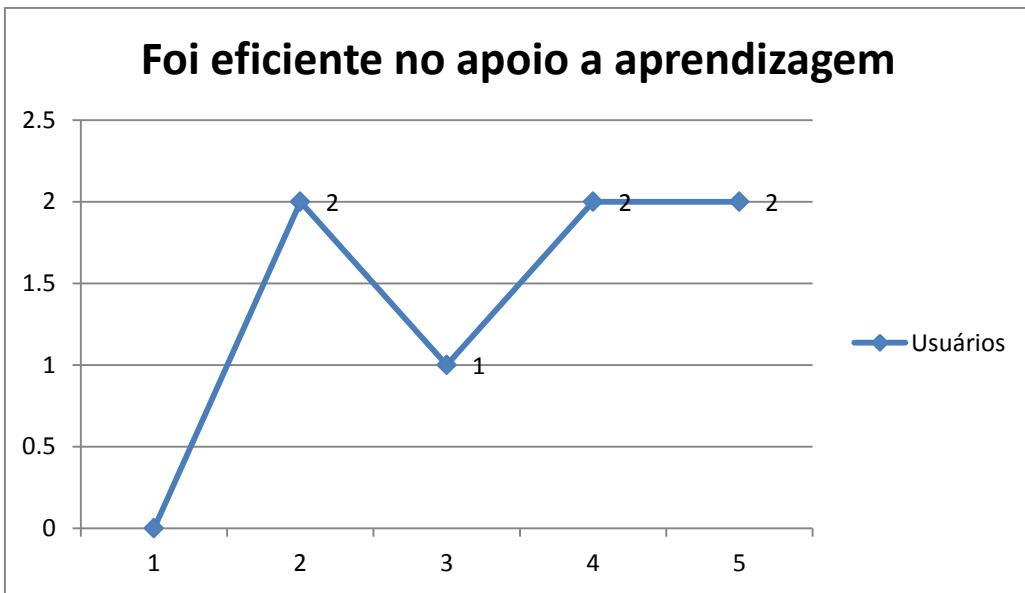


Gráfico 8 Sobre a eficiência do jogo em apoiar a aprendizagem de Java. Em que 1 representa discordo fortemente, 2 discordo, 3 neutro, 4 concordo e 5 concordo fortemente.

Na contribuição os alunos-avaliadores apresentaram indícios que o jogo contribuiria, mas que essa contribuição seria mais para alunos iniciantes como eles mesmos destacaram “*acredito que o jogo possa ajudar pessoas que não tenham tanta noção a respeito dos conceitos de programação*”, “*o jogo serve como um meio de exercitar o conteúdo aprendido*” e “*O jogo não contribui para a minha aprendizagem pelo fato de abordar questões pertinentes, mas direcionadas a estudantes de programação em níveis iniciais*”. Nessa ultima

frase vale ressaltar que o foco do jogo é alunos iniciantes então mesmo o jogo não tendo contribuído para o aluno ele mesmo afirmou que o jogo ajudaria alunos iniciantes.

Em relação à eficiência no apoio a aprendizagem teve-se uma tendência mais favorável à eficiência do jogo, diante dos comentários realizados pelos alunos-avaliadores que afirmaram que o jogo foi eficiente, como um afirmou “O jogo passa de forma divertida pontos específicos do conteúdo, o que torna a aprendizagem mais significativa, facilitando a absorção do conteúdo.”.

O jogo Imersão Total deu indícios que seria mais eficiente para alunos iniciantes, pois o mesmo apresenta assuntos pertinentes a esse nível e também para motiva-los a não deixarem o curso por saberem que eles também podem criar algo parecido.

5. CONCLUSÃO

O principal objetivo do presente trabalho foi à criação de um jogo educativo que sirva de apoio à aprendizagem dos conceitos introdutórios de programação em Java como variável, condicional e laço de repetição para os alunos iniciantes. Para tanto, foi idealizado e desenvolvido o jogo Imersão Total. Para atingir esse objetivo foi realizado um levantamento dos jogos já existentes destinados ao aprendizado de programação.

Na avaliação o jogo Imersão Total deu indícios que a sua variedade de desafios, as explicações contextualizadas com o enredo do jogo, as recompensas dadas aos jogadores ao superar os desafios e uma grande liberdade que o usuário possui são elementos que motivaram os alunos a jogarem e aprenderem os elementos de Java.

O jogo apresentou indícios que proporciona aos jogadores momentos prazerosos e que os motivem nos seus estudos de Java, por facilitar a compreensão de conceitos não vistos ainda pelos alunos iniciantes abordados nas disciplinas de programação, nas quais a linguagem abordada seja Java.

Como trabalhos futuros sugere-se a ampliação dos desafios no jogo, assim como as explicações presente no jogo. De forma geral, a ampliação do mundo do jogo com mais personagens para o jogador interagir e armas que ele possa usar assim como novas habilidades. Podem-se adicionar novos sistemas no jogo como um sistema de fome e sono, para que o jogo se pareça mais com o mundo real.

Podem-se ter desafios em que o jogador tenha que codificar, e assim abordar uma característica muito comum na programação que são os erros de sintaxe e lógica. E, por fim, realizar uma avaliação com os usuários finais que estejam vendo pela primeira vez a linguagem de programação Java para uma melhor avaliação.

6. REFERÊNCIAS

- ALBUQUERQUE, Rafael M. e ANTONIO, Francisco A. P. F. Concepção de jogos eletrônicos educativos: proposta de processo baseado em dilemas. In: VIII Brazilian Symposium on Games and Digital Entertainment, Rio de Janeiro, RJ, 2009.
- BARBOSA, L. S.; FERNANDES, T. C. B; CAMPOS, A. M. C. Takkou: Uma Ferramenta Proposta ao Ensino de Algoritmos. In: WEI – XIX Workshop sobre Educação em Computação, XXXI Congresso da Sociedade Brasileira de Computação. Natal-RN, 19 a 22 jul. 2011.
- BITTENCOURT, J.R. e GIRAFFA, L.M. Modelando Ambientes de Aprendizagem Virtuais Utilizando Role-Playing Games. In: XIV Simpósio Brasileiro de Informática na Educação, p. 683-692, 2003.
- BORGES, M. A. F. Avaliação de uma metodologia alternativa para a aprendizagem de programação. VIII Workshop de Educação em Computação WEI, 2000.
- CARLOS, José, RAPKIEWIEZ, Clevi Elena. O Processo de Ensino-Aprendizagem de Programação: Uma Visão Crítica da Pesquisa no Brasil. Campos dos Goytacazes, Rio de Janeiro, 2003.
- CHAVES DE CASTRO, T., CASTRO JÚNIOR, A., MENEZES, C., BOERES, M. e RAUBER, M. (2003) “Utilizando Programação Funcional em Disciplinas Introdutórias de Computação”. *XI Workshop de Educação em Computação – WEI 2003*. Campinas, SP, Brasil.
- CLUA, Esteban e BITTENCOURT, João Ricardo. Uma nova concepção para a criação de jogos educativos. PUC, Rio de Janeiro/RS, 2004.
- CODEHUNT. CodeHunt. Disponível em: <https://www.codehunt.com/>. Acesso em 13/05/2016.
- CRISTINA, V., CABRAL, P.. Ensino-aprendizagem de Programação para Iniciantes: uma Revisão Sistemática da Literatura focada no SBIE e WIE. In: 23º Simpósio Brasileiro de Informática na Educação, 2012.
- DELGADO, C., XEXEO, J. A. M., SOUZA, I. F., CAMPOS, M., RAPKIEWICZ, C. E. (2004) “Uma Abordagem Pedagógica para a Iniciação ao Estudo de Algoritmos”. In:*XII Workshop de Educação em Computação (WEI'2004)*. Salvador, BA, Brasil.
- DETERS, J. I.; SILVA, J. M. C.; MIRANDA, E. M.; FERNANDES, A. M. R. Proposta de Diretrizes para Avaliação de Objetos de Aprendizagem Considerando Aspectos Pedagógicos e Técnicos. *XIX SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO*. 2008.
- DIJKSTRA, E. W. (1989). On the Cruelty of Really Teaching Computing Science. In: Communications of ACM, Issue 12, (vol.32), 1398-1404.
- ENTERBRAIN. Enterbrain. Disponível em: <https://www.enterbrain.co.jp>. Acesso em 09/05/2016.
- FALKEMBACH, G. A. M., AMORETTI, M. S. M., TAROUCO, L. R., VIERO, F. (2003) “Aprendizagem de Algoritmos: Uso da Estratégia Ascendente de Resolução de Problemas”. 8º Taller Internacional de Software Educativo. Santiago, Chile.

GRANDO, R. C. O jogo na educação: aspectos didático-metodológicos do jogo na educação matemática. Unicamp, 2001
<www.cempem.fae.unicamp.br/lapemme/cursos/el654/2001/jessica_e_paula/JOGO.doc>
Acesso em 10/04/2016.

HALSTEAD, N. “Uses of Pseudo Code in development”. 2007. Disponível em: <http://blog.assembleron.com/2007/06/03/uses-of-pseudo-code-in-development/3>. Acesso em: 07/05/2016.

HIENTERHOLZ, JR, O. Tepequém: uma nova Ferramenta para o Ensino de Algoritmos nos Cursos Superiores em Computação. XVII WEI – Anais do Workshop sobre Educação em Informática. Bento Gonçalves – RS, 2009.

KAMIYA, R. R. e BRANDÃO L. O.. iVProg – um sistema para introdução à Programação através de um modelo Visual na Internet. In: Instituto de Matemática e Estatística – Universidade de São Paulo (USP). São Paulo, SP, 2009.

KISHIMOTO, Tizuko M. O jogo e a educação. São Paulo, SP: Pioneira, 2003.

LIGHTBOT. LightBot. Disponível em: <https://lightbot.com/>. Acesso em: 14/05/2016.

MAIKE, V. R. M. L.. MIRANDA, L. C.. BARANAUSKAS M. C. C. 2011. Investigando sobre Requisitos para um Jogo de RPG com Professores de uma Escola Pública de Ensino Fundamental. XXII SBIE. Aracajú SE.

MALONEY. J. Peppeler, K. and Kafai, Y. Resnick, M. Rusk, N. Programming by Choice: Urban Youth Learning Programming with Scratch. In ACM SIGCSE Bulletin archive, Vol 40, 367-371, ISSN:0097-8418, 2008.

MANSO, António, OLIVEIRA, Luís e MARQUES, Célio Gonçalo. Portugol IDE – Uma ferramenta para o ensino de programação. 2004.

MARTINS, S. W.; MENDES, A. J.; FIGUEIREDO, A. D. Comunidades de Investigação em Programação: Uma Estratégia de Apoio ao Aprendizado Inicial de Programação. IEEE-RITA (Revista Iberoamericana de Tecnologias de Aprendizagem), v.5, n.1, p. 39-46, feb. 2010.

MÉLO, F. N., CUNHA, R. R. M., SCOLARO, D. R. e CAMPOS, J. L.. Do Scratch ao Arduino: uma proposta para o ensino introdutório de programação para cursos superiores de tecnologia. In: XXXIX Congresso Brasileiro de Educação em Engenharia, Blumenau, SC, 2011.

MOREIRA, J. V.; YNOGUTI, C. A. Piratas do Futuro - Ferramenta para suporte ao ensino de algoritmos e lógica de programação. In: Anais do congresso de iniciação científica do INATEL - INCITEL, 2012.

MORENO, A., MYLLER, N., SUTINEN, E., BEN-ARI, M. Visualizing Programs with Jeliot 3, Proceedings of the Advanced Visual Interfaces. 2004.

MOTA, M.P., PEREIRA, L.W.K. e FAVERO, E.L. JavaTool: Uma Ferramenta para o Ensino de Programação. In: Congresso da Sociedade Brasileira de Computação. Belém. XXVIII Congresso da Sociedade Brasileira de Computação, 2008.

O’KELLY, J; GIBSON, J. P. RoboCode&Problem-Based Learning: A non-prescriptive approach to teaching programming. SIGCSE conference on Innovation and technology in

computerscienceeducation. 2006. Pages 217-221, Disponível em:<http://dl.acm.org/citation.cfm?id=1140182>. Acesso em: 1 de abril de 2016.

OLIVEIRA, W. K. F. Vila Python: um jogo RPG para auxiliar a aprendizagem de programação com uso de linguagem Python. Rio Tinto/PB, 2015.

PASSERINO, L.M. (1998). Avaliação de jogos educativos computadorizados. Disponível em: <<http://www.c5.cl/tise98/html/trabajos/jogosed/>> Acesso em: 1 de dezembro de 2015.

PERKINS, D. N., SCHWARTZ, S. and SIMMONS, R.; (1988). Instructional Strategies for the Problems of Novice Programmers. In R. E. Mayer (ed.), Teaching and Learning Computer Programming, p.153-178.

RAPKIEWICZ, C. E., FALKEMBACK, G., SEIXAS, L., SANTOS, N. S., CUNHA, V. V., KLEMANN, M. Estratégia Pedagógica no Ensino de Algoritmos e Programação Associadas ao Uso de Jogos Educacionais. Revista Novas Tecnologias na Educação, v.4, n.2, p.1-11, 2006.

ROBOZZLE. Robozzle. Disponível em: <https://www.robozzle.com>. Acesso em: 20/03/2016.

RODRIGUES, M. C. Como Ensinar Programação? In: *Informática – Boletim Informativo Ano I n° 01*, ULBRA. Canoas, RS, Brasil, 2002.

RPGMAKERVXACE. RPG Maker Vx Ace. Disponível em: <http://store.steampowered.com/app/220700/?l=portuguese>. Acesso em: 19/05/16.

SANTIAGO, R. Ferramenta de apoio ao ensino de algoritmos. In: XII Seminário de Computação. Blumenau/SC, 2004.

SOUZA, C.M. (2009). VisuAlg – Ferramente de Apoio ao Ensino de Programação. In: Revista TECCEN – Vol. 2, no.2; ISSN 1984-0.

STEAM. Steam. Disponível em: <http://store.steampowered.com/>. Acesso em: 25/05/2016.

VAHLDIK, A; MATTOS, M; HUGO. M; CORREA, C. H. Futbot. Disponível em: <https://www.inf.furb.br/poo/futbot> . Acesso em: 20/05/2016.

7. APÊNDICE A – Documento de game design do jogo Imersão Total

Game Design:

Imersão Total

“Um RPG Educativo para apoiar a Aprendizagem de Programação”

Todos Direitos Reservados © 2016

Gilson Trajano

1.0

Abril de 2016

Índice

1. Introdução.....	46
2. Visão Geral do Jogo.....	47
Quanto ao tipo de objetos manipulados	47
Quanto aos tipos de desafio do jogo.....	47
Quanto ao número de usuários.....	47
Quanto aos oponentes do jogo	47
Quanto ao tipo de câmera e visão de jogo.....	47
Quanto ao tipo de cenários e mundos	48
3. Outras Questões sobre o Jogo.....	49
Que tipo de jogo é este?	49
Qual é o seu público-alvo?	49
Qual é o objetivo principal?.....	49
O que leva o jogador a perder o jogo?.....	49
Qual o tempo de duração previsto para o jogo?	49
4. Enredo.....	50
5. Recursos do Jogo	52
Motor de Jogo (Game Engine)	52
Recursos Gerais	52
Características Principais.....	52
Sistema Mínimo / Recomendado.....	53
6. Gameplay	54
Funcionamento Básico	54
Teclas e controles	54
Regras gerais do jogo.....	54
Pontuação.....	54
7. Interface de Jogo	56
8. Menus / Passo a Passo	57
Abertura	57
Menu Principal	57
Menu Opções.....	58
Menu Iniciar Partida	58
Menu Fim de Jogo.....	60
9. Personagens do Jogo.....	61
10. Cenários de Jogo.....	62
11. Músicas e Sons	64

1. Introdução

Este é um jogo do tipo RPG desenvolvido para apoiar a aprendizagem da sintaxe básica de Java, como por exemplo, tipos de dados e operações com variável, estrutura condicional e laço de repetição, tendo como elementos diferenciais uma contextualização do assunto para o mundo em que se apresenta o jogo, com uma variedade de estilos de desafios como questões de múltiplas escolhas, práticas (desafios em que o jogador deve realizar certas tarefas que são propostas em um código) e a utilização de analogias para apresentar os conceitos de programação como o assunto de laço de repetição.

Os conteúdos não se limitam a um assunto específico, como se encontra em outros jogos, mas sim abordando o conteúdo da sintaxe básica de Java, como tipos de variáveis, estrutura condicional e laço de repetição, sempre apresentando uma explicação sobre o porquê das respostas no desafio, podendo ser no inicio, meio ou fim da *quest*, o jogador, em alguns casos, pode escolher se quer a ajuda, dando maior liberdade que o estilo do jogo exige.

Para finalizar, tem-se um sistema diferente no tratamento dos erros com o jogador. No jogo, em certos casos, quando o jogador erra, pode-se utilizar os possíveis erros do jogador como parte do enredo do jogo, pois nesse momento seus companheiros darão dicas ou explicações de forma contextualizada.

2. Visão Geral do Jogo

Quanto ao tipo de objetos manipulados

O jogador controlará um personagem que é a sua representação no jogo, podendo selecionar seu sexo e nome. Ele é um programador que está participando de uma olimpíada de programação e que busca ser o melhor na competição. Na olimpíada, o jogador se une a outros dois personagens que vão fazer parte do seu grupo ajudando-o com explicações no desenvolver do jogo.

Dentre as ações dele, tem-se a possibilidade de interagir com outros personagens que irão apresentar um desafio ao jogador, apresentar alguma dica específica sobre o desafio ou apenas fazer um comentário referente a história do jogo. O jogador pode também ganhar ou encontrar moedas que podem ser usadas para comprar armas ou itens, assim como também poderá receber itens e armas ao realizar alguma *quest* no jogo ou ao encontrar algum baú.

Quanto aos tipos de desafio do jogo

Os desafios que o personagem principal encontrará são variados abordando: *quests* (desafios) com assuntos educativos como perguntas; busca por item para abrir baús ou portas; e analogias que são abordadas pelos acontecimentos do jogo e que leva a outro tipo de desafio com monstros que o jogador precisar derrotar ou fugir, dependendo do caso, que irão atacá-lo quando ele estiver na floresta.

Quanto ao número de usuários

O jogo é *Single player*.

Quanto aos oponentes do jogo

O principal adversário do jogador é o hacker, mas ele só fará sua aparição em dois momentos no jogo, uma inicialmente para apresentá-lo e no final, onde terá o confronto final. Entretanto, ele será responsável por alguns desafios que o jogador encontrará no jogo. Além do Hacker, o jogador vai ter vários inimigos, que serão os monstros que estão na floresta, sendo estes animais como lobos, aranhas, slimes, dragões, medusas entre outros.

O jogador pode interagir com esses inimigos fugindo deles ou utilizando o arsenal de armas que ele pode comprar ou ganhar no jogo, para atacar ou se defender dos monstros. Quando for atingido por um inimigo, ele perde um percentual da sua HP e que é baseado na força do inimigo. Ao atingir um monstro, se o mesmo já se encontrar fraco, o monstro morre.

Por fim, terão personagens no jogo que não são necessariamente inimigos, mas que dificultam a passagem do jogador, como por exemplo, os guerreiros que ficam na saída das cidades do jogo e que só permitem a passagem do jogador se ele responder a uma pergunta.

Quanto ao tipo de câmera e visão de jogo

A visão que o jogador tem é uma visão geral de cima, mostrando o jogador e o ambiente com uma visão aérea, permitindo o jogador ver sua equipe e os monstros no seu campo de visão.

Quanto ao tipo de cenários e mundos

Os cenários do jogo apresentam basicamente três tipos: as cidades, onde o jogador pode entrar em certos estabelecimentos como lojas, tabernas e estalagens; as florestas, com rios e montanhas, sendo que as cidades e florestas apresentam uma iluminação que muda com o tempo e cria um ambiente de dia e noite para o jogador; e a masmorra que são cavernas com buracos e cristais em suas paredes com uma iluminação mais escura para ambientar essa área do jogo.

Apresentando uma grande variedade de cores, pois utiliza muitos elementos como árvore, troncos, arbustos, gramas, rochas, áreas secas e rios, recriando assim um ambiente parecido com o mundo real.

3. Outras Questões sobre o Jogo

Que tipo de jogo é este?

O gênero do jogo é RPG (Role-Playing game) educativo.

Qual é o seu público-alvo?

Por se tratar de um jogo educativo, o público alvo são estudantes das disciplinas iniciais de programação que estejam aprendendo a linguagem Java.

Qual é o objetivo principal?

O jogador concluirá o game quando encontrar o Hacker e o derrotá-lo em uma batalha ou luta, mas para chegar nesse ponto, o jogador tem que superar vários desafios que o guiarão na localização do Hacker, ao vencer as *quests*. O personagem principal vai evoluindo e ganhando novas habilidades que o ajudará no seu desafio final.

O que leva o jogador a perder o jogo?

O jogador perderá no jogo quando seu HP (pontos de vida) chegar a zero, mas no ambiente do jogo existem item que podem recuperar os danos que o personagem principal sofre.

Qual o tempo de duração previsto para o jogo?

A estimativa de tempo varia muito de usuário para usuário por se tratar de um RPG, mas a previsão geral para finalizar o jogo é de trinta minutos à uma hora de jogo.

4. Enredo

No ano de 2116, a Ciência da Computação evoluiu de tal forma que está em todas as áreas. E, para encontrar a nova geração de programadores, é usada a olimpíada de programação, com a técnica de imersão total em que os competidores são levados a um mundo virtual onde precisarão superar os desafios para vencer e sobreviver.

O jogo se passa em um mundo virtual com cenários da idade média, com monstros e criaturas mágicas, aonde os competidores da olimpíada iriam inicialmente transpor os desafios e derrotarem os monstros para decidir quem venceria a competição, utilizando as características do MMORPG (Massively Multiplayer Online Role-Playing Game) para analisar os conhecimentos dos competidores em relação ao conteúdo e ao trabalho em equipe dos jogadores.

Mas, pouco tempo depois de iniciada, a olimpíada é atacada por um hacker que utiliza o sistema de imersão para prender os competidores no jogo e agora os participantes da olimpíada vão ter que buscar uma saída do jogo, transpondo os obstáculos do jogo e os deixados pelo hacker, mas apenas três competidores decidem enfrentar o hacker e encontrar uma maneira de saírem dessa prisão em que o jogo se tornou.

O jogo segue o seguinte fluxo:

Primeiro ponto: Inicialmente, o jogador está na cidade inicial. Cidade em que o jogo é iniciado e o lugar onde o jogador adquire itens e equipamentos, e os jogadores descobrem que estão presos dentro do game. Diante disso, os jogadores procuram alguma forma de sair. Ao tentarem sair da cidade respondendo a uma pergunta, o hacker surge, reafirmando que eles estão presos, mas que vai deixar uma pista sobre a sua localização dentro do jogo insultando os competidores ao insinuar que eles não são capazes de encontrá-lo, pois não conseguiram superar os desafios e nem teriam o conhecimento necessário para chegar até ele.

Segundo ponto: Ao seguir buscando a saída do jogo, o grupo chegará à cidade condicional. Para entrar, precisarão encontrar a combinação certa das alavancas que está representada por um código na placa de entrada da cidade. O código é formado por uma estrutura condicional de if dentro de if e que, para abrir a passagem, todas as condições precisam ser verdadeiras. A cidade encontra-se assim para que os monstros que estão na floresta não invadam a cidade.

Quando o grupo entrar na cidade, eles descobrem que o medo dos monstros na cidade é grande e para sair precisam provar que tem inteligência para sobreviver lá fora. Mas para chegar onde o hacker está, eles precisam sair da cidade Condisional e chegarem a Masmorra, o lugar onde o hacker está controlando o sistema de imersão do jogo e deixando os competidores presos.

Terceiro ponto: Ao tentar sair da cidade, o grupo é abordado por uma guerreira que só os deixará passar se responderem corretamente a uma pergunta sobre a diferença em usar o if e o else if em relação a apenas usar if. No caminho para a masmorra, eles ficam presos em um loop em que para sair dele, precisam derrotar um monstro no tempo limite, senão serão tele-transportados para a saída da cidade Condisional não conseguindo sair desse loop.

Ao sair do loop, o jogador chega à masmorra, mas para conseguir acessar a parte em que possivelmente está o hacker, ele precisa abrir uma porta, e para isso tem que usar o sistema de pesos e polias para abrir, tendo que buscar pequenos sacos para usar como pesos e abrir a porta. Tendo acesso a essa nova área, o jogador finalmente encontra o hacker, mas ele foge para outro plano. Os companheiros do jogador conseguem abrir uma passagem ao invadir o sistema e como bônus, também conseguem alterar o sistema de batalha, aumentando o nível do personagem do jogador e lhe dando novas habilidades para enfrentar o hacker em uma batalha final.

Quando o hacker é derrotado, o jogador descobre que o sistema nunca foi invadido e que tudo não passa de um teste que consistia da avaliação dos competidores sobre muito estresse para verificar suas ações e raciocínio lógico sobre pressão.

5. Recursos do Jogo

Motor de Jogo (Game Engine)

O jogo Imersão Total foi desenvolvido utilizando a engine RPG Maker Vx Ace da *Enterbrain* que é um aplicativo que permite a criação de jogos para a plataforma *Windows*, sem que seja necessário codificar em alguma linguagem de programação propriamente dita.

A ferramenta traz uma maneira de desenvolvimento orientada a eventos, ou seja, a cada elemento da tela pode-se associar um conjunto de eventos, possuindo várias versões da ferramenta, sendo a mais atual intitulada RPG Maker MV, embora tenha sido utilizada no presente trabalho, a versão RPG Maker Vx Ace. Mas estão disponíveis em português e podem ser encontradas na *Steam*.

O RPG Maker Vx Ace possui uma ampla base de dados com imagens de personagens, dos cenários, sons e animações. Tendo ainda uma comunidade de usuários ativa, que disponibilizam materiais gratuitamente, além de oferecer a venda de materiais, permitindo a criação de jogos de RPG das mais variadas formas e estilos.

Recursos Gerais

- Um banco de dados com Personagens/ Inimigos/ Habilidades/ Equipamentos;
- Uma grande base de dados com imagens, sons e animações;
- Um editor de mapa aprimorado para construção de mundo;
- Desenvolvimento na ferramenta fácil e intuitiva;
- Capacidade de adicionar novos recursos;
- Desenvolvimento orientado a eventos;
- *Engine* em português.

Características Principais

- Múltiplos ambientes para explorar;
- Sistema de batalha ativa;
- Múltiplos sons no jogo;
- Variedade de desafios.

Sistema Mínimo / Recomendado

Requisitos de sistema mínimo:

SO: Windows XP/Windows Vista/Windows 7/8/10

Processador: 1.6 GHz

Memória: 128 MG de RAM

Placa de vídeo: DirectX 9.0 Compatible

DIRECTX: Versão 9.0

Armazenamento: 100 MB de espaço disponível

Placa de som: DirectX 9.0 Compatible Sound

6. Gameplay

Funcionamento Básico

O jogo consiste em explorar os cenários em busca de pistas deixadas pelo Hacker e resolver desafios, tendo que enfrentar criaturas e monstros mágicos quando estiver nas florestas ou na masmorra, utilizando os controles para atacar de forma física ou com alguma habilidade.

Teclas e controles

Utilizando as teclas de navegação do teclado (cima, baixo, esquerda e direita) ou o mouse ao segurar o botão esquerdo para o jogador seguir o ponteiro, onde o jogador poderá controlar a movimentação do seu personagem nos cenários do jogo. Para atacar, o jogador deve usar as teclas ENTER/Z ou o botão direito do mouse, para defender utiliza a tecla A, para usar alguma habilidade usa a tecla D, para utilizar um item utiliza S e utiliza a tecla Q para acessar um menu rápido que contém uma lista dos itens e se pressionar novamente irá apresentar uma lista das habilidades do personagem, permitindo que ele configure qual é a habilidade que o personagem vai usar no jogo.

Se pressionada uma das teclas ENTER/ESPAÇO/Z próximo de um NPC (personagem controlado pelo computador) o jogador poderá interagir com o mesmo através de caixas de diálogos, se for algum item ele poderá pegá-lo.

Ao pressionar as teclas ESC/X irá cancelar ou voltar nas opções do menu principal, apagará quando for escrever o nome do personagem e para acessar o menu principal com os status dos personagens quando estiver jogando.

Regras gerais do jogo

O jogador poderá movimentar seu personagem em varias direções como cima, baixo, esquerda, direita e diagonalmente. Ele poderá correr, só que por um curto período de tempo e poderá pular também, mas em certas áreas o personagem não pode passar como em árvores, arbustos, casas, água, paredes, pedras e buracos, sendo que pode realizar ataques que destroem alguns elementos do cenário como arbustos e no caso das pedras, se for utilizado uma bomba.

Em áreas descampadas, o jogador pode se movimentar livremente, podendo andar em cima de gramas e flores que se encontram nessas áreas. Ele não pode voar ou nadar nos rios, lagos e nem escalar as montanhas, apenas podendo subir nelas se tiver um acesso em que ele possa ir andando e estando em cima da montanha, ele também não pode pular de lá para baixo.

O jogador pode comprar, encontrar ou ganhar itens que podem melhorar seus atributos, como aumentar o ataque ou a defesa, a sorte, inteligência, resistência e agilidade, mas da mesma forma, terão itens que podem diminuir esses mesmos atributos. Para comprar os itens, é necessário dinheiro, que no jogo são moedas, pois o personagem pode ganhar ao realizar um desafio ou quando derrotar um monstro que deixou moedas ou itens.

O personagem principal pode dormir nas estalagens, mas para isso ele tem que ter moedas, e, ao dormir, o jogador poderá salvar o jogo e recuperar todos os danos que levou e poderá também salvar o jogo em cristais que estão espalhados pelo jogo, só que neles o jogador não recupera o dano que possa ter.

Não é possível ao jogador matar os personagens que são essenciais ao desenvolvimento da história do jogo. E de uma forma geral, o jogo é de mundo aberto por dar muita liberdade ao jogador.

Pontuação

Para cada desafio superado em relação ao conteúdo Java, o jogador aumentará seu nível, lhe dando novas habilidades. Para cada monstro derrotado, o jogador ganhará XP (Experience Points), permitindo que o jogador aumente seu nível, podendo também ganhar moedas e itens como poções de cura.

7. Interface de Jogo

A interface do jogo consiste em um ambiente com cidades, florestas e masmorras onde o jogador pode se movimentar pelos bosques, enfrentando os monstros. Sendo apresentada uma HUD com as informações básicas como seu HP, SP e quais são os equipamentos que ele está utilizando no momento, como itens que podem ser poções que vão curá-lo ou que lhe apresente uma mensagem no caso de um pergaminho.

Sendo o jogo no ambiente da idade media fantasiosa, com seres que podem usar magia e seres como demônios, dragões, medusas e fantasmas.

8. Menus / Passo a Passo

Abertura

Iniciando-se o jogo, a seguinte ordem de apresentação será utilizada:

1. Será apresentada uma tela inicial com o logotipo do jogo na abertura.
2. O Menu principal será acessado.

Menu Principal

Ao iniciar o jogo, será apresentado ao jogador um menu principal com as opções “Novo Jogo” para iniciar um novo jogo e “Continuar” que levará a uma nova tela em que o jogador poderá selecionar o jogo de um ponto já salvo, a opção “Ajuda” apresenta uma tela com as teclas utilizadas para controlar o personagem e para acessar os menus do jogo, créditos do jogo e sair. Na Figura 1 têm-se a tela do menu principal.



Figura 9 Menu Principal do jogo

Menu Opções

Ao se clicar em Novo Jogo, irá aparecer o sub-menu para selecionar o sexo do personagem e posteriormente colocar um nome, onde pode ser visto na Figura 2.



Figura 10 Submenu seleção do personagem

Menu Iniciar Partida

Ao estar jogando, o jogador tem acesso a dois menus, um é o menu rápido, que pode ser visualizado na Figura 3, e que dá acesso a uma lista de itens e habilidades do jogador.



Figura 11 Menu Rápido

O outro menu que aparece durante o jogo apresenta informações mais completa do estado do personagem e da sua equipe, como os itens, habilidades, equipamentos que apresenta, a armadura, espada, escudo que está usando e o que esse equipamento melhora no personagem, apresenta também condições que mostram o estado completo do personagem, por exemplo, como está o seu nível atual, quantos pontos ele precisa para aumentar de nível, seu ataque, defesa, inteligência, resistência, agilidade, sorte e também para sair do menu, mostrado na Figura 4. Na Figura 5, temos o sub-menu de itens, na Figura 6 mostra as habilidades, na Figura 7 vemos os equipamentos, na Figura 8 temos as condições e na Figura 9 mostra como sair do menu.



Figura 12 Menu Partida

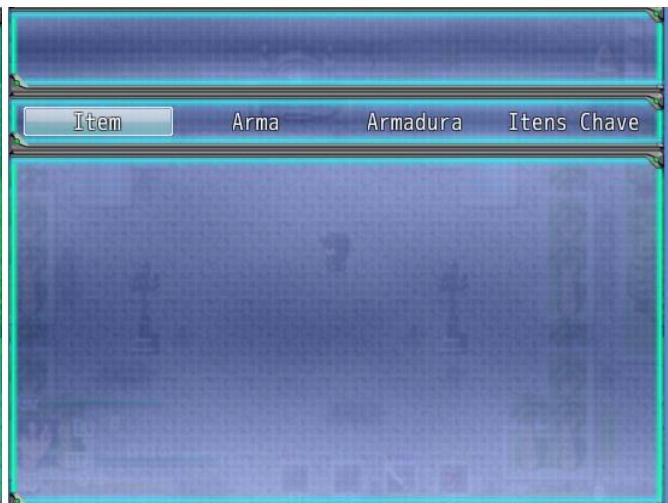


Figura 13 Submenu item

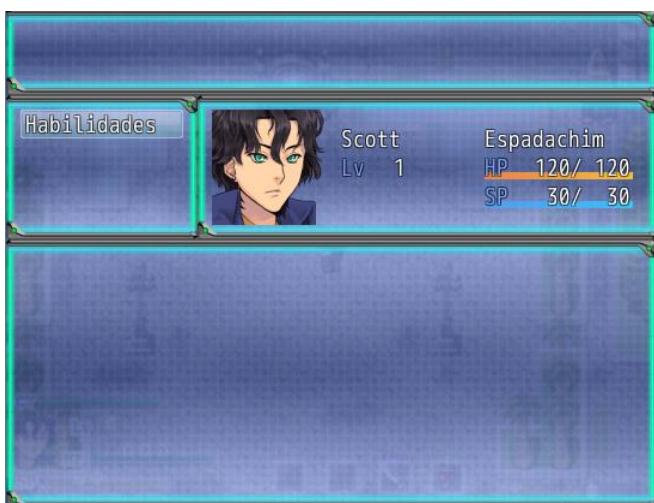


Figura 14 Submenu Habilidade

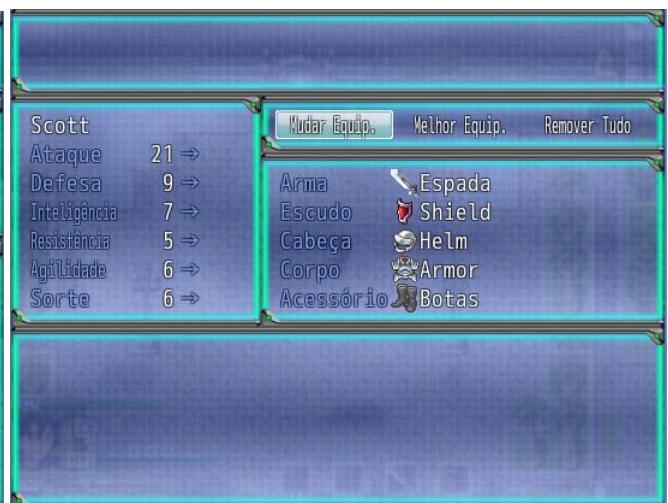


Figura 15 Submenu Equipamento



Figura 16 Submenu Condições



Figura 17 Submenu Sair

Menu Fim de Jogo

Não tem um menu para sair do jogo. Se o jogador for morto, será mostrada uma tela de *game over* e depois será direcionado para o menu principal do jogo. Quando finalizar o jogo, a cena vai escurecendo até ser levado ao menu principal.

9. Personagens do Jogo

Personagens:

• Personagem principal – O personagem que será controlado pelo jogador. O jogador pode informar qualquer nome para o avatar e pode escolher o sexo do personagem. Por padrão, eles terão os nomes de Scott e Glazy.

- Linus – O NPC que faz parte da equipe do personagem principal.
- Emily – A NPC feminina que faz parte da equipe do personagem principal.
- Hacker – O NPC que é o principal inimigo de Scott.
- Guarda – Um NPC que fica na entrada da cidade e só permite a entrada se responder corretamente a sua pergunta, cobrando um valor caso eles errarem.

• Senhor – Um NPC que precisa de ajuda para recolher cogumelos para a sua esposa que está doente.

• Dono do PUB (Taverna) – NPC que gerencia um estabelecimento, e que pode informar ao jogador o funcionamento básico do jogo, como controles, informação sobre monstros e o sistema de fome do jogo.

• Dono da INN (Hospedaria) – NPC que gerencia uma estalagem, permitindo ao jogador dormir por um preço e que possibilita salvar o jogo.

- Dono da loja de item – NPC que vende itens em sua loja.
- Dono da loja de armas – NPC que vende armas em sua loja.
- Monstros – NPCs que atacarão o jogador quando estiverem na floresta.

Armas:

- Espada – Uma espada leve e fácil de manusear com lâmina curta.
- Escudo – Escudo de aço para a guerra.
- Bomba – Bomba utilizada para explodir objetos.

Itens:

- Poção – Item que recupera os pontos de vida do jogador.
- Ether – Item que recupera os pontos mágicos do jogador.
- Gancho – item que permite a travessia de penhascos ou rios.
- Cogumelo – É um item que é recolhido na realização de uma *quest*.

10. Cenários de Jogo

Os cenários são formados por duas cidades, a cidade Inicial e a Condicional. Na Figura 10, têm-se a visão geral da cidade Inicial, onde o jogador poderá entrar em algumas lojas para comprar item e equipamentos. O jogo possui quinze cenários de florestas com rios, montanhas, matas fechadas e áreas abertas. Será nas florestas que o jogador encontrará os monstros que terá que derrotá-los ou fugir, e encontrará obstáculos naturais como riachos e rochas impedindo a passagem, como vemos na Figura 11. Na figura 12, vemos uma masmorra formada por labirintos com cristais e buracos e um espaço vazio onde será travada a batalha final contra o Hacker.



Figura 18 Visão geral da Cidade Inicial

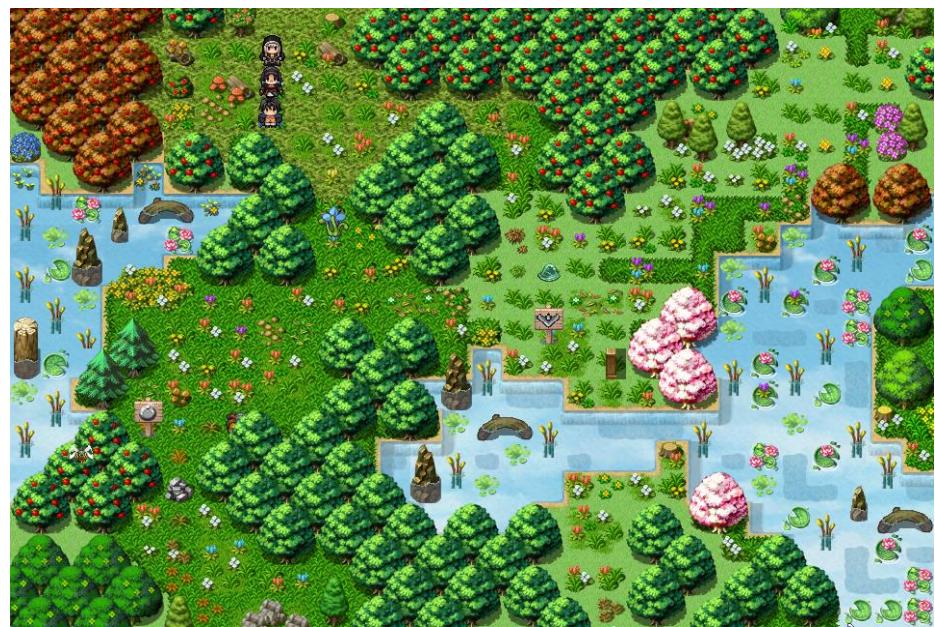


Figura 19 Floresta



Figura 20 Masmorra

11. Músicas e Sons

No background do jogo serão usadas músicas variadas. Algumas músicas sendo utilizadas apenas em alguns ambientes, como é o caso da masmorra em que a música ambiente dela só será usada lá. Além do som ambiente, quando ocorre o evento de chuva ou uma tempestade, serão utilizados sons para apresentar a chuva e o vento. Sendo também usado sons como ao abrir uma porta ao entrar em uma loja.

Os monstros emitirão sons de impacto ao realizar um ataque. Se for utilizada habilidade ou magia, as mesmas também terão sons, assim como o jogador que ao atacar com sua espada terá os sons cortante e de impacto.

8. APÊNDICE B – Roteiro do jogo Imersão Total

Roteiro do jogo

Início do jogo

Jogador: A competição foi atacada por um hacker e agora estamos presos no jogo. Temos que sair.

Linus: Não podemos ficar aqui sem fazer nada. Temos que buscar uma forma de sair desse mundo virtual.

Emily: Vamos sim, Linus e NOME DO JOGADOR.

Desafio: abrir caixa deixada pelo hacker

Jogador: Mas o que foi isso? Quase que tenho um ataque cardíaco!

Hacker: Vocês se acham os melhores programadores, mas agora são minhas peças de xadrez e terão que mostrar o quão bom vocês são. Deixarei uma pista sobre meu paradeiro, descubram como abrir esse baú e ficarão mais próximos de saber onde estou. Aviso-lhes que não será tão fácil assim...

Mensagem da caixa: Selecionar a chave do tipo inteiro para abrir o baú das chaves que se encontram acima.

Escolheu: chave= “3”;

Emily: Essa chave é do tipo *String*, logo não é um valor inteiro e, por isso, esta chave é errada NOME DO JOGADOR.

Escolheu: chave= 4.6;

Linus: Essa não funcionou por que é do tipo *double*, ou seja, tem um valor decimal.

Conteúdo da caixa: Mensagem: Meus parabéns, jogadores, se querem tanto me encontrar, estou na masmorra próxima da cidade Condisional, preparem-se, pois, ao longo do caminho existem desafios, além de uns “presentinhos” feitos especialmente para vocês.

Desafio: abrir baú

Emily: O baú está trancado com magia. E a única forma de abri-lo é com um item *boolean* que tenha o valor verdadeiro. Esses itens são como esferas vermelhas.

Saída da cidade Inicial

Guerreiro: Olá jovens aventureiros! Para sair da cidade vocês precisam responder a uma pergunta. Se acertarem poderão ir, mas, se errarem terão que pagar 30 moedas.

Qual é o valor exibido na tela?

```
int m = 1 - 25 / 2;  
System.out.println(m);  
Escolhas: -11, 11, -11.5 e 11.5 Correta: -11
```

Mensagem de resposta errada

Guerreiro: Errado! Perderam 30 moedas.

Mensagem de resposta correta

Guerreiro: Meus parabéns! Podem passar. Pela resposta correta, lhes concederei esses itens que irão ajudá-los em sua jornada. E não esqueçam de utilizar seus itens pressionando Q ou X. Vocês podem salvar o progresso do jogo em cristais que estão espalhados pelo jogo. Usem suas habilidades com cautela algumas delas consomem SP (Pontos Mágicos) do usuário.

Se não tiver moedas

Guerreiro: Não há moedas suficientes.

Linus: Deve ter algumas moedas pela cidade, vamos à procura.

Dúvida ao sair da cidade Inicial

Jogador: Não comprehendo! Por que o valor de m foi -11 e não -11.5? No int m = 1-25/2;

Emily: Você esqueceu de uma coisa básica NOME DO JOGADOR, os tipos de variáveis.

Linus: Porque estava dividindo um inteiro por outro inteiro e daí Java assume a divisão inteira eliminando o ponto flutuante, o 0.5.

Jogador: Entendo! Não lembrava disso. Você tem toda razão.

Linus: Vamos economizar umas moedas, vai que isso aconteça novamente.

Emily: Boa idéia, Linus! Vamos fazer isso, NOME DO JOGADOR.

Quest: busca pelos cogumelos

Senho: Olá aventureiros!

Jogador: Oi senhor!

Emily: Olá!

Senhor: Vocês poderiam ajudar um velho? Minha esposa está doente e eu preciso de alguns cogumelos para fazer o remédio, mas nesta floresta há muitas criaturas perigosas e eu já estou velho demais para enfrentá-las sozinho.

Senhor: Se me ajudarem, irei recompensá-los com 40 moedas e uma poção de cura. Aceitam?

Se sim:

Jogador: Claro, o ajudaremos.

Linus: Mas NOME DO JOGADOR, não podemos perder tempo.

Jogador: Linus, vamos ganhar moedas! Precisamos delas.

Senhor: Obrigado por aceitarem. São uns...

```
int quantCogumelos = 4;
```

```
quantCogumelos += 4;
```

cogumelos que vocês terão que pegar. E eles são verdes.

Linus: O que foi, NOME DO JOGADOR?

Emily: Não entendeu o que é para fazer?

Se sim:

Jogador: Não entendi, por que o `+=`? Para que serve?

Emily: Por exemplo, o código `num += 1` é igual ao `num = num + 1`, o `+=` é só uma forma de simplificar a linha de código.

Jogador: Agora entendi.

Se não:

Jogador: Já sei o que precisamos fazer, Emily.

Emily: Certo, NOME DO JOGADOR!

Se não:

Senho: Ah, certo! Vocês devem estar ocupados demais.

Mensagem de quando a quest é terminada.

Senho: Conseguiram? Muito obrigado! Como prometido, peguem sua recompensa.

Mensagem de quando o jogador falar com o senhor, com a quest em andamento

Senhor: São uns...

`int cogumelos = 4;`

`cogumelos += 4;`

cogumelos que vocês terão que trazer para mim. E eles são verdes.

Entrada da cidade Condisional

Linus: Enfim, estamos chegando à cidade Condisional.

Mensagem da placa:

```
if (alavancaVermelha.equals("Cima"))
    if(alavancaVerde.equals("Baixo"))
        if(alavancaAzul.equals("Cima"))
            abrirPortão();
```

Emily: Olhe! No muro existem alavancas. São condições dentro de outras condições, então para abrir o portão teremos que deixar as alavancas iguais ao que o código pede.

Desafio: sair da cidade Condisional

Guerreira: Não podemos deixar qualquer um sair, fora da cidade existem muitos perigos. Para sair vocês devem provar que sabem sobre condicional. Se errarem o desafio, terão que pagar 30 moedas.

Pergunta:

Qual é o valor da variável `num` no final de cada código?

<code>int num = 1;</code>	<code>int num = 1;</code>
<code>if(num == 1){</code>	<code>if(num == 1){</code>
<code> num = num + 1;</code>	<code> num = num + 1;</code>
<code>}</code>	<code>}</code>
<code>if(num != 1){</code>	<code>else if(num != 1){</code>
<code> num++;</code>	<code> num++;</code>
<code>}</code>	<code>}</code>

Escolhas: 3 e 2, 2 e 3, 2 e 2, 3 e 3 Correta: 3 e 2

Mensagem de resposta errada

Guerreira: Errado! Perderam 30 moedas.

Mensagem de resposta correta

Guerreira: Muito bem! Acertaram. Como vocês sabem, usando o *if* e o *elseif* ao entrar no *if*, não vou mais executar o *elseif*, mesmo que a condição esteja correta, pois, o Java já pula todos os *elseif*. Mas no outro, como eram dois *if's*, logo, são executados os dois quando ambas as condições forem corretas. Tenham cuidado ao sairem da cidade.

Se não tiver moedas

Guerreira: Vocês não têm moedas suficientes.

Desafio: fantasma em loop

Jogador: O que estamos fazendo aqui? Fomos trazidos de volta?

Linus: Isso é como um loop, NOME DO JOGADOR. O Hacker nos prendeu em um loop.

Jogador: Era por isso que quando estávamos enfrentando as criaturas havia um tempo passando. Mas, Linus, o que é esse loop?

Linus: Você lembra de laço de repetição como, por exemplo, o *while* e o *for*?

Se sim:

Jogador: Claro! Lembro. Pergunta idiota a minha.

Se não:

Linus: Um laço de repetição é quando executamos o mesmo código, no nosso caso esse código seria para voltar pra cá, enquanto a condição é verdadeira. Eu acho que se enquadra como um *while* no nosso caso.

Emily: Mas, como vamos sair do loop? Como vamos deixar a condição falsa?

Linus: Deve ser um monstro que tem essa habilidade. O único que não tínhamos encontrado ainda foi aquele fantasma de olhos vermelhos. Ele é a condição do *while*, se o derrotarmos, vamos sair do *while*.

Desafio: abrir passagem

Mensagem: Para abrir a porta, temos que colocar três sacos na plataforma vermelha.

Emily: Vejam! Isso é como um comando *for*. Estamos realizando a mesma coisa três vezes, ou seja, estamos repetindo o código aqui.

Linus: O *for* é um loop como o *while* né.

Hacker: Como conseguiram chegar até aqui? Não pensem que me encurralaram.

Linus: Vejam! É um acesso emergencial dos ADM's. Era assim que ela estava acessando os comandos do jogo. Usaremos esse acesso emergencial para libertar todos.

Jogador: Temos que ir atrás do Hacker.

Linus: Espere, NOME DO JOGADOR, não podemos ir assim. Ela fugiu porque lá estará em vantagem. Você é o melhor lutador, NOME DO JOGADOR, aumentaremos seus atributos através do acesso dos ADM's. Espere, tem algo errado, não venha até aqui, NOME DO JOGADOR. Emily?!

Emily: Conseguimos! Iremos mandá-lo para onde está a Hacker, mas não poderemos ir, pois ficamos presos em uma armadilha e não podemos nos mexer.

Linus e Emily: Treine suas novas habilidades antes de enfrentar a Hacker, NOME DO JOGADOR, e equipe seus novos itens.

Jogador: Conseguí! Finalmente derrotei a Hacker.

ADM: Meus parabéns! Você acabou de vencer as olimpíadas de programação, NOME DO JOGADOR. A invasão nunca aconteceu realmente, tudo o que aconteceu fazia parte da competição. Estamos desconectando você no momento. Aguarde!

9. APÊNDICE C – Questionário utilizado na avaliação do jogo Imersão Total

Questionário de avaliação de jogos educacional Imersão Total

Imersão Total - 13/05/2016

Gostaríamos que você respondesse as questões abaixo para nos ajudar a melhorar este jogo. Todos os dados são coletados anonimamente e somente serão utilizados no contexto desta pesquisa.

Gilson Trajano Soares Costa - gilson.costa@dce.ufpb.br

UFPB – Campos VI

Por favor, marque **um número** de acordo com o quanto você concorda ou discorda de cada afirmação abaixo.

1. Você gosta do gênero de jogo RPG?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão:

2. O design do jogo é atraente (interface ou personagens)?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão:

3. Houve algo interessante no início do jogo que capturou sua atenção?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão:

4. A variação (de forma, conteúdo ou de atividades) ajudou a mantê-lo atento ao jogo?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão:

5. O jogo evolui num ritmo adequado e não fica monótono – oferece novos obstáculos, situações ou variações de atividades?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão: _____

6. Você se divertiu com o jogo?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão: _____

7. Você recomendaria este jogo para seus colegas?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão: _____

8. Gostaria de utilizar este jogo novamente?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão: _____

9. O jogo contribuiu para a sua aprendizagem na disciplina?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão: _____

10. O jogo foi eficiente para sua aprendizagem, em comparação com outras atividades da disciplina?

Discordo Fortemente	1	2	3	4	5	Concordo Fortemente
---------------------	---	---	---	---	---	---------------------

Comentário sobre a questão: _____

11. Cite pontos fortes do jogo: _____

12. Por favor, dê sugestões para a melhoria do jogo: _____
