

UNIOESTE – Universidade Estadual do Oeste do Paraná

CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS

Colegiado de Ciência da Computação

Curso de Bacharelado em Ciência da Computação

**Especificação e Protótipo de um Jogo Educativo para
Aprendizagem de Conceitos por Crianças Surdas**

Jeferson José Baqueta

CASCABEL

2012

JEFERSON JOSÉ BAQUETA

**ESPECIFICAÇÃO E PROTÓTIPO DE UM JOGO EDUCATIVO PARA
APRENDIZAGEM DE CONCEITOS POR CRIANÇAS SURDAS**

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação, do Centro de Ciências Exatas e Tecnológicas da Universidade Estadual do Oeste do Paraná - Campus de Cascavel.

Orientador: Prof. Dr. Clodis Boscaroli

CASCADEL

2012

JEFERSON JOSÉ BAQUETA

**ESPECIFICAÇÃO E PROTÓTIPO DE UM JOGO EDUCATIVO PARA
APRENDIZAGEM DE CONCEITOS POR CRIANÇAS SURDOS**

Monografia apresentada como requisito parcial para obtenção do Título de *Bacharel em Ciência da Computação*,
pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos
professores:

Prof. Dr. Clodis Boscarioli (Orientador)
Colegiado de Ciência da Computação,
UNIOESTE

Prof. Dr. Adair Santa Catarina
Colegiado de Ciência da Computação,
UNIOESTE

Prof. Heloir Aparecido Montanher
Colegiado de Pedagogia,
UNIOESTE

Cascavel, 29 de outubro de 2012.

DEDICATÓRIA

Dedico esse trabalho aos meus familiares, que me deram apoio nos momentos difíceis, meus professores, que me ajudaram durante a graduação contribuindo para minha formação e, sobre tudo, a Deus que me proporcionou mais essa conquista.

Lista de Figuras

3.1	Jogo Multi-trilhas	20
3.2	Jogo Libras Brincando e Aprendendo	20
3.3	Jogo Colisões	21
3.4	Jogo Kinble	21
4.1	Representação da arquitetura do jogo Conceitos	31
4.2	Exemplos de imagens para composição de um personagem	33
4.3	Exemplo de imagens utilizadas para composição por sobreposição	33
4.4	Resultado do processo de composição de um personagem	35
5.1	Diagrama MOLIC do jogo Conceitos	41
5.2	Diagrama CTT referente à criação de personagens	43
5.3	Diagrama CTT referente à realização de objetivos	44
5.4	Opções proposta ao jogador a iniciar o Jogo	45
5.5	Tela inicial implementada no Jogo	46
5.6	Mapeamento da funcionalidade Logar no jogo	47
5.7	Processo de para criação de conta	48
5.8	Processo para criação de personagem	49
5.9	Tela para criação de personagem	50
5.10	Formas de acessar a tela Escolha de Ambiente	51
5.11	Liberando um novo ambiente	52
5.12	Dinâmica empregada no jogo Conceitos	53
5.13	Interface do jogo Conceitos em sua parte 3D	54

5.14	Seleção de objeto correto.....	55
5.15	Seleção de objeto incorreto.....	55
5.16	Fim de ambiente.....	56
5.17	Tela de opções de jogo.....	57
5.18	Mensagem de conclusão de salvamento.....	58
5.19	Diagrama MOLIC redefinido.....	60
A.1	Tela inicial para criação de um projeto Java.....	71
A.2	Demonstração das opções que devem ser alteradas.....	71
A.3	Demonstração de como adicionar bibliotecas externas ao projeto.....	72
A.4	Lista de bibliotecas do jMonkey.....	72
A.5	Lista de bibliotecas da LWJGL.....	73
A.6	Adicionando bibliotecas nativas da LWJGL.....	74
A.7	Tela padrão do jMoneky para <i>display</i>	75
A.8	Resultado do código de configuração proposto.....	76
A.9	Acessando as propriedades do Computador.....	77
A.10	Configurações do sistema.....	78
A.11	Propriedades do Sistema.....	78
A.12	Variáveis de ambiente.....	79
A.13	Arquivos do fobs4 que devem ser copiados.....	80
A.14	<i>Plugin</i> fobs4 adicionado ao diretório do JMF.....	80
A.15	Caminho do <i>plugin</i> fobs4 adicionado a variável de ambiente CLASSPATH.....	81
A.16	Bibliotecas utilizadas para execução do jmf e do fobs4.....	81
A.17	Classe MediaPlayer.....	82
A.18	Classe principal AbrirVideo.....	83
A.19	Janela para seleção do vídeo.....	83
A.20	Resultado da execução da classe AbrirVideo.....	84

A.21	Acessando a opção <i>User Preference</i>	86
A.22	Janela para adicionar plug-ins.....	86
A.23	Janela do Blender já configurada para exportação Ogre3D	87
A.24	Janela do Blender para definir configurações de exportações	88
A.25	Salvando as preferências do usuário.....	89
A.26	3DView e janela de confirmação de exportação	90
A.27	Arquivos gerados pela exportação.....	91
A.28	Cabeça de macaco renderizada em jMonkey	92

Lista de Tabelas

2.1	Deficiências e alternativas de auxílio tecnológico	10
3.1	Tecnologias direcionadas a pessoas surdas.....	17
3.2	Exemplos visuais de tecnologias para surdos	18
4.1	Requisitos funcionais do jogo Conceitos.....	27
6.1	Critérios e nível da equipe de desenvolvimento	64
6.2	Funcionalidades 2D do Jogo	65
6.3	Funcionalidades 3D do Jogo.....	65

Lista de Abreviaturas e Siglas

ATC	Árvores de Tarefas Concorrentes
CAT	Comitê de Ajudas Técnicas
GNU	<i>General PublicLicense</i>
IDE	<i>Integrated Development Environment</i>
IHC	Interação Humano Computador
JMF	<i>Java Media Framework API</i>
LIBRAS	Língua Brasileira de Sinais
LWJGL	<i>Lightweight Java Game Library</i>
MOLIC	<i>Modeling Language for Interaction as Conversation</i>
SDK	<i>Sun Development Kit</i>
TA	Tecnologia Assistiva
TIC	Tecnologia de Informação e Comunicação

Sumário

Lista de Figuras	vi
Lista de Tabelas	ix
Lista de Abreviaturas e Siglas	x
Sumário	xi
Resumo	xiii
1 Introdução	1
1.1 Justificativas.....	2
1.2 Objetivos.....	3
1.3 Organização do Texto.....	3
2 Educação Especial e Inclusiva	5
2.1 Tecnologia como Auxílio a Educação.....	8
3 Surdez e Aspectos Cognitivos	11
3.1 A Educação do Surdo e o uso de Imagens.....	14
3.2 Tecnologias e Influência dos Jogos.....	16
4 Proposta do jogo Conceitos	22
4.1 Desenvolvimento de Jogos.....	22
4.2 O jogo Conceitos.....	24
4.3 Análise e Elicitação de Requisitos.....	26
4.4 O Processo de Desenvolvimento.....	29
4.4.1 Arquitetura.....	30

4.4.2	Composição de Imagens.....	32
5	Interação e Interface do Jogo	37
5.1	Diagrama MOLIC	39
5.2	Diagrama CTT	42
5.3	MOLIC e a Interação do Jogo Conceitos.....	45
5.3.1	Opções Iniciais.....	45
5.3.2	Carregar Jogo ou Logar.....	47
5.3.3	Criação e Edição de Conta	48
5.3.4	Criar Personagem.....	49
5.3.5	Escolher Ambiente de Jogo	50
5.3.6	Interação com o Ambiente 3D.....	53
5.3.7	Pausar e Opções de Jogo	56
5.3.8	Opções de Fim de Jogo	58
5.4	MOLIC para uma Nova Versão do Jogo Conceitos	59
6	Considerações Finais	63
6.1	Conclusão.....	63
6.2	Trabalhos Futuros	65
	Apêndice A	69
	Referências	94

Resumo

Atualmente os jogos estão cada vez mais presentes na vida das pessoas, dado que o acesso a computadores e *videogames* está ao alcance da maioria da população. Dessa forma, o consumo e a procura pelo entretenimento digital ganha espaço e prioridade na vida de muitos indivíduos. Além do uso de jogos para atividades relacionadas ao lazer e diversão, esses podem ser também empregados em escolas, sendo conhecidos como jogos educacionais. Os jogos educacionais podem ser utilizados no processo de ensino e aprendizagem de pessoas com deficiência, maximizando suas capacidades cognitivas. Este trabalho apresenta uma proposta sobre um jogo educacional, denominado Conceitos, para crianças surdas de 1^a a 5^a séries, que por meio de recursos visuais como imagens, textos, vídeos em Libras e animações poderá auxiliá-las no aprendizado da língua portuguesa e também da língua de sinais, contribuindo com a memorização e fixação de conceitos presentes no cotidiano dessas crianças a partir da interação com objetos tridimensionais presentes no universo virtual do jogo. O jogo ora apresentado, na forma de um protótipo, é apoiado em uma série de tecnologias livres que serão discutidas no decorrer do trabalho.

Palavras-chave: Tecnologia Assistiva, Jogos Educativos, Interação Humano-Computador, Computação Gráfica.

Capítulo 1

Introdução

Desde o nascimento as pessoas buscam formas de se expressar e interagir com os demais indivíduos a sua volta. O processo de interação com o mundo contribui para a construção do conhecimento e, sobretudo, no desenvolvimento do indivíduo. Assim, a partir de novas experiências o sujeito consegue aprender e armazenar novas informações.

Segundo Schirmeret *et al.* (2007), uma pessoa ao aprender algo novo acaba passando por três processos distintos: **(i) experiência sensorial**, onde a informação é recebida por órgãos sensoriais; **(ii) processamento das informações captadas**, onde o indivíduo tenta compreender e interpretar o significado da informação percebida e **(iii) emissão do comportamento**, onde é determinada a resposta que será executada pelo corpo. Quando um desses processos apresenta falha, essa disfunção pode ser identificada como uma deficiência.

A deficiência implica em sérias limitações e necessidades para o cotidiano do indivíduo, e estas podem prejudicar as relações estabelecidas por esse sujeito, tanto no meio social como educacional.

As limitações presentes na vida de um indivíduo com deficiência acabam se apresentando de forma mais severa do que para um indivíduo dito normal, principalmente em relação ao processo de ensino e aprendizagem, interpretação e concepção de informações vindas do seu entorno. A deficiência acarreta sérios problemas na construção do conhecimento e na interação com os demais indivíduos.

Quando a deficiência é relacionada a educação, ou seja, ao processo de ensino-aprendizagem, a inserção do sujeito com deficiência no meio escolar exige uma série de mudanças que muitas vezes não são possíveis de serem atendidas pelas escolas, como apontado por Correia (2006):

A diferença imposta pela deficiência se torna um fator relevante para a inserção do aluno com deficiência no meio do ensino, pois devido às

necessidades educacionais apresentadas pelo mesmo, certos recursos e metodologias devem ser adotados e agregados ao meio escolar para que o sujeito com deficiência possa ter uma educação adequada.

Contudo, grande parte das escolas ainda não possui condições de incluir o aluno com deficiência em seu meio, tal que esse aluno recebe sua educação em um modelo de ensino integrado ou especial paralelo ao ensino convencional.

Para Filho (2009), se o aluno não consegue ou é incapaz de compreender e interpretar as informações do meio, a escola que até então deveria exercer uma função de educar visando autonomia e independência, passa a educar de forma dependente e submissa. Se as dificuldades de comunicação proveniente de uma deficiência forem mitigadas a partir do uso de recursos tecnológicos ou de métodos alternativos de ensino, o aluno com deficiência pode ter suas limitações minimizadas.

A utilização de recursos educacionais de auxílio ao processo de ensino e aprendizagem como jogos, imagens, tecnologias computacionais, entre outros, pode influenciar de forma positiva na educação de indivíduos com deficiência, pois esses recursos podem contribuir tanto no desenvolvimento educacional como social do mesmo. Porém, cabe à escola e/ou à família a apropriação de tais recursos de modo a integrá-los em seu cotidiano.

1.1 Justificativas

Todos os indivíduos tem o direito de participar de um meio de ensino que lhe possibilite desenvolvimento adequado e, sobretudo, de um ensino de qualidade. No caso dos surdos, onde o processo de aprendizagem é baseado quase que exclusivamente em um sistema de informação composto por imagens, e devido às peculiaridades e diferenças entre a cultura surda e ouvinte, muita vezes a interação de indivíduos surdos com os demais integrantes de seu meio não traz contribuição cognitiva para o surdo.

Porém o ensino pela imagem, fonte de informação utilizada pela pessoa surda, deve ser apoiado nos conhecimentos ou experiências já vivenciadas pelo aluno surdo, pois o ensino baseado em imagens só fará sentido se o conceito referente à mesma for previamente compreendido e memorizado pelo indivíduo, como apontado por Lencastre e Chaves (2003):

É importante que no ensino pela imagem esta deverá ser associada ao que o aluno já conhece no momento da aquisição, ou seja, para que uma informação ganhe sentido e possa ser utilizada efetivamente como auxílio no processo de aprendizagem está deve integrar-se no

que o aluno já sabe sobre o assunto. Só assim o papel da imagem pode ser reconhecido perante o processo de ensino-aprendizagem.

A partir da utilização de recursos tecnológicos adequados, como um jogo educativo o aprendizado pela imagem pode ser maximizado, pois o aluno surdo poderá se sentir mais à vontade e se divertir enquanto aprende. Um jogo, ou até mesmo mídias digitais como vídeos e músicas, podem trazer grandes benefícios para o processo de ensino e aprendizado de alunos com deficiência melhorando seu desempenho escolar e auxiliando no processo de interação com os demais integrantes do meio escolar.

A utilização de jogos educativos que possibilitem o uso de imagens, animações e legendas para maximizar o aprendizado de crianças surdas pode contribuir para o seu desenvolvimento, e principalmente, facilitar o processo de aprendizagem pela imagem, de forma mais agradável e atrativa.

1.2 Objetivos

Os objetivos desse trabalho podem ser divididos em duas frentes principais:

1. Prototipação do jogo proposto com base nas principais funcionalidades básicas elicitadas, detalhadas no Capítulo 4.
2. *Design* de uma interface que ofereça suporte a agregação de recursos, como imagens e vídeos em Libras, para auxiliar na interação do aluno surdo com o jogo.

1.3 Organização do Texto

Esse trabalho está organizado da seguinte maneira:

O Capítulo 2 explora de forma geral as peculiaridades existentes no processo de inserção do aluno com deficiência na Escola, discutindo as abordagens educacionais existentes para o ensino destes, destacando o movimento educacional inclusivo e as dificuldades de sua implantação, além do sistema especial de educação e a integração e inclusão do aluno com deficiência na escola convencional.

O Capítulo 3 ressalta alguns aspectos e dificuldades presente na educação de alunos surdos, principalmente as relacionadas à alfabetização, pois mesmo a Libras sendo reconhecida pela lei 10.436/2002, como forma legítima de comunicação e expressão no Brasil, as diferenças entre a mesma e língua portuguesa ainda hoje acarretam várias

dificuldades para o aprendizado, principalmente de crianças surdas. Este capítulo também discute o papel da imagem na educação de crianças surdas e a contribuição da tecnologia, principalmente dos jogos educacionais, no processo de ensino-aprendizagem dessas crianças.

O Capítulo 4 apresenta o jogo Conceitos, com ênfase no processo de concepção e desenvolvimento, sendo enquadradas aqui as tecnologias utilizadas e a arquitetura montada para sanar os requisitos do usuário surdo. Esse capítulo também descreve o processo de composição de imagens utilizado no jogo para criação dos personagens, detalhando assim algumas peculiaridades que possibilitam a montagem do personagem.

No Capítulo 5, são apresentados alguns aspectos que devem estar presentes na interface de um jogo educativo direcionado ao público surdo, além da apresentação de dois importantes diagramas que foram utilizados para elaboração da interface do jogo. Também nesse capítulo é realizado um comparativo entre a interface final desenvolvida e a proposta a partir do diagrama MOLIC, além do detalhamento das principais funcionalidades implementadas, tanto da parte 2D como da parte 3D desenvolvida.

Por fim, o Capítulo 6 traz algumas conclusões da pesquisa, e os trabalhos que ainda serão desenvolvidos. Sendo esse seguido pelo Apêndice A, referente às tecnologias utilizadas no desenvolvimento do jogo e que oferecem o suporte necessário para a execução do mesmo. Esse apêndice pode ser entendido como um pequeno tutorial sobre como configurar o ambiente de programação do jogo de forma a agregar as tecnologias utilizadas em uma arquitetura responsável pelo funcionamento do mesmo.

Capítulo 2

Educação Especial e Inclusiva

Conforme Ferreira (2006), no Brasil até 1960, a educação de alunos portadores de deficiência era realizada por meio do sistema de educação especial, que acontecia em instituições especializadas públicas ou privadas, ou seja, o aluno com deficiência era educado fora da escola convencional. Em 1978, os alunos que apresentavam um grau de deficiência reduzido eram chamados de alunos excepcionais e enviados às classes convencionais de ensino; já os alunos que apresentavam alguma deficiência mais severa recebiam sua educação em classes de ensino especial. Ainda conforme Ferreira (2006), na década de 1990, um grande marco na história da inserção de aluno com deficiência no ensino convencional foi estipulado, pois, nessa década, vários eventos internacionais referentes aos direitos educacionais de alunos com deficiência foram realizados, como a Conferência Mundial de Salamanca, que teve grande importância para o amadurecimento da ideia inclusiva, favorecendo a ampliação da participação de alunos com necessidades especiais no ensino convencional.

Segundo Ferreira (2006), atualmente grande parte dos alunos com deficiência ainda tem sua educação paralela ao ensino convencional, mas, diferentemente de alunos que vivenciaram a década de 1960, esses possuem o direito a educação assegurado e contam com uma diversidade de recursos e metodologias que favorecem seu aprendizado, além do amadurecimento dos movimentos inclusivos que buscam a implantação efetiva da educação inclusiva, que defende a inclusão desses alunos no ensino convencional.

Conforme definido na *Declaração de Salamanca* (UNESCO, 1994) a educação inclusiva pressupõe que todas as crianças e jovens com necessidades educativas especiais devem ter acesso às escolas regulares, sendo essas escolas entendidas como meios para combater as atitudes discriminatórias, construindo uma sociedade inclusiva e atingindo a educação para todos.

A diferença entre educação inclusiva e educação especial pode ser entendida a partir das características e peculiaridades de cada um desses modelos de ensino, pois, segundo Mantoan (2006), no modelo integrado de ensino, ou ensino especial, nem todos os alunos com deficiência podem frequentar turmas de ensino regular, pois há uma seleção prévia dos que estão aptos à inserção, sendo que as turmas recebem uma educação diferenciada e paralela ao sistema de ensino convencional. Já a educação inclusiva, de acordo com Gil (2005), refere-se a considerar a deficiência de uma criança ou de um jovem como mais uma das muitas características diferentes que os alunos podem ter, ou seja, há que se respeitar essa diferença e encontrar formas adequadas para transmitir o conhecimento e avaliar o aproveitamento de cada aluno.

Ainda de acordo com Correia (2006), no sistema inclusivo de educação se enquadra todo o aluno que, por algum motivo, não consegue alcançar seu desempenho máximo no meio escolar, ou ainda está privado dele por alguma razão sócio econômica. Dessa forma, a ideia de inclusão escolar é contraditória ao modelo aplicado ao aluno com deficiência do sistema educacional convencional, pois atualmente indivíduos com necessidades especiais recebem sua educação a partir de um sistema integrado, ou especial, sendo esse um meio de ensino paralelo ao sistema convencional.

A grande diferença entre inclusão e integração está, portanto, na forma como a pessoa com deficiência é vista e tratada, pois na ótica do ensino especial o aluno com deficiência é integrado tendo que se adaptar aos métodos e paradigmas de ensino, sendo excluído dos demais. Já a visão inclusiva pressupõe uma participação plena do aluno em uma estrutura que se preocupa com o interesse, necessidades e direito de todos os indivíduos participantes do meio educacional e, principalmente, elimina as diferenças entre os alunos, as quais não são tratadas pela educação especial.

Contudo, para incluir o aluno com deficiência no ensino convencional de forma efetiva, vários aspectos relacionados à escola devem ser repensados, pois como a inclusão tem por objetivo diminuir, ou ainda, eliminar as diferenças e necessidades entre os indivíduos pertencentes a um meio comum, uma série de recursos de apoio pedagógico deve ser empregada pela escola, além da reformulação da infraestrutura escolar para que o meio educacional fique mais acessível e adequado às necessidades de cada aluno.

Ao propor a inclusão do indivíduo com deficiência na escola uma série de mudanças deve ser realizada, e justamente são essas mudanças que inviabilizam a implantação da inclusão educacional, pois grande parte das escolas não possuem condições de implantá-las.

No meio educacional convencional a diversificação entre os alunos já acarreta grandes dificuldades ao docente, pois cada aluno possui diferentes capacidades cognitivas, que podem facilitar ou dificultar o processo de ensino e aprendizagem, exigindo certo grau de atenção e dedicação do professor. Ao inserir um aluno com deficiência em uma classe regular a diversidade se torna um fator agravante, caso a escola e o docente não estejam preparados para lidar com essa situação. A escola deveria oferecer recursos educacionais capazes de atender às dificuldades desse aluno, e o professor deveria estar apto a fazer o uso adequado de tais recursos.

A inclusão educacional passa a exigir então, para sua implantação, que novas metodologias de ensino e recursos pedagógicos sejam agregados, objetivando garantir o atendimento adequado a todos os alunos, independente de suas diferenças e necessidades. Além disso, mudanças devem ocorrer na escola, onde novos aparatos devem ser incorporados à infraestrutura escolar para tornar o meio escolar acessível a qualquer aluno e na formação do docente, que deve ser contextualizada de modo que o educador possa trabalhar com a diferença em sala de aula de forma adequada, além de prepará-lo para utilizar qualquer tipo de recurso educacional para maximizar o aprendizado do aluno.

Segundo Hack e Negri (2008), o professor da escola inclusiva deve saber lidar criticamente com todo e qualquer aparato educacional utilizando-o pedagogicamente para prover o ensino de qualidade ao aluno, ou seja, para atender as necessidades de todas as partes em sala, o docente deve contar com tecnologias voltadas a educação, como recursos visuais e sonoros, aulas dinâmicas entre outros, além do uso de ferramentas ou aplicações que facilitem a comunicação entre o aluno e professor. Portanto para os autores, o docente presente no ensino inclusivo atuará tendo que aprender a utilizar qualquer mídia no processo de ensino e aprendizagem, deixando de ter um papel de mediador que apenas repassa informações, assumindo um papel mais abrangente, baseando suas aulas no diálogo com aluno.

Todos os fatores citados não viabilizam hoje o modelo educacional inclusivo, pois é muito complicado exigir que o docente esteja preparado a utilizar todos os recursos pedagógicos existentes para sanar as dificuldades, muitas vezes específicas, de um determinado aluno, sem falar no fato de que a todo o momento surgem novas tecnologias e aparatos educacionais

direcionados ao ensino de alunos com deficiência. Também não há como exigir que as escolas tenham recursos de acessibilidade em sua estrutura, uma vez que isso envolve questões financeiras.

Mesmo ao pensar na escola ideal, constituída pelos mais preparados professores, com a melhor infraestrutura e com todos os recursos tecnológicos disponíveis, não é possível garantir o desempenho máximo do aluno, pois o processo de ensino e aprendizagem está relacionado, além do meio, ao sujeito, ou seja, se o aluno não quiser aprender, tenha algum problema pessoal externo à escola, ou ainda, se não estiver se sentindo à vontade no meio educacional, nem a situação ideal poderá garantir seu aprendizado. Sendo assim, a inclusão vai muito além do meio escolar, uma vez que está totalmente ligada ao modo como o próprio indivíduo com deficiência se vê e se sente em relação à escola, que deve fazer o possível para facilitar sua inclusão, mas não pode forçá-lo a se incluir.

2.1 Tecnologia como Auxílio à Educação

O uso de tecnologias de apoio educacional deve ser incentivado e efetivamente realizado nas escolas tanto nas pautadas pelo ensino convencional como no ensino especial, pois contribuem de forma significativa para o desenvolvimento do alunado.

Apesar da resistência da utilização de recursos tecnológicos direcionados a educação por parte da grande maioria das escolas, seja por falta de recursos financeiros, ou por despreparo do próprio professor, que por vezes desconhece essas ferramentas de ensino ou não se sente apto a uso, esta deve ser incentivada, pois a tecnologia empregada em sala de aula como recursos pedagógicos pode maximizar as capacidades cognitivas de alunos com deficiência.

Entre os recursos que podem garantir uma melhoria significativa no processo de ensino-aprendizagem, além do desenvolvimento social e auxílio na execução de tarefas cotidianas, podem ser citadas as Tecnologias de Informação e Comunicação (TIC), utilizadas de forma ampla oferecendo apoio a todo e qualquer indivíduo, além compreender os recursos computacionais capazes de facilitar ou contribuir para a qualidade de vida de um determinado indivíduo. Para auxiliar na qualidade de vida do indivíduo com deficiência as Tecnologias Assistivas (TA), ramificação das TIC, mais especificamente recursos utilizados no auxílio a indivíduos com deficiência e idosos com o objetivo de facilitar a execução de algumas tarefas do cotidiano tanto social como educacional do usuário, podem ser utilizadas.

Para o Comitê de Ajudas Técnicas (CAT), instituído pela portaria nº 142 de 16 de novembro de 2006, Tecnologia Assistiva pode ser definida da seguinte maneira:

Tecnologia Assistiva é uma área do conhecimento, de característica interdisciplinar, que engloba produtos, recursos, metodologias, estratégias, práticas e serviços que objetivam promover a funcionalidade, relacionada à atividade e participação, de pessoas com deficiência, incapacidades ou mobilidade reduzida, visando sua autonomia, independência, qualidade de vida e inclusão social.

Ainda, para Carbonero, *et al.* (2011), as TA correspondem a uma ampla variedade de serviços e equipamentos aplicados para minimizar os déficits funcionais de pessoas com deficiência, ou seja, podem ser utilizadas como auxílio por indivíduos portadores de necessidades especiais.

A adoção TA no meio educacional pode contribuir no desenvolvimento dos alunos, visto que seu uso pode sanar ou minimizar dificuldades provindas da deficiência, além de contribuir com a autonomia e independência do aluno. Contudo, é importante ressaltar que as TA podem ser aplicadas de forma positiva não apenas na escola, mas também no dia-a-dia fora dela, contribuindo na melhoria da qualidade de vida do indivíduo com deficiência.

A utilização do recurso de apoio educacional quando relacionado ao ensino, deve ter seu uso planejado e bem pensado, pois a tecnologia por si só não irá contribuir para o desenvolvimento educacional do aluno, ou ainda, não irá agregar tanto valor a educação dos alunos caso não haja uma metodologia de apoio que maximize os resultados de seu uso frente ao aproveitamento do aluno, pois todo e qualquer aparato tecnológico direcionado ao auxílio de indivíduos com deficiência deve ser usado junto a uma metodologia de ensino empregada por um professor, e não como uma solução milagrosa capaz de eliminar todas as dificuldades e limitações educacionais de um determinado indivíduo.

Dessa forma o recurso educacional deve ser utilizado com a devida atenção para que não seja abusivo ou inadequado. Por meio da utilização correta no processo educacional do aluno com deficiência pode trazer vários benefícios para o mesmo, principalmente em relação às capacidades cognitivas do mesmo. No caso do aluno surdo, recursos como vídeos, imagens, animações, entre outros, além de favorecer o aprendizado, podem minimizar as diferenças existentes nos meios de ensino referente ao porte da deficiência.

A Tabela 2.1 indica alguns recursos que podem ser utilizados com apoio no cotidiano de pessoas com deficiência.

Tabela 2.1: Deficiências e alternativas de auxílio tecnológico, adaptado de (BAQUETA; BOSCARIOLI; BIDARRA, 2010).

Deficiência	Recurso
Deficiência Visual (Cegueira)	As ferramentas utilizadas são baseadas no sistema de leitura em Braille ou de âmbito computacional, se apropriando de recursos que possibilitem a tradução para informações sonoras, como leitores de tela e sintetizadores de voz. Além da tradução de informação do computador para o Braille, com a ajuda das impressoras Braille.
Deficiência Visual (Baixa Visão)	O indivíduo com baixa visão pode fazer uso de qualquer ferramenta usada pela pessoa cega, ou ainda se utilizar de recursos desenvolvidos especificamente para suas próprias limitações, os quais proporcionam a aplicação funcional da visão restante, como ampliadores de tela, ampliadores de imagem e óculos binoculares.
Deficiência Física	Existem muitas tecnologias assistivas que podem ser usadas como suporte ou apoio para essa deficiência, dentre as quais podem ser citadas, vocalizadores, software especiais com pranchas dinâmicas, programas de síntese de voz, teclados virtuais e ferramentas de locomoção.
Deficiência Intelectual	Dado que a grande dificuldade está no processo de aprendizagem, existem alguns estudos que exploram a influência de jogos educativos no desenvolvimento do indivíduo com deficiência intelectual.
Surdez	Há softwares que possibilitam ou facilitam a interação de pessoas surdas com ambientes virtuais a partir de vídeos, texto ou imagens, além de aparelhos auditivos analógicos e digitais, ou ainda, a adoção de um sistema de comunicação, como a língua de sinais e a leitura labial.

Capítulo 3

Surdez e Aspectos Cognitivos

Até esse momento o presente trabalho tratou os assuntos deficiência, educação e tecnologias de forma ampla e abrangente, servindo de base para a discussão da proposta do trabalho. Dessa forma a discussão que até então vinha sendo trabalhada de forma ampla terá seu escopo reduzido, sendo direcionada a surdez, mais especificamente às dificuldades e peculiaridades no processo de ensino e aprendizagem de alunos surdos e aos recursos que podem contribuir na maximização das capacidades cognitivas desses alunos, entre outros aspectos fundamentais presentes na educação de surdos. A partir do estudo sobre necessidades do usuário, o jogo poderá agregar recursos que facilitem a interação entre sistema e usuário e, conseqüentemente, será mais adequado às necessidades de aprendizado de uma criança surda.

No caso da surdez, a comunicação entre a pessoa surda e o mundo pode ser prejudicada, além de dificultar a recepção e interpretação de informações baseadas em interações com pessoas ouvintes. Os problemas de comunicação se refletem no processo educacional de alunos surdos, principalmente em relação à sua inserção na escola convencional, afetando seu desempenho escolar, pois, geralmente, a troca de informações entre o aluno surdo e os demais integrantes da escola se torna mais complexas devido a barreira imposta pela diferença entre a comunicação visual e oral.

Segundo Surdez (2007), o porte da surdez entre outros problemas pode ocasionar alterações de aprendizado, insatisfação e solidão, sendo classificada basicamente em dois tipos: (i) surdez de condução: menos frequente, afeta o ouvido externo ou médio e acontece quando as ondas sonoras não são bem conduzidas para o ouvido interno; (ii) surdez neurosensorial, que ocorre quando a cóclea que é o órgão interno da audição não consegue transformar a energia mecânica da vibração que o som produz em energia elétrica para transmiti-la ao cérebro que irá entender o som. Em certos casos a pessoa pode apresentar

problemas de condução neurossensorial e de condução juntos, sendo essa peculiaridade chamada de surdez mista.

No Brasil, devido à falta de acesso a língua portuguesa em sua modalidade oral, o sujeito surdo acaba sendo excluído, incompreendido ou não compreende os demais indivíduos de seu entorno e, conseqüentemente, em relação à sua educação, seu processo de aprendizagem é prejudicado por problemas relacionados à falta de comunicação.

A diferença imposta pela surdez, assim como as limitações causadas pela mesma estão vinculadas efetivamente ao uso da língua, pois em comunidades surdas que adotam a Libras como língua base os indivíduos participantes conseguem se comunicar e interagir entre si de forma “normal”, natural, justamente pelo uso de uma língua a qual todos estão familiarizados. Assim, como afirmado por Gesser (2009), a língua acaba tendo uma enorme influência nas relações estabelecidas entre os indivíduos de uma sociedade, pois a língua estreita as relações diminuindo as diferenças.

A comunicação, ou o direito de se expressar e interagir com as demais pessoas, ainda hoje é um grande problema enfrentado pelo indivíduo surdo, seja pela não compreensão da língua de sinais por meio dos ouvintes, ou pelo fato de grande parte dos surdos não conseguirem expressar suas vontades de desejos por meio da Libras aos ouvintes. O déficit de comunicação, que afeta de maneira negativa a relação entre ouvintes e surdos está relacionada principalmente ao choque cultural entre a cultura surda e a ouvinte, as quais possuem características próprias em relação à forma de pensar, de agir, de se relacionar e, principalmente, expor suas ideias.

A língua acaba então se tornando uma grande barreira que dificulta a inclusão do surdo na sociedade ouvinte, pois por um lado, no caso dos ouvintes, a língua é definida por meio de fontes sonoras, e no caso do surdo, o processo de comunicação é definido basicamente e exclusivamente por meio de fontes de informações visuais. As diferenças estabelecidas entre a língua portuguesa e a língua brasileira de sinais acabam dificultando a vida do indivíduo surdo, que em muitos casos acaba tendo que se adaptar a realidade ouvinte, deixando de lado sua própria cultura e se aproximando cada vez mais da cultura ouvinte no intuito de ser aceito socialmente.

Atualmente as comunidades surdas, em geral, adotam duas principais formas de comunicação: bilinguismo ou oralismo. Essas são abordagens totalmente distintas, pois o bilinguismo preserva a cultura e a identidade surda por meio do uso da língua de sinais e da

língua do país, enquanto o oralismo propõe que o surdo faça o uso da leitura labial com principal forma de comunicação, se assemelhando assim a um ouvinte.

Para Perlin e Strobel (2006), o bilinguismo é a forma mais adequada para o ensino de crianças surdas, pois nessa abordagem a Libras é considerado como a principal forma de comunicação, porém não a única, tanto é que só a partir do domínio da língua de sinais que o indivíduo surdo passa a aprender a segunda língua, o idioma do país, dando assim o devido valor à identidade surda. Já conforme Kubaski e Moraes (2009), o bilinguismo permite que, dada a relação entre o adulto e a criança surda, a mesma possa construir uma autoimagem positiva como sujeito surdo, sem perder a possibilidade de se integrar em uma comunidade de ouvintes.

Em relação ao oralismo, conforme Damázio (2007), a capacitação do surdo acontece por meio da leitura labial. Nessa abordagem o indivíduo surdo pode utilizar a língua da comunidade ouvinte na modalidade oral, como única possibilidade linguística, tanto na vida social como na escola, sendo essa abordagem mais aproximada da realidade ouvinte.

Apesar das duas abordagens de comunicação serem distintas e se basearem em princípios diferentes, ambas possuem vantagens e desvantagens em relação ao desenvolvimento de uma pessoa surda. A escolha pela metodologia de comunicação que será adotada na educação de uma criança surda é muito dependente da família e também da Escola. Contudo, o bilinguismo é visto como o principal e mais adequado método de educação de indivíduos surdos, como relatado por Perlin e Strobel, (2006), pois não nega a identidade e a cultura surda, além de adotar a língua de sinais como língua base.

O uso de Libras no processo de ensino de surdos contribui para melhorias e facilitação no aprendizado desses indivíduos, pois, de acordo com Gesser (2009), a língua de sinais adotada no bilinguismo é a língua natural dos surdos, o que favorece seu processo de comunicação e interação com o mundo a sua volta.

Com base na história e também a partir da análise dos métodos de comunicação de surdos aqui discutidos, pode-se inferir que a língua de sinais pode trazer melhorias em seu processo de aprendizagem e percepção de informações de mundo, não só pelo fato dessa língua ser baseada em imagens, fonte de informação a qual os indivíduos surdos estão habituados a utilizar desde muito cedo, mas também por ser a forma de comunicação mais natural para a comunidade surda, além da não resistência de uso por grande parte dos surdos.

3.1 A Educação do Surdo e o Uso de Imagens

Além do uso da língua de sinais na educação de pessoas surdas, pode-se também utilizar recursos complementares de apoio educacional, como o uso de imagens para auxiliar no processo de assimilação e fixação de conceitos de mundo referente a pessoas surdas. A imagem é um recurso adicional que amplia as capacidades do indivíduo surdo em relação à recepção de informação, visto que a imagem está presente na vida do indivíduo surdo desde muito cedo, seja no processo de construção de conhecimento ou na forma de comunicação através da língua de sinais.

O recurso visual pode contribuir juntamente com a língua de sinais na concepção de elementos e informações de mundo enriquecendo o processo de aprendizagem principalmente de crianças surdas, pois a fase de alfabetização, que geralmente acontece na infância, exige muito da criança em aspectos de memorização e assimilação de conceitos.

Na alfabetização a criança surda irá aprender a língua portuguesa em sua forma escrita, sendo que ao final desse processo a mesma terá condições de ler e escrever adequadamente textos em português, contudo, na prática o domínio da língua portuguesa não é tão trivial para um indivíduo surdo, podendo durar anos, pois, devido a inconsistências entre a língua portuguesa e a Libras, como a falta de significado para algumas palavras do português na língua de sinais, as crianças surda acabam não compreendendo certas informações apresentadas em forma de textos em português durante seu processo de alfabetização.

A alfabetização geralmente acaba se tornando um processo complexo e complicado para o surdo, e dessa forma é importante que nesse processo seja utilizado todo e qualquer recurso que maximize as capacidades cognitivas da criança.

Mesmo que a criança entenda e conheça a língua de sinais, o processo de alfabetização não se torna trivial, pois muitas palavras presentes no português não possuem tradução equivalente a Libras, o que dificulta o esclarecimento do conceito da palavra para a criança surda. Segundo Quadros (1997), um problema que deve ser reconhecido é que a escrita alfabética da língua portuguesa no Brasil não serve para representar a significação de conceitos elaborados em Libras, sendo essa uma língua visual espacial. Tanto é que um grafema, uma sílaba, uma palavra escrita no português não apresenta nenhuma analogia com um fonema, uma sílaba e uma palavra da língua de sinais, mas sim com o português em sua

forma sonora. Ao contrário da Libras, a língua portuguesa não é a língua natural da criança surda, sendo esse um grande percalço na educação da mesma.

Conforme Medeiros *et al.* (2005), as diferenças existentes na aprendizagem entre a criança surda e a ouvinte podem ser descrita da seguinte maneira:

As diferenças entre as oportunidades proporcionadas a uma criança ouvinte para aprender a ler e as que são oferecidas a uma criança surda são totalmente discrepantes. Uma criança ouvinte, que está iniciando sua aprendizagem da leitura e escrita, possui uma língua normalmente desenvolvida, pois se comunica sem restrições com o seu meio e tem um vocabulário amplo. Estas palavras e ideias que evocam ao serem reencontradas em textos, possuem total significado, porque a criança tem o conceito de referência. Dessa forma a criança ouvinte pode conversar sobre o que foi lido com outros leitores compartilhando experiências. O mesmo não acontece com a criança surda. Esta, via de regra, não dispõe da língua oral suficiente para a interação com o material escrito.

Assim, as capacidades cognitivas das crianças surdas devem ser respeitadas, e sempre que for necessário o professor deve lhe explicar certos conceitos esclarecendo as dúvidas e enganos referentes ao conceito representado pelas palavras da língua portuguesa. Além de expor o verdadeiro significado da palavra o professor deve, sempre que possível, ilustrar o conceito com a imagem, ou com a sequência destas. O uso da imagem juntamente com a Libras pode ajudar a criança a compreender o real significado da palavra do idioma português, transmitindo o conceito referente à mesma de forma adequada, sendo a imagem utilizada para memorização do significado, garantindo o aprendizado.

Tanto a língua de sinais, língua natural da comunidade surda, como a imagem como recurso educacional, devem estar presentes no decorrer da alfabetização e no decorrer da educação de crianças surdas, pois ambos favorecem o aprendizado. Porém, a criança deve sempre poder contar com a ajuda de um professor ou alguém que conheça Libras no intuito de auxiliar no esclarecimento de conceitos.

Como forma de minimizar os problemas deixados na vida da criança surda pela falta da informação sonora, deve-se adotar outras fontes de informações para a transmissão do conhecimento a indivíduos surdos, ou seja, fazem-se necessárias alternativas de interação por meio da Libras, legendas, animações, imagens, entre outros canais de comunicação (OHIRA, 2009).

Portanto, o uso de imagens no processo de aprendizado do aluno surdo deve ser bem explorado no desenvolvimento de uma tecnologia educativa, pois com o uso da imagem a

ferramenta acaba se tornando mais próxima da realidade do aluno surdo, uma vez que desde muito cedo este já está habituado a utilizar como forma de interpretação de mundo um sistema de comunicação baseado em imagens, além de tornar o uso da ferramenta mais agradável e intuitivo, deixando o aluno mais a vontade ao utilizar algo mais próximo de sua realidade.

3.2 Tecnologias e a influência dos Jogos

Durante seu processo educacional o aluno surdo acaba enfrentando dificuldades devido às várias diferenças existentes entre sua cultura e a ouvinte, sejam essas relacionadas a alfabetização, utilização da língua de sinais ou, até mesmo, a interpretação de conceitos. Geralmente as dificuldades são ocasionadas pela falta de adequação do meio às necessidades do indivíduo surdo, ou seja, o surdo está inserido em uma sociedade ouvinte, onde o principal meio de comunicação é o oral, o que influencia de forma significativa nas relações de seus integrantes e, conseqüentemente, dificultando o cotidiano da pessoa surda, tanto em relação a educação quanto no âmbito social.

O contexto social pode afetar drasticamente o processo educacional de pessoas surdas, que deve assimilar informações que para um ouvinte possui total significado devido à representação sonora das mesmas, contudo para o surdo essas podem não representar nenhum significado especial por conta das limitações da representação visual. Para sanar ou minimizar os problemas relacionados à concepção de informações acarretada pela ausência da fonte sonora é interessante que durante o processo educacional de alunos surdos sejam utilizadas tecnologias de apoio educacionais, sejam convencionais como imagens, murais de fotografias, tematização de ambientes, entre outros, ou ainda, ferramentas computacionais de apoio pedagógico, como a internet e jogos de computador.

A tecnologia ou o recurso de apoio pedagógico pode oferecer várias contribuições para a vida do sujeito com deficiência. No caso da surdez a tecnologia pode ser utilizada para maximizar os resultados obtidos no processo educacional de indivíduos surdos, em especial, o ensino da língua portuguesa, Libras e conseqüentemente, a concepção de conceitos. As tecnologias de cunho computacional, ou tecnologias digitais, são frequentemente utilizadas para auxiliar o processo cognitivo de pessoas surdas, na maioria das vezes, no ensino de Libras e da língua portuguesa. Esse tipo de recurso é geralmente bem atrativo e agradável, principalmente quando utilizado no processo educacional de crianças surdas, além de ser

facilmente encontrado na internet. A Tabela 3.1 apresenta algumas tecnologias desenvolvidas para surdos e Tabela 3.2 exemplifica essas tecnologias por meio de imagens.

Tabela 3.1: Tecnologias direcionadas a pessoas surdas. Adaptada de Baqueta; Boscaroli (2011)

Tecnologia	Descrição
Vídeo Chamada	Possibilita ao surdo se comunicar por vídeo, de qualquer lugar, com um intérprete da Associação Portuguesa de Surdos (http://www.apsurdos.pt/), para a solicitação de serviços, como um táxi ou marcar uma consulta médica (DEFICIENTE.NET, 2011).
Programas que Utilizam Webcam	Programas de comunicação on-line como OOVOO, MSN, SKYPE, CAMFROG, entre outros, possibilitam comunicação em Libras pelo uso de <i>Webcam</i> e, por meio da seção de bate-papo on-line (JÚNIOR, 2010).
Rybena	As pessoas surdas podem se comunicar em Libras por animações de imagens no celular, Figura 3.1. Ouvintes podem enviar textos em português aos surdos, que receberão a mensagem em Libras (RYBENÁ, 2011).
Babá Eletrônica	Possui estímulo vibratório para quando o bebê começar a chorar. O aparelho pode ser guardado no bolso pelos pais e quando começa a vibrar indica que o bebê está chorando (JÚNIOR, 2010).

Tabela 3.2: Exemplos visuais de tecnologias para surdos, baseada em Junior (2010)

Tecnologia	Exemplos
Vídeo Chamada	
Programas que Utilizam Webcam (OOVVOO)	
Rybená	
Babá Eletrônica	

De acordo Arcoverde (2006), as tecnologias digitais conquistam cada vez mais usuários, contribuindo para a instauração de interações sociais cada vez mais amplas e permitindo, por meio das ferramentas de comunicação, principalmente pelo uso do computador (e-mail, chat,

lista de discussão), uma multiplicidade de dinâmicas linguístico-discursivas que possibilitam o uso e apropriação da língua, tanto a portuguesa como a Libras.

Ainda segundo Arcoverde (2006), as oportunidades de comunicação oferecidas pelas tecnologias digitais, principalmente as vinculadas à internet, permitem novas possibilidades de interação e aprendizado com muitos outros indivíduos, favorecendo a troca de informações e agregação de conhecimento através da comunicação entre seus usuários, os quais compartilham seus conhecimentos e constituem a imensa diversidade que institui a sociedade em rede.

Assim, pode-se dizer que a tecnologia digital favorece o aprendizado por meio de um espaço amplo e abrangente onde os indivíduos podem trocar suas experiências e aprender de forma coletiva, sendo essa uma abordagem educacional apropriada para o indivíduo surdo, pois o mesmo além de se comunicar com indivíduos ouvintes ou surdos, pode fazer o uso da tecnologia para praticar sua escrita, ou ainda, aprender sobre novos assuntos ampliando assim seu conhecimento e aprimorando sua capacidade de comunicação.

Outro recurso que pode ser utilizado como importante ferramenta de auxílio ao aprendizado são os jogos computacionais de apoio pedagógico, pois em sua grande maioria enfatizam o uso de imagens e animações no processo de interação, além de apresentarem as informações de forma atrativa e agradável para o aluno.

Segundo Taurouco *et al.* (2004), os jogos podem ser ferramentas instrucionais eficientes, pois divertem enquanto motivam, facilitando assim o aprendizado e aumentando a capacidade de retenção do que foi ensinado, exercitando as funções mentais e intelectuais do jogador.

O jogo educativo aplicado no processo educacional de alunos surdos deve respeitar as limitações dos mesmos, ou seja, esse deve ser adequado ao nível educacional e experiências vivenciadas pelo aluno surdo, dessa forma o jogo como recurso educacional oferecerá efetivamente o apoio necessário para o aluno surdo.

É importante ressaltar que o jogo direcionado a criança surda não necessariamente precisa apresentar, em sua interação com o aluno, recursos em Libras, dado que a partir da ajuda de um educador a criança pode fazer o uso de grande parte dos jogos educacionais disponíveis, pois por meio da ajuda de um profissional especializado na área da surdez a criança pode recorrer ao mesmo em caso de dúvidas no decorrer da utilização do jogo.

Os jogos Multi-trilhas e Libras Brincando e Aprendendo são exemplos de softwares que podem ser utilizados no processo educacional de pessoas surdas, mais especificamente no ensino de crianças. Segundo Couto (2008), o Multi-trilhas (Figura 3.1) tem por objetivo auxiliar crianças surdas no processo inicial de aquisição do português escrito como segunda língua, por meio de uma abordagem multidisciplinar, com foco no *design* e com base na abordagem bilíngue para educação de indivíduos surdos. Já conforme Librasnet (2012), o software Libras Brincando e Aprendendo (Figura 3.2) tem por objetivo evidenciar conceitos ligados ao aprendizado escolar como matemática, ciências, geografia e português. Esse software apresenta várias categorias, onde o aluno pode escolher uma. Cada categoria apresenta um ou mais cenários que abordam atividade sobre o tema escolhido. Certas categorias são representadas na forma de aulas, que objetivam ensinar ao aluno o significado e o conceito de imagens.



Figura 3.1: Jogo Multi-trilhas



Figura 3.2: Jogo Libras Brincando e Aprendendo

Existem vários outros trabalhos que propõem jogos para auxiliar no processo de ensino-aprendizagem, não necessariamente projetos para usuários com algum tipo de deficiência. Os abaixo relacionados são de interesse no escopo deste trabalho por terem recursos 3D.

Apresentado por Leitão *et al.* (2011), o jogo Colisões (Figura 3.3), foi desenvolvido a partir de uma abordagem experimental, sendo um ambiente 3D desenvolvido a partir da *engine* e do modelador Blender, em que o aluno é desafiado a realizar “experimentos virtuais” em busca da compreensão dos conceitos fundamentais relacionados à disciplina de física como velocidade, energia e momento. Nesse jogo o aluno é confrontado com desafios que abordam o conceito de velocidade e sua determinação experimental, energia e suas transformações, leis de conservação, para então analisar os processos de choque, elásticos e inelásticos, em uma e duas dimensões. A aplicação do jogo em ambientes reais de

aprendizagem está em andamento, bem como sua implementação em ambiente *web* através datecnologia X3DOM, que conforme X3Dom (2012) é um *framework* de código aberto capaz de reproduzir jogos via internet.

Outro jogo que possui a característica de auxiliar o aprendizado do usuário é o Kinble, Figura 3.4, que segundo Figueiredo *et al.* (2011) é um ambiente educativo que visa promover conhecimento aos estudantes nas matérias lecionadas no ensino médio. Neste, o jogador interage com o ambiente, uma ilha, por meio de um robô chamado “Kinble”, que deve reunir o conhecimento necessário para sobreviver a partir de questões que lhe são submetidas.

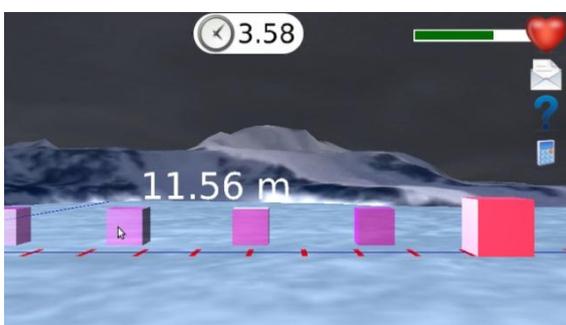


Figura 3.3: Jogo Colisões



Figura 3.4: Jogo Kinble

Tanto o Colisões como o Kinble são desenvolvidos por meio de uma abordagem 3D, o que facilita e torna a interação com o jogador mais interessante, pois jogos tridimensionais apresentam as informações de uma forma mais fiel a realidade do usuário. Ambos os jogos podem ser utilizados como recursos no processo de ensino, porém, apenas com o público-alvo para o qual foram desenvolvidos, pois devido às peculiaridades dos usuários, certos pontos da interface do jogo podem se tornar críticos para a interação para um usuário que fuja ao perfil almejado. O jogo Conceitos a ser na sequência apresentado também agrega em sua composição recursos 3D visando facilitar a percepção da criança surda em relação aos objetivos da interação.

Capítulo 4

Proposta do Jogo Conceitos

Independente do fim o para qual um jogo é desenvolvido, este deve levar em consideração as habilidades do jogador, pois os jogos são totalmente baseados na interação e na comunicação com o usuário. A qualidade da interação entre o jogo e jogador é crucial para o aprendizado e tomada de decisão do usuário frente a uma situação imposta pelo jogo.

Quando um jogo é direcionado à educação, a comunicação ou a troca de mensagens entre o sistema e o usuário deve ser tomada como fator primordial no desenvolvimento, uma vez que o jogo será utilizado como objeto de aprendizagem, contribuindo para o desenvolvimento educacional de seu usuário. Logo, de acordo com as características do usuário, o jogo deve propor recursos e artefatos que maximizem a recepção de informações frente ao sistema.

Esse capítulo tem como objetivo apresentar o jogo Conceitos, em sua concepção e detalhes de implementação. Este jogo é direcionado a crianças surdas, com o propósito contribuir ao ensino da língua brasileira de sinais e do português, além de auxiliar no processo de concepção e interpretação da informação pelo usuário.

4.1 Desenvolvimento de Jogos

Jogos são desenvolvidos sobre uma abordagem bidimensional (2D) ou tridimensional (3D), e oferecem uma série de recursos e possibilidades relacionadas a aprendizagem e comunicação com o usuário. No entanto, o jogo 3D, de forma geral, torna-se mais atrativo e agradável ao usuário, pois apresenta um universo de interação semelhante ou próximo da realidade do jogador. A partir da interação com cenários e elementos de um jogo tridimensionais o jogador tem a impressão de estar inserido no espaço virtual, o que torna esse tipo de jogo o melhor aceito no mercado atual, apesar de serem mais complexos de desenvolver.

Conforme Clua e Bittencourt (2005), um jogo 3D é um software especial que contém elementos muito variados, como módulos de Computação Gráfica, Inteligência Artificial, Redes de Computadores, Multimídia, entre outros. Estes módulos devem funcionar em perfeita harmonia, garantindo a principal característica de um jogo: ser um software em tempo real. Para que isto seja possível, é necessário explorar ao máximo o *hardware* dedicado e as conhecidas placas gráficas aceleradoras 3D. Para este propósito é fundamental que o jogo esteja baseado sobre diversas API ou bibliotecas, como a OpenGL ou o DirectX, bibliotecas gráficas de baixo nível que oferecem suporte para vários cálculos matemáticos exigidos por um jogo 3D.

Em relação ao uso de jogos 3D no processo educacional de crianças, o mesmo pode agregar muitas vantagens, pois acaba prendendo a atenção do aluno de forma potencial, ou seja, o aluno acaba se sentindo um personagem do jogo e, conseqüentemente, se esforça mais para cumprir seu objetivo.

De acordo com Barcelos (2008), ambientes virtuais 3D obtêm maiores resultados em relação ao processo educacional, pois proporciona a criação de ambientes mais lúdicos. Esses tipos de jogos elevam o grau de envolvimento e imersão dos alunos com o conteúdo estudado, com outros usuários e com o próprio professor. A agregação de recursos como a realidade virtual 3D, áudio, vídeo, entre outros presentes em jogos tridimensionais podem auxiliar no processo educacional motivando os usuários.

Já os jogos em 2D, conforme Silva *et al.* (2009), são puramente bidimensionais com uma única camada, ou seja, a cena se compõe sobre um universo formado por duas dimensões, largura e altura, que determinam uma mapa de *bits*. É importante ressaltar que esta possui menor custo, complexidade de desenvolvimento e processamento em relação à tecnologia 3D, podendo ser bem aproveitada em aplicações com objetivos voltados para um determinado tipo de público e interesse, sendo muito utilizada atualmente em aparelhos celulares ou assistentes digitais pessoais.

Independentemente da abordagem utilizada para desenvolvimento de jogos, os mesmos são importantes e interessantes recursos que podem auxiliar as capacidades cognitivas do indivíduo, tornando o processo educacional mais atrativo.

4.2 O Jogo Conceitos

Conceitos é um jogo educativo, em fase de desenvolvimento, que poderá ser utilizado como um aparato de auxílio educacional no processo de ensino e aprendizado de crianças surdas, pois proporciona, em sua interação com usuário, recursos que podem ser aproveitados para ensino da língua portuguesa, no ensino da Libras e, principalmente, na fixação e memorização de conceitos do cotidiano.

O jogo é direcionado para crianças de 1^a a 5^a séries, pois segundo professoras do Centro de Formação de Profissionais da Educação e de Atendimento às Pessoas com Surdez (CAS), a criança surda responde de forma muito positiva aos recursos dessa natureza, ou seja, todo o tipo de recurso, principalmente imagens e vídeos, contribuem de forma muito significativa para o desenvolvimento educacional de alunos com essa faixa-etária de idade, haja vista que nessa fase as crianças surdas são alfabetizadas e nesse processo a criança acaba apresentando várias dificuldades de assimilação de informação, devido às diferenças e disparidades entre a língua portuguesa e Libras.

Conceitos é um jogo baseado em ambientes virtuais semelhantes aos ambientes frequentados pelos usuários em seu cotidiano, como uma sala de aula, uma cozinha, quarto, entre outros. Cada ambiente é composto por uma coleção de objetos, como uma mesa, cadeira, geladeira, etc. Esses objetos são representados por meio de imagens, legendas e vídeos em Libras. O objetivo do jogo é que dado um objeto em um determinado ambiente, o aluno deverá encontrá-lo em meio a todos os objetos desse ambiente até atingir uma pontuação mínima necessária, definida por uma quantidade mínima de objetos que devem ser encontrados para o aluno finalizar a interação.

A interação no jogo é baseada no uso de imagens, vídeos, textos e objetos tridimensionais, sendo dividida em três fases distintas, onde a criação do perfil, ou cadastro do aluno, e o acesso a jogo, são realizadas sobre uma abordagem 2D e a interação com o ambiente virtual acontece a partir do uso da tecnologia 3D.

A fase de cadastro acontece quando o aluno acessa o sistema pela primeira vez, no qual realiza duas operações distintas, o cadastro de um novo usuário no sistema e depois, o acesso para utilização das funcionalidades do jogo. No processo de cadastro o aluno criará uma conta, constituída por um *login*, nome e sexo. Em cada conta será expresso um status que indica quanto do jogo já foi concluído pelo usuário, sendo que quanto mais ambientes o aluno

completar maior será seu status. É importante ressaltar que o jogo deve apenas requisitar informações básicas ao usuário, pois o excesso de informações pode prejudicar a interpretação da criança surda sobre os dados requisitados pelo jogo, devido à poluição visual.

Ainda no processo de cadastro, o aluno deve criar o seu personagem, ou seja, um personagem criado de acordo com as preferências do usuário, a ser utilizado na interação com os ambientes. Cada personagem está vinculado a uma conta específica, além de possuir um nome, sexo e uma imagem para sua representação. A imagem do personagem será constituída por sub-imagens que representam partes do corpo como rosto, cabelo, corpo e roupa. As sub-imagens podem ser alteradas de acordo com a preferência do aluno, assim o personagem poderá ter seu cabelo, olho, rosto e roupas alterados, sendo que cada parte do corpo do personagem poderá assumir um modelo e uma cor específica, por exemplo, cabelo preto cacheado e um cabelo loiro liso.

Ao final da fase de cadastro o aluno poderá acessar as funcionalidades do jogo, escolhendo um ambiente e iniciando uma partida. Tal fato acontece de forma semelhante à fase de acesso, onde o aluno que já possui uma conta a partir de seu *login* acessa as funcionalidades de jogo dando início a interação com os ambientes. Cada ambiente possui um status referente ao aproveitamento do aluno em relação ao mesmo; esse status está ligado diretamente à quantidade de objetos encontrados pelo aluno no ambiente, ou seja, se o aluno encontrou todos os objetos que deveriam ser encontrados o ambiente terá o status de completo.

Em cada ambiente encontram-se objetos tridimensionais, sendo que cada objeto possui uma coleção de imagens referente a si, uma textura, posição topológica, um vídeo representando o significado do objeto em Libras, uma descrição em português também referente ao significado e uma imagem do material que compõe o mesmo, dando ao aluno a opção de navegação entre os objetos, sendo que a partir do objeto qualquer o aluno poderá acessar o material que o compõe.

Cada ambiente terá associado a si um determinado grau de dificuldade, referente a uma quantidade mínima de objetos que deve ser encontrada; quanto maior a quantidade de objetos no ambiente maior a sua dificuldade. Durante o jogo o aluno poderá acessar vários ambientes, contudo deve respeitar uma ordem de dificuldade, ou seja, o usuário só poderá utilizar ambientes de dificuldade mais elevada caso já tenha completado os de nível inferior, ou ainda a fase anterior.

A interação com os ambientes e objetos tridimensionais, ou ainda a comunicação entre o aluno e o ambiente virtual, é realizada com auxílio de um personagem. Esse personagem irá, no decorrer da interação, por meio de Libras, textos em português ou ainda imagens, pedir para que o aluno encontre um objeto presente no ambiente, cabendo ao aluno interpretar o pedido e buscar o objeto requisitado.

Ao término de cada partida o aluno poderá visualizar o resultado da interação, sendo esse constituído pela quantidade de objetos encontrados. Caso o aluno tenha encontrado todos os objetos requisitados pelo personagem o ambiente receberá o estado de completo e um novo ambiente será liberado para o aluno. Dessa forma, acredita-se que o sistema acabará contribuindo para o ensino da Libras e do português, além de auxiliar no processo de assimilação e fixação do conceito referente ao objeto requisitado pelo personagem dada a sua representação gráfica.

4.3 Análise e Elicitação de Requisitos

Para concepção do jogo Conceitos foram realizadas reuniões com professoras do CAS, que motivaram o desenvolvimento de um ambiente de interação aluno-computador para auxiliar o aluno surdo, em especial crianças, no processo de fixação de conceitos, a partir do uso de imagens presentes em ambientes de seu cotidiano.

O principal problema apontado nas reuniões realizadas refere-se à assimilação de conceitos abstratos e de objetos físicos do meio em que os alunos estão inseridos, uma vez que devido à falta das informações sonoras os alunos não conseguem assimilar o conceito com a imagem ou palavra que o representa de forma adequada. Assim, os requisitos coletados sugeriram uma ferramenta capaz de minimizar os potenciais problemas existentes na alfabetização de crianças surdas, visto que é nessa fase que os alunos aprendem a língua portuguesa, sendo exatamente nesse ponto onde os problemas de conceituação são mais frequentes.

Assim, para suprir as necessidades apontadas pelos professores entrevistados e a partir da elicitação de requisitos funcionais e não funcionais, pensou-se no desenvolvimento de um jogo educativo capaz de auxiliar no processo de aprendizagem da criança surda. O sistema está sendo desenvolvido para ambiente *desktop* com base na linguagem de programação Java, visando a portabilidade do mesmo, sendo esse requisito de extrema importância devido aos recursos oferecidos pelo CAS, além do fato de que esse sistema futuramente poderá ser

implantado em outras instituições de ensino. Os requisitos licitados por meio das reuniões realizadas com as professoras do CAS estão expressos na Tabela 4.1:

Tabela 4.1: Requisitos Funcionais do jogo Conceitos

[RF-01] Criar Conta no Sistema
Descrição: O sistema deverá prover uma opção onde o usuário, aluno, possa criar seu cadastro no mesmo. Sendo o cadastro constituído de poucas informações, entre as quais deve haver um campo para nome e outro para o <i>login</i> . Segundo o cliente as crianças surdas possuem dificuldades em interagir com muitas informações, sendo assim é interessante que o sistema não ofereça uma grande quantidade de informação em uma única tela.
Prioridade: Alta
Solicitante: Aluno
[RF-02] Criar Personagem
Descrição: O sistema deverá prover meios para que o aluno consiga criar um personagem 2D de acordo com sua preferência, possibilitando a edição de certas partes do corpo do personagem, como cabelo, roupa, e rosto. Deve também haver um campo para que o aluno escolha o nome de seu personagem. Esse personagem será responsável pela interação aluno e sistema. Contudo depois de criado não será possível editá-lo.
Prioridade: Alta
Solicitante: Aluno
[RF-03] Escolher Ambiente
Descrição: O sistema deve possibilitar vários ambientes para a partida fornecendo meios para que o aluno consiga escolher o ambiente desejado. Cada ambiente deve ser inicializado como incompleto e quando o aluno terminar o ambiente, o sistema deve mudar sua condição para completo.
Prioridade: Alta
Solicitante: Aluno
[RF-04] Jogar
Descrição: O sistema deve possibilitar meios para que o aluno consiga realizar o que seu personagem pede, como um cursor ou um objeto de seleção capaz de indicar objetos dispersos no ambiente, além de possibilitar que o aluno interaja com o ambiente e os

<p>objetos 3D, sendo que a partir do pedido do personagem (busca de um objeto) o aluno deverá encontrar o mesmo no ambiente acumulando pontos até atingir a pontuação mínima exigida para completar o ambiente. A quantidade de ambientes completos influencia na porcentagem total do jogo, pois ao acessar o jogo o aluno visualiza a porcentagem de conclusão do mesmo, sendo essa baseada na quantidade de ambientes completos pelo aluno.</p>
<p>Prioridade: Alta.</p>
<p>Solicitante: Aluno</p>
<p>[RF-05] Visualizar Objetos</p>
<p>Descrição: O sistema deve prover meios para que o aluno consiga visualizar e interagir com um objeto específico, fornecendo opções para rotacionar, editar texturas e visualizar a palavra correspondente ao mesmo, sendo essa visualização realizada apoiada por meio de Libras e/ou português.</p>
<p>Prioridade: Média</p>
<p>Solicitante: Aluno</p>
<p>[RF-06] Salvar o Jogo</p>
<p>Descrição: O sistema deve possibilitar que os avanços do usuário no decorrer do jogo sejam salvos quando esse solicitar, mesmo que a partida não tenha acabado. Assim quando o aluno acessar novamente o ambiente sua pontuação irá continuar do ponto de parada e não será reiniciada.</p>
<p>Prioridade: Média</p>
<p>Solicitante: Aluno</p>
<p>[RF-07] Visualizar Resultados</p>
<p>Descrição: Ao fim de cada partida o sistema deve fornecer a opção para visualização do resultado da partida, sendo esse calculado pela quantidade de objetos encontrados pelo usuário. Nessa opção deve ser exibido em tela o resultado (derrota ou vitória), pontuação, uma opção para reiniciar a partida e outra para Terminar partida.</p>
<p>Prioridade: Alta</p>
<p>Solicitante: Aluno</p>
<p>[RF-08] Terminar Partida</p>
<p>Descrição: O sistema deve apenas exibir essa opção quando o usuário requisitar a saída do jogo. Se essa for selecionada o sistema deve verificar se o aluno conseguiu completar o</p>

ambiente, se sim o sistema deve marcar o ambiente como completo e salvar o progresso do aluno de forma automática. Ao finalizar uma partida o aluno deverá ser direcionado ao menu inicial do jogo.
Prioridade: Alta
Solicitante: Aluno
[RF-09] Reiniciar Partida
Descrição: No caso do aluno não atingir o resultado mínimo do ambiente o sistema deve fornecer uma opção para reiniciar a partida no ambiente.
Prioridade: Média
Solicitante: Aluno
[RF-10] Abandonar Partida
Descrição: O sistema deve fornecer meios para que o aluno consiga abandonar a partida sem que essa esteja terminada. O sistema deve considerar como resultado dessa opção que o aluno não completou o ambiente.
Prioridade: Baixa
Solicitante: Aluno
[RF-12] Visualizar históricos de resultados
Descrição: O sistema deve prover alternativas para que o usuário consiga visualizar os resultados obtidos em outras interações.
Prioridade: Baixa
Solicitante: Aluno

4.4 O Processo de Desenvolvimento

O jogo Conceitos será dividido em duas partes distintas, uma parte 2D e outra 3D. Em relação a ambas as partes uma das exigências é a presença de alguma funcionalidade no jogo que facilite a comunicação via Libras. Dessa forma, para sanar essa exigência, a interação entre o usuário e o jogo será regida por uma série de animações e vídeos utilizados para representar conceitos em Libras.

As próximas subseções serão destinadas a especificação da arquitetura elaborada para o desenvolvimento do jogo Conceitos, além da descrição e fundamentação do processo de

composição de imagens utilizado durante a interação para construção do personagem. Em resumo, serão descritos processos e tecnologias que foram estudados e aplicados no desenvolvimento do jogo para garantir certas funcionalidades proposta na especificação de requisitos.

4.4.1 Arquitetura

A partir das necessidades elicitadas, na parte 2D os vídeos, tanto relacionados a tradução de informações para Libras como empregados no sistema de ajuda, são incorporados por meio do *Java Media Framework API (JMF)*, um *framework* que, segundo Deitel e Deitel (2010), possibilita a criação de aplicativos Java que reproduzem, editam, transmitem e capturam muitos tipos de mídias populares, incluindo os formatos de arquivo do tipo avi, swf, spl, mp3, mpeg, mpg, entre outros.

Porém, devido a alguns problemas de compatibilidade de arquivos de vídeos com JMF, está sendo utilizado também o *plugin* FOBS JMStudio ou fobs4, que potencializa o *framework*, oferecendo suporte a uma variedade maior de formatos, tanto de vídeo como de áudio, além de corrigir alguns problemas presentes na execução de vídeos no formato avi, formato esse utilizado no jogo. Já a parte 3D não possui nenhum recurso para execução vídeo devido às restrições da *engine* utilizada. Assim, estão sendo estudadas formas de compensar as limitações da *engine* por meio da reprodução de *gifs* animados e até mesmo, a possibilidade da criação de um *codec* de vídeo avi.

Para manipulação das informações presentes no jogo será utilizado o sistema gerenciador de banco de dados PostgreSQL, responsável por armazenar as informações da conta do aluno, do personagem, dos objetos presentes no jogo, entre outros. O banco de dados será a principal forma de comunicação entre os componentes do jogo, pois tanto a parte 3D como a 2D acessaram o banco de dados para verificar o *status* de conta, para carregar o personagem e os modelos presentes em cena, entre outros dados que serão utilizados durante a interação do jogo, além do acessos a dados confidenciais de cada usuário como o nome e o *login*.

O desenvolvimento das funcionalidades presentes na parte 3D do jogo é baseado na interação entre os modelos tridimensionais, sendo para isso utilizadas as seguintes tecnologias: o modelador Blender para criação de modelos 3D, a *engine* jMonkey responsável pela implementação de regras e eventos do jogo, e o *plugin* Ogre3D que possibilita exportar os objetos tridimensionais modelados em Blender para o jMonkey.

Para informações mais detalhadas sobre as tecnologias utilizadas no desenvolvimento do jogo o Apêndice A, que descreve o processo de configuração dessas tecnologias, pode ser consultado.

A arquitetura do jogo pode ser visualizada na Figura 4.1 que explica o relacionamento de todas as tecnologias utilizadas.

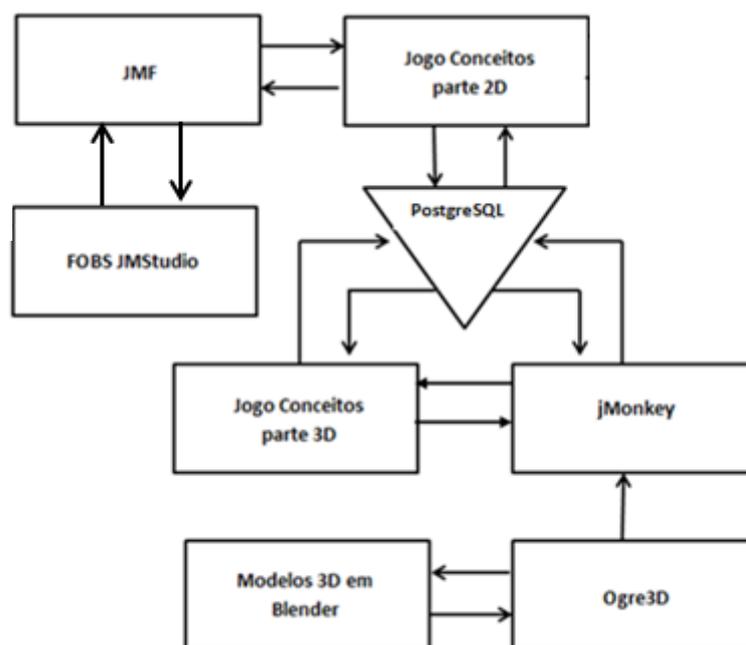


Figura 4.1: Representação da arquitetura do jogo Conceitos

Segundo Leitão *et al.* (2011), o Blender é um programa que tem por objetivo a criação de conteúdo 3D, livre e de código aberto, sob a licença GNU (*General Public License*). Por meio da utilização do Blender é possível realizar tarefas como a modelagem, animação, texturização, composição, renderização, edição de vídeo e criação de aplicações interativas em 3D. Ele também permite a criação de imagens, vídeos, jogos ou apresentações.

A engine jMonkey, de acordo com Jmonkey engine (2012) é um *framework Open Source* que oferece vários artifícios que facilitam o desenvolvimento de jogos otimizando a criação de aplicações Java que utilizam recursos 3D. Essa engine utiliza a biblioteca gráfica Open GL e LWJGL (*Lightweight Java Game Library*) para renderização de modelos do jogo, cujo funcionamento foi pensado para otimizar os recursos computacionais oferecidos durante a interação, uma vez que nessa engine uma cena é representada por uma estrutura hierárquica em árvore, onde um nodo pai, no caso a cena, pode possuir vários filhos, porém cada filho

possui apenas um ponteiro para seu pai. A estrutura de árvore garante que apenas os objetos necessários serão renderizados evitando desperdícios de processamento.

Conforme Clua e Bittencourt (2005), na área de jogos, durante a execução de um jogo a *engine* deve-se comunicar com o *hardware* gráfico, controlar os modelos para serem renderizados, tratar das entradas de dados do jogador, tratar de todo o processamento de baixo nível, entre outros detalhes que normalmente estão fora da responsabilidade do desenvolvedor.

O conjunto de tecnologias que define a arquitetura do jogo Conceitos possibilita o desenvolvimento das funcionalidades básicas elicitadas, além de deixar o jogo com um aspecto mais agradável e atrativo ao público alvo. A incorporação do JMF e do *plugin* FOBS JMStudio tornam a interação melhor adequada as limitações do usuário, e a partir uso da *engine* jMonkey, do modelador Blender e do *plugin* Ogre3D, é possível inserir e manipular os objetos e ambientes tridimensionais, contribuindo assim com a aprendizagem do usuário e tornando o jogo mais atrativo e lúdico.

4.4.2 Composição de Imagens

Para construção do personagem, funcionalidade presente na parte 2D do jogo, foi necessário realizar um estudo sobre composição de imagens e adotar um método para a realização de tal processo. Como previsto na elicitação de requisitos o usuário pode, a partir de um conjunto de imagens, montar seu personagem por composição de imagens.

As imagens utilizadas para composição estão no formato BMP devido à utilização de um *codec* BMP desenvolvido na disciplina de Processamento de Imagens Digitais. O uso desse *codec* facilita a manipulação de certas informações presentes em imagens com essa extensão, além do desenvolvimento da funcionalidade “criar personagem”.

Para compor uma imagem é necessário selecionar duas imagens distintas, uma definida como fundo e outra que é utilizada como frente. No processo de criação de personagens o conjunto de partes do corpo do personagem que pode ser editado pelo usuário é também o conjunto de imagens de frente, e a imagem de fundo é a imagem que representa o personagem em si, como mostra Figura 4.2.



Figura 4.2: Exemplos de imagens para composição de um personagem

Para que a composição funcione de forma adequada as imagens frente e fundo devem possuir as mesmas dimensões, sendo assim, na tela de criação de personagens a imagem que o usuário escolhe é apenas um ícone e a imagem que é utilizada para composição possui o mesmo tamanho da imagem que representa o personagem como um todo (Figura 4.3).

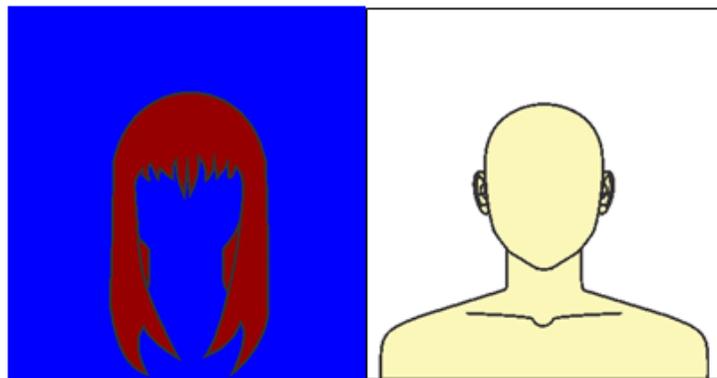


Figura 4.3.: Exemplo de imagens utilizadas para composição por sobreposição

Todo personagem possui como atributos variantes cabelo, roupa e rosto. O atributo corpo do personagem é estático, variando apenas por conta da escolha do sexo do personagem. Cada um desses atributos é armazenado como uma imagem, sendo essa definida pela escolha do usuário, ou definida pelo sistema no momento em que a tela de criação de personagem é carregada, definindo assim um personagem padrão.

Tanto no processo de construção de um personagem padrão, como na edição de partes de um personagem, uma sequência de composição deve ser respeitada para que a imagem

resultante esteja correta, pois para obter o personagem desejado várias composições podem ser realizadas, uma vez que uma imagem composta é o resultado de uma sobreposição de duas outras imagens.

Dessa forma, por meio do armazenamento das imagens selecionadas como parte do personagem a imagem que irá representar o mesmo será obtida a partir de três composições consecutivas:

- (i) Uma composição entre a imagem do corpo do personagem, imagem de fundo, com a imagem do rosto selecionado, imagem de frente;
- (ii) Composição entre a imagem resultante do processo anterior, imagem fundo, juntamente com a imagem que representa a roupa selecionada pelo usuário, imagem de frente;
- (iii) E a última composição tem como resultado a sobreposição da imagem do cabelo selecionado sobre a imagem de fundo resultante da composição anterior.

Assim, devido a uma sequência específica de composição os olhos do personagem não ficarão sobre o seu cabelo, e o cabelo não ficará por baixo da roupa selecionada.

Um processo realizado antes da composição de imagens propriamente dita é a seleção de cores para cada parte do corpo do personagem, ou seja, quando o usuário seleciona uma parte do corpo do personagem, o mesmo pode também selecionar uma cor para essa parte. Sendo assim, toda vez que o usuário seleciona uma cor, a imagem referente à parte selecionada é armazenada como atributo do personagem, é modificada com base em duas variáveis de controle, uma que indica a cor do item atual e outra que indica a cor que o item irá assumir. Com o auxílio de uma máscara de *bits*, semelhante à utilizada para composição, que será adiante explicada com mais detalhes, todos os *pixels* da imagem referente à parte do corpo selecionada que estiverem como cor atual terão sua cor substituída pela selecionada.

O algoritmo de composição de imagens utilizado no jogo se baseia em uma máscara binária implícita, detalhada posteriormente, tanto para sobreposição de cores, como para seleção de áreas de uma imagem, além do uso de uma função de opacidade utilizada para compor efetivamente a imagem do personagem.

Todas as imagens referentes às partes do corpo do personagem possuem um fundo de cor única, azul. A partir da adoção do método *Blue Screen* ou *Chroma Key*, onde o fundo de cor única de uma imagem é substituído por outra imagem, é possível separar somente as partes da imagem fundamentais à composição. Esse processo é ilustrado pela Figura 4.4.

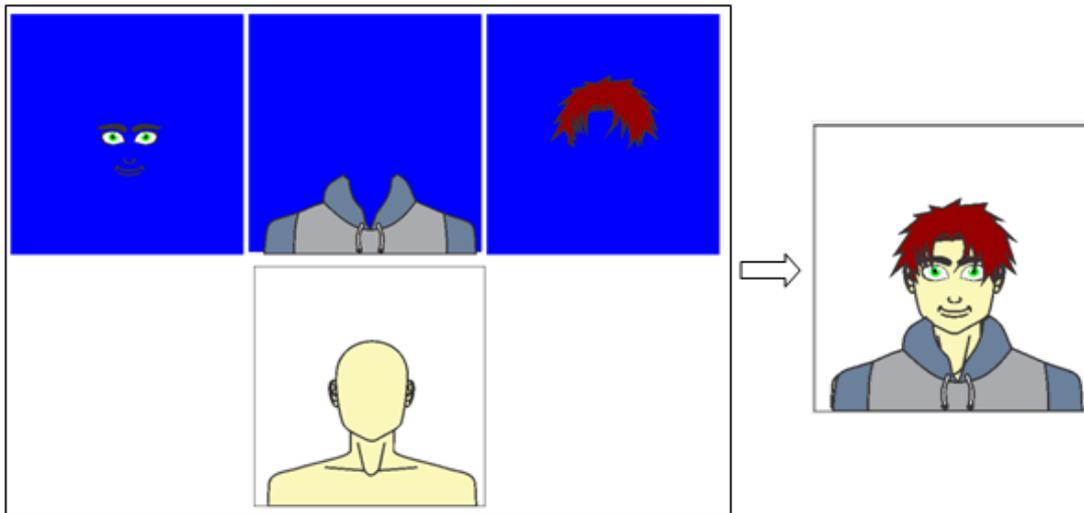


Figura 4.4: Resultado do processo de composição de um personagem

No caso da composição realizada no jogo, todos os *pixels* que tiverem uma cor diferente de azul na imagem selecionada pelo usuário, serão mapeados para uma máscara binária por meio do valor 255, e todos os *pixels* que estiverem com a cor azul serão mapeados para a mesma máscara com o valor 0, tal que todo *pixel* da imagem selecionada que possuir o valor 255 na máscara em relação à sua posição na imagem original será pintado sobre a imagem de fundo.

A máscara binária acaba se tornando implícita pela utilização do processo *Chroma Key* e, sobretudo, devido às características das imagens utilizadas para composição, pois a aplicação do *Chroma Key* sobre as imagens as divide em frente e fundo com base na cor de fundo, no caso azul. Sendo assim, apenas com dois valores é possível representar o que é frente e o que é fundo.

Além da máscara de *bits* para sobreposição de imagens a implementação utiliza uma função de opacidade que determina quanto de contribuição, referente a cor, um *pixel* da imagem de frente ou da imagem de fundo possui em relação à imagem final, considerando os três canais de cores (*Red*) R, (*Green*) G, (*Blue*) B. A função alfa é dada pela Equação 4.1.

$$f(\text{corPixel})_{xy} = \left(\left(\left(1 - \left(\frac{\alpha_{xy}}{255} \right) \right) * \text{corFundo} \right) + \left(\frac{\alpha_{xy}}{255} * \text{corFrente} \right) \right) \quad (4.1)$$

onde:

α é o coeficiente de opacidade do *pixel*.

xy são índices do *pixel* na imagem de frente, na imagem de fundo e imagem composta.

A função alfa é baseada na contribuição de cada *pixel* da imagem de frente e de fundo, que é definida geralmente como um fator que varia de zero a um. No caso do jogo, esse fator pode apenas assumir os valores 0 e 255. Quando α assumir o valor 0, o *pixel* da imagem de frente não será pintado e o *pixel* da imagem de fundo será pintado, quando α for 255 o *pixel* da imagem de frente será pintado e o da imagem de fundo não.

Tanto a arquitetura do jogo como o processo de composição de imagens, incorporado ao mesmo, são de extrema importância para a interação entre o jogador e jogo, pois esses oferecem a base para a implementação das funcionalidades propostas. Porém, outro grande aspecto que deve ser levado em consideração se refere ao desenvolvimento da interface do jogo, pois será por meio desta que o usuário irá acessar as funcionalidades desenvolvidas. Além do que, uma interface bem elaborada e planejada pode atender de forma mais adequada às necessidades do usuário, influenciado na aceitação do mesmo em relação ao sistema.

Capítulo 5

Interação e Interface do Jogo

Assim como no desenvolvimento de qualquer sistema computacional, em um jogo o perfil e as necessidades do usuário devem fazer parte do processo de análise, planejamento e desenvolvimento, ou seja, certos cuidados e decisões devem ser tomados para garantir que a interação entre usuário-sistema seja agradável e intuitiva. Porém, diferentemente de um sistema comercial, um jogo incorpora a arte e tem por requisitos básicos, além ser intuitivo e claro, ser atrativo e divertido ao jogador.

Desse modo, é importante que um jogo ofereça recursos capazes de prender a atenção do usuário, ou ainda, atraí-lo. Assim, a interface de um jogo acaba se tornando um fator chave e crucial para a interação com o jogador, pois é por meio dela que este irá interagir com os cenários, personagens e objetos, e que quanto mais atrativa e fácil de usá-la maior a sensação de imersão provocada ao usuário.

De acordo com Barros (2011), um jogo deve possuir uma interface intuitiva, usual, clara, consistente e atrativa, pois dessa forma o mesmo poderá alcançar as expectativas e a aceitação do usuário. Assim um jogo deve possuir uma interface bem estruturada e, sobretudo, lúdica, pois fará o papel de mediador entre o mundo virtual apresentado pelo jogo e o usuário.

O *designer* deverá agregar à interface do jogo recursos capazes de suprir as necessidades do usuário, como em qualquer sistema computacional. Para isso, a interface desenvolvida deverá oferecer boa comunicabilidade e usabilidade, ou seja, as mensagens emitidas pelo *designer* devem ser claras para o receptor, o usuário, e as opções disponíveis devem estar apresentadas de forma simples, intuitiva e autoexplicativa.

Segundo Prates *et al.* (2000), comunicabilidade pode ser definida como uma propriedade presente em um software que comunica eficientemente e efetivamente aos usuários as intenções do *design* e princípios interativos expressos por ele. Já usabilidade, conforme

Barbosa e Silvia (2010), tradicionalmente enfoca a maneira como o uso de um sistema interativo no ambiente de trabalho é afetado por características do usuário, bem como, capacidades cognitivas, capacidade de agir sobre a interface e capacidade de perceber as respostas do sistema.

No jogo proposto, além da atratividade atribuída à interface, os aspectos de aprendizagem devem ser levados em consideração. Logo, certos recursos devem ser agregados à interface para garantir uma interação adequada a realidade do usuário surdo, como utilização de vídeos, textos e imagens.

Também é de extrema importância que as mensagens transmitidas ao usuário sejam claras de forma que o mesmo não se sinta perdido durante a interação, além da preocupação existente em relação ao aprendizado via interface, pois o jogo tem como proposta contribuir na aprendizagem do jogador. O jogo Conceitos agrega em sua estrutura alguns recursos pensados para minimizar as dificuldades de interação entre o usuário e a interface, aumentando assim a qualidade da interação, além da satisfação do usuário com o jogo:

- Utilização de imagens para tradução de funcionalidades básicas do jogo para a língua de sinais, ou ainda, a tradução de palavras que representam ações na interface, como botões e opções de controle.
- Utilização de vídeos como forma de instrução e ajuda ao usuário, sendo esses vídeos uma composição entre um vídeo tutorial, mostrando os principais componentes presentes em uma determinada tela, e outro vídeo em Libras explicando as funcionalidades de cada componente apresentado.
- Utilização de vídeos em língua de sinais para auxiliar o aluno na compreensão dos pedidos do personagem.
- Preocupação com a confusão visual: devido às limitações do usuário as interfaces do jogo são apresentadas com fundos mais amenos destacando as funcionalidades, ou ações que o usuário deve realizar na interação.
- Redução das informações requisitadas pelo sistema ao usuário, ou seja, o usuário deverá inserir no sistema apenas os dados fundamentais.
- Personalização de alguns componentes presentes na interface Java para atender as necessidades do usuário do sistema e tornar a interação mais agradável.

Além de recursos utilizados na interface para auxiliar o processo de interação entre o usuário e jogo, o desenvolvimento foi precedido por uma fase de análise e planejamento, onde

foram estudadas as possibilidades de recursos que poderiam ser utilizados na interface e também possíveis cenários de interação, no intuito de evitar algumas situações não muito claras ao usuário durante o jogo.

Para realização de um estudo prévio sobre as peculiaridades da interface que iriam agregar valor na interação jogo-jogador foram desenvolvidos o modelo de Árvores de Tarefas Concorrentes (ATC) e o Diagrama *Modeling Language for Interaction as Conversation* (MOLIC), discutidos nas próximas subseções, além da apresentação de uma subseção que tem por objetivo detalhar a interação entre o jogo e jogador a partir da interface e exemplificar algumas decisões tomadas na elaboração o diagrama MOLIC que foram mapeadas para a interface do protótipo do jogo.

5.1 Diagrama MOLIC

O Diagrama MOLIC, segundo Neto e Leite (2011), possibilita ao projetista uma visualização que lhe permite planejar e avaliar a dinâmica (comportamento) do sistema face a interação do usuário, demonstrando a interação entre o usuário final e o projetista do sistema, além de estabelecer uma análise prévia sobre a interface, utilizada como critério de escolha e mapeamento dos componentes que irão compor a interface real.

O Diagrama MOLIC é constituído de várias cenas, sendo essas uma representação da interface do sistema real, demonstrando toda a interação entre o usuário e o sistema. O Diagrama MOLIC elaborado expressa todas as funcionalidades e as ações que o jogador pode realizar durante a interação, como a criação de conta, *login*, criação de personagens, interação com os ambientes e objetos, entre outras. Também, por meio de sua estrutura, contribui para que o *designer* consiga expressar determinados cenários de erros (*Breakpoints*), apontando possíveis pontos críticos que deverão ser levados em consideração no desenvolvimento da interface, como é o caso da conta e do personagem do jogo que, ao serem definidos pelo usuário, podem apresentar problemas no momento em que o jogador requisitar que esses sejam salvos. A Figura 5.1 traz o Diagrama MOLIC do jogo Conceitos com seus possíveis cenários de interação, sendo esses a criação de conta, acesso ao jogo e interação com os ambientes tridimensionais.

O diagrama descreve toda a interação entre o usuário e o jogo, tanto a parte de 2D como a 3D. Ao acessar o jogo o usuário irá se deparar com um menu de ações; caso o mesmo não possua conta no jogo terá que escolher a opção criar conta, a qual direcionará o usuário à

criação de personagens e a seleção de ambiente, caso o usuário já possua conta poderá, a partir da efetuação do *login*, acessar a seleção de ambiente e em seguida iniciar o jogo. O sistema de ajuda do jogo Conceitos é baseado em vídeos, os quais explicarão ao usuário as funcionalidades presentes em cada tela por meio da Libras.

A parte 3D se baseia na interação entre o usuário e seu personagem, sendo tarefa do usuário encontrar os objetos requisitados pelo personagem. Para todos os objetos o usuário poderá acessar um modo de edição, onde o mesmo poderá visualizar imagens relacionadas ao objeto selecionado ou, até mesmo, alterar algumas de suas propriedades. Em qualquer interação com o jogo o usuário terá acesso a um menu de opções, onde este poderá reiniciar a partida, salvar o progresso, abandonar a partida e verificar sua pontuação.

A utilização do diagrama MOLIC possibilita a elaboração prévia da interface final do sistema identificando pontos críticos presentes na interface que poderiam gerar confusão durante a interação do usuário. Diferentemente de um Diagrama de Sequência da Engenharia de Software, que especifica a comunicação entre componentes (classes) de um sistema, o MOLIC destaca a comunicação e as mensagens trocadas em determinados momentos da interação em relação ao uso da interface, ou seja, expressa a comunicação entre usuário e sistema, contribuindo para o aumento da qualidade da interação e satisfação do usuário, o que para um jogo é um fator crucial.

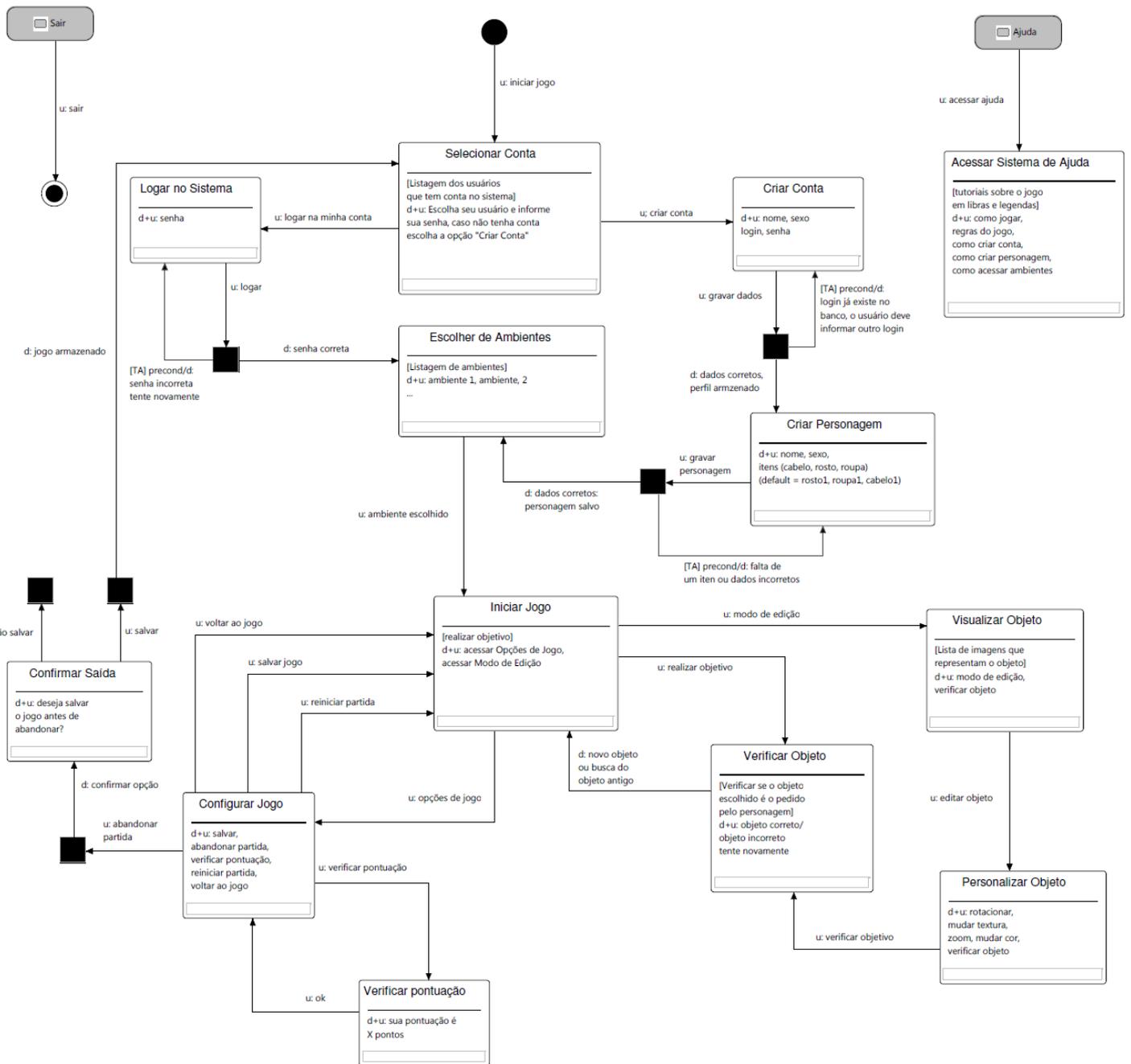


Figura 5.1: Diagrama MOLIC do jogo Conceitos

5.2 Diagrama CTT

Segundo Barbosa e Silva (2010), o CTT tem como objetivo auxiliar o *designer* no processo de construção de interfaces e a avaliação da Interação Humano-Computador (IHC), ressaltando a especificação de tarefas na interação. Esse modelo possibilita que o projetista do sistema decomponha determinadas funcionalidades presentes no sistema em tarefas, sendo esse processo vantajoso para o desenvolvimento da interface do sistema, pois descreve o processo de uso das principais funcionalidades do sistema pelo usuário, além de especificar certas situações que poderiam acontecer durante a utilização do sistema pelo usuário.

Devido à modelagem CTT ser restrita a uma determinada funcionalidade, ou seja, cada diagrama CTT se refere a uma funcionalidade específica do sistema, além do desenvolvimento do diagrama MOLIC que esclarece várias peculiaridades da interface, para construção do jogo Conceitos foram elaborados apenas dois desses diagramas, sendo esses referentes a duas funcionalidades essenciais para a interação, sendo essas a criação de personagens e a realização de objetivos do jogo por parte do usuário.

No processo de criação de personagem, assim como mostrado na Figura 5.2, o usuário define primeiramente os dados do personagem, fornecendo o nome e o sexo de seu personagem, porém, a ordem em que essas informações são repassadas para o sistema é irrelevante; outro detalhe evidenciado é que a submissão do nome e sexo do personagem é totalmente dependente do preenchimento dos campos.

Somente depois de definir os dados de seu personagem o usuário poderá prosseguir para a definição de detalhes, sendo importante ressaltar que depois dos dados submetidos ao sistema não é possível realizar alterações nos mesmos. Na definição dos detalhes do personagem o usuário irá escolher as cores e as partes do corpo de seu personagem, sendo a escolha das cores dependente da escolha das partes do corpo; em seguida o usuário poderá concluir o processo de criação de seu personagem, porém esse só poderá ser realizado se os detalhes do personagem já estiverem definidos.

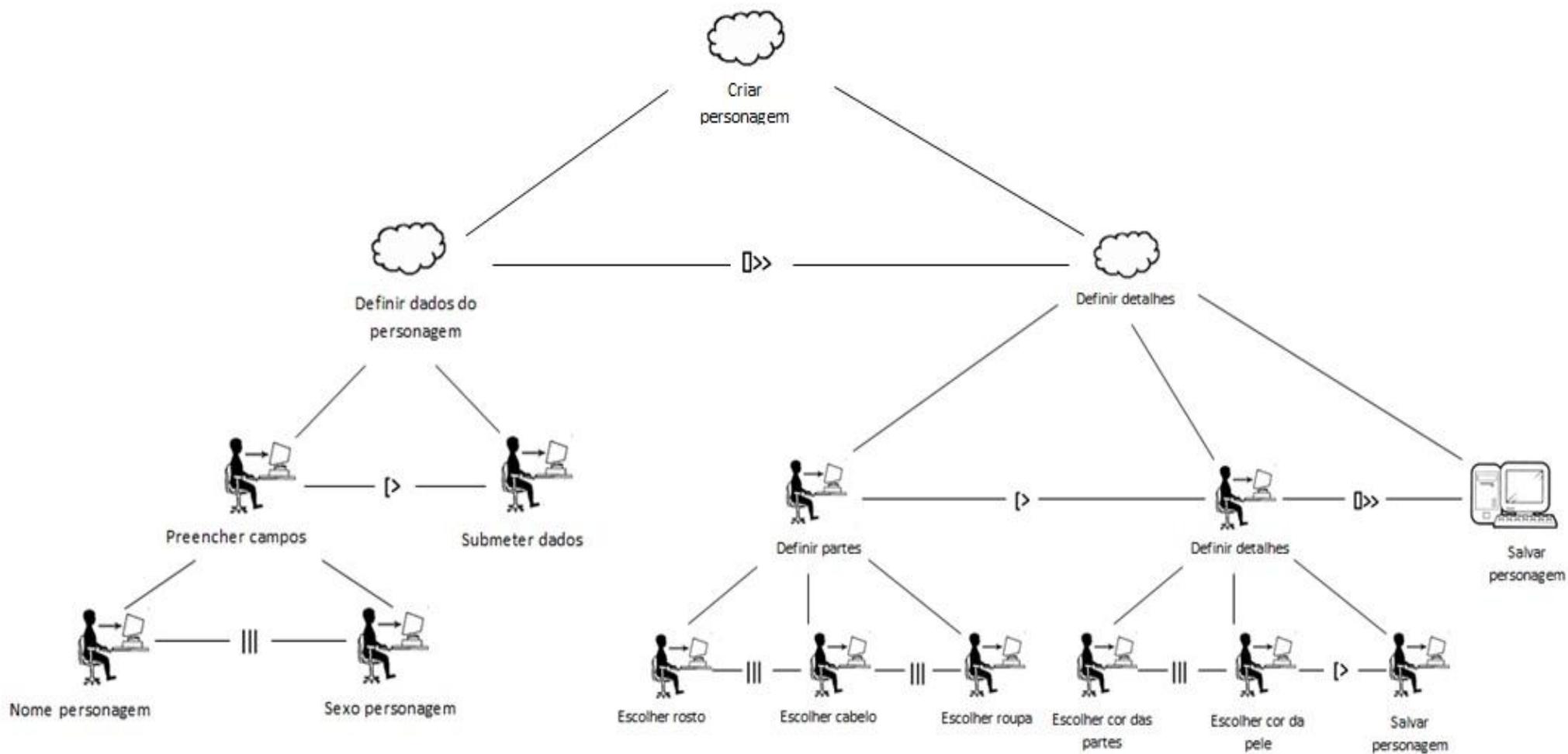


Figura 5.2: Diagrama CTT referente à criação de personagens

Na realização de objetivos, mostrado na Figura 5.3, o usuário terá que encontrar em um determinado ambiente um objeto requisitado por seu personagem. O jogador terá que se deslocar em meio ao cenário selecionando objetos até encontrar o que lhe foi requisitado.

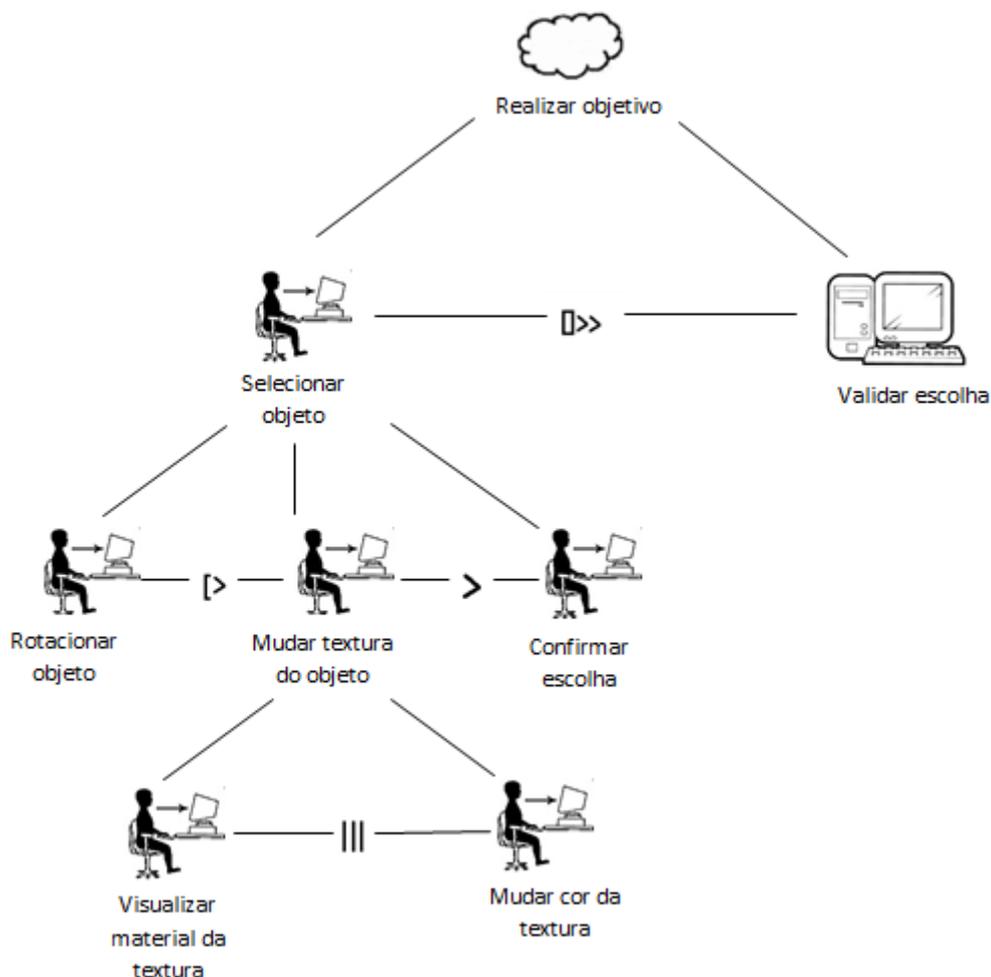


Figura 5.3: Diagrama CTT referente à realização de objetivos

Como evidenciado pela Figura 5.3, ao selecionar um objeto o jogador poderá rotacionar, visualizar o material que compõe o mesmo, mudar esse material e em seguida confirmar a escolha do objeto. Somente depois que o usuário definir o objeto como escolhido o sistema do jogo irá verificar se o objeto selecionado é o requisitado.

As opções de personalização vinculadas a um objeto, como a mudança de textura, cor e material favorecem o aprendizado do aluno, pois como a concepção de informação e o processo de aprendizado, no caso do aluno surdo, acontece de forma totalmente visual ao se passar a ideia de que um objeto pode assumir diferentes propriedades e mesmo assim não deixa de ser o mesmo objeto contribui para a fixação e vinculação do conceito, palavra e

imagem do objeto em especial, pois mesmo uma cadeira sendo de madeira marrom ou de plástico branco não deixa de ser uma cadeira.

5.3 O MOLIC e a Interação do Jogo Conceitos

A partir da utilização do diagrama MOLIC foi estabelecida uma análise sobre as possíveis soluções que deveriam estar presentes na interface do jogo Conceitos, ou ainda, quais componentes gráficos seriam os mais adequados para minimizar as necessidades do usuário, além de favorecer suas capacidades cognitivas. Recursos, como o uso de vídeos e imagens em lugares específicos, redução de informação em certas telas, organização e planejamento dos caminhos cognitivos, entre outros aspectos, foram estudados e definidos para elaboração da interface do jogo. A seguir serão apresentadas as cenas que compõem o digrama MOLIC desenvolvido, divididas com base nas funcionalidades implementadas, e em como essas foram mapeadas para a interface implementada no jogo.

5.3.1 Opções Iniciais

Ao iniciar o jogo o jogador poderá escolher o que deseja fazer a partir de duas opções principais, criar uma nova conta ou logar, como mostrado na Figura 5.4. Na interface implementada alguns recursos visuais foram adicionados visando prover uma melhor adequação da interface ao jogador, como a exibição *gifs* animados e espaços para a reprodução de vídeos, como ilustra Figura 5.5.

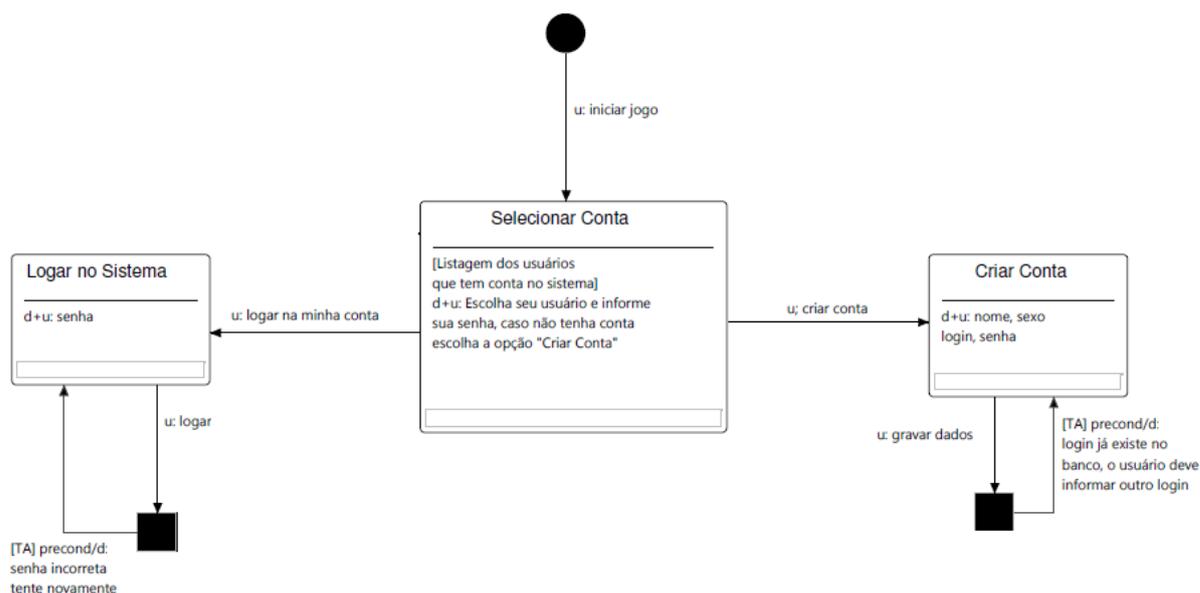


Figura 5.4: Opções proposta ao jogador a iniciar o Jogo

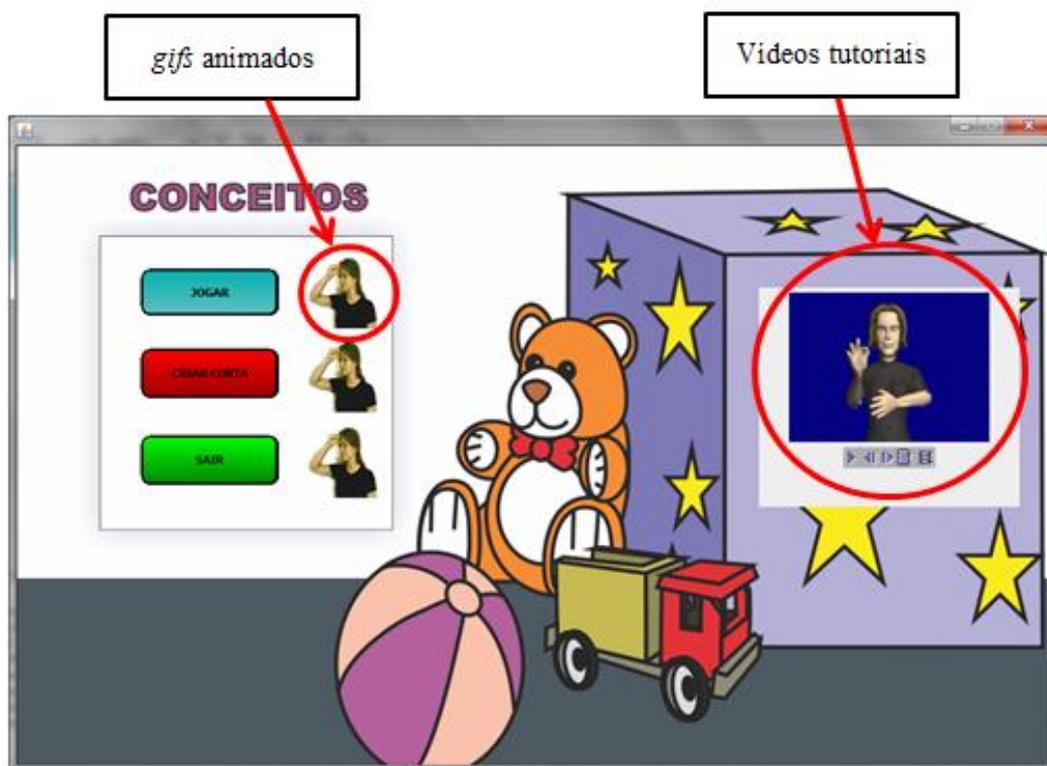


Figura 5.5: Tela inicial implementada no Jogo

Como é possível verificar os fluxos de interação foram mantidos na Figura 5.5, ou seja, caso o jogador possua uma conta o mesmo irá selecionar a opção JOGAR, representada pelo botão azul, caso contrário o mesmo terá que criar uma conta, a partir da opção CRIAR CONTA representada pelo botão vermelho. A opção SAIR foi adicionada como funcionalidade básica na interface.

Algumas inconsistências entre a tela implementada e proposta no diagrama MOLIC da Figura 5.4 podem ser verificadas, pois no diagrama a tela inicial deveria possuir uma listagem de todos os jogadores cadastrados no sistema, e a criança surda ao iniciar o jogo acessaria sua conta ou até mesmo criaria uma nova conta por meio de uma listagem dos usuários cadastrados no sistema. Essa opção não foi mapeada para a interface final devido à intervenção de um profissional da área da surdez, que afirmou que as informações presentes na interface devem ser minimizadas para que a criança não se confunda durante a interação, ou seja, a exibição de uma listagem de usuários poderia confundir a criança no momento em que a mesma fosse encontrar seu *login*, ou ainda, poderia ficar sem saber o que fazer caso desejasse criar uma conta.

Alguns recursos foram adicionados à interface final para aumentar a qualidade da interação, como é o caso dos *gifs* animados e dos vídeos tutoriais, já citados. Esses recursos irão

contribuir para que as mensagens apresentadas na interface sejam recebidas e entendidas pelo jogador, pois serão representadas por meio das Libras, língua mais próxima de sua realidade.

5.3.2 Carregar Jogo ou Logar

Quando um jogador logar no sistema esse irá apenas inserir seu *login*, definido no processo de criação de conta. Após essa ação o sistema irá validar o *login* informado, liberando acesso ao jogador. Caso o *login* esteja incorreto o sistema avisará e irá requisitar ao usuário um novo *login*.

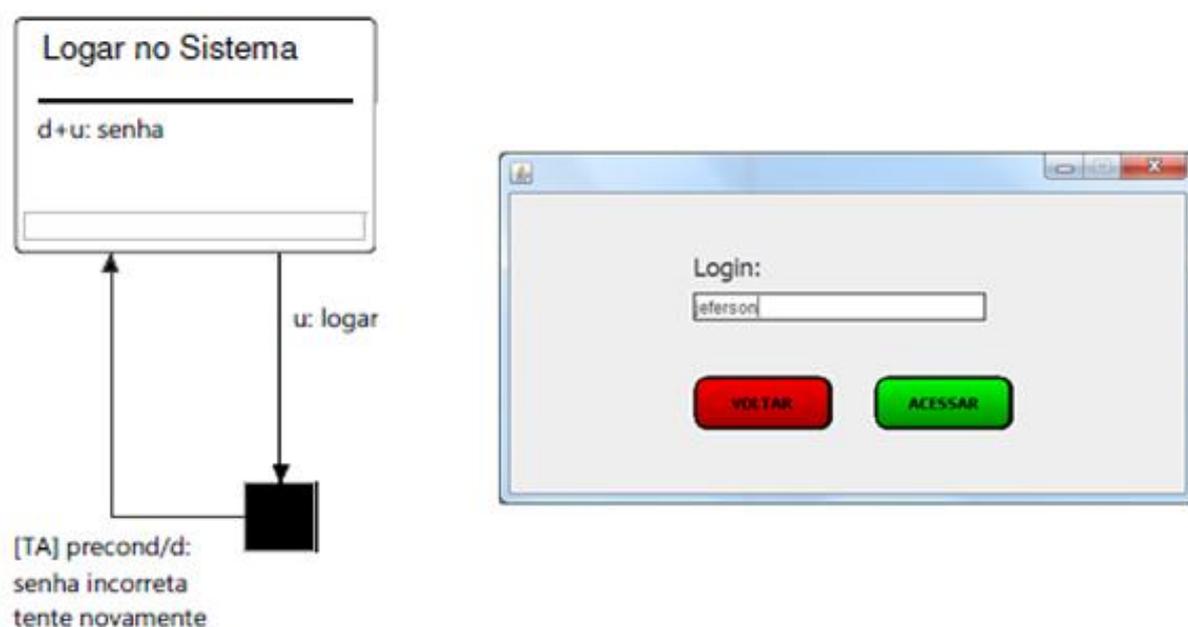


Figura 5.6: Mapeamento da funcionalidade Logar no jogo

No modelo MOLIC apresentado na Figura 5.6, mais especificamente no lado esquerdo da mesma, o aluno realiza seu *login* informando a senha da conta, contudo, após algumas reuniões como professores da área de ensino relacionada a surdez, a senha foi substituída pelo *login* do aluno, pois segundo esses professores, como os usuários são crianças surdas de 1ª a 5ª series, segurança não é um fator primordial para o jogo. Além do que o uso de uma senha poderia dificultar o acesso à conta, assim como uma listagem de *login*, pois a criança teria que se lembrar de uma sequência numérica e/ou mais alguns caracteres, além de poder se confundir ao tentar procurar seu *login*. Com o *login* de forma direta, possivelmente o próprio nome, o acesso a conta se torna mais adequado a criança. As funcionalidades presentes na tela de *login* da interface implementada, se resumem em ACESSAR, botão verde, direcionando o

jogador para a tela de seleção de ambientes, e o botão vermelho, opção VOLTAR direcionando o jogador à tela inicial do jogo.

5.3.3 Criação e Edição de Conta

Como as demais telas apresentadas, as informações presentes na interface final foram reduzidas para auxiliar no processo de aprendizado e memorização do jogador, como apresentado na Figura 5.7.

A tela de edição dos dados do jogador, no caso a edição da conta, e a tela de criação de conta são idênticas, pois as informações presentes na edição e na criação de conta são iguais, sendo que a única diferença está no preenchimento, pois ao editar os dados as informações do jogador são carregadas do banco, e já ao criar a conta o jogador insere seus dados no sistema.

Na tela de criação de conta e edição dos dados do jogador, ao contrário do modelo a informação de senha foi retirada da interface, as opções adicionais de vídeo e *gifs* para representar a tradução das informações em português para Libras foram agregadas a interface de criação de conta.



Figura 5.7: Processo para criação de conta

Como pode ser observado na Figura 5.7 após a inserção dos dados, tanto para edição como inserção, o jogador terá duas opções, (i) SALVAR: botão verde que confirma a operação de criação de conta ou atualização dos dados no banco de dados e direciona a criação para a tela de criação de personagens, e (ii) VOLTAR: botão vermelho, retorna a tela inicial do jogo. A

escolha do sexo influencia na criação do personagem, pois na tela de criação de personagens o sexo do personagem será o mesmo que o do jogador.

5.3.4 Criar Personagem

O processo de criação de personagem como visto na Figura 5.8, é totalmente dependente da criação da conta, e somente depois da conta criada o jogador terá acesso a essa tela. A tela de criação de personagem é apenas acessada no momento seguinte à criação da conta, ou seja, depois do personagem criado em nenhum momento da interação o jogador conseguirá alterar ou criar um novo personagem, salvo o jogador criar uma nova conta.

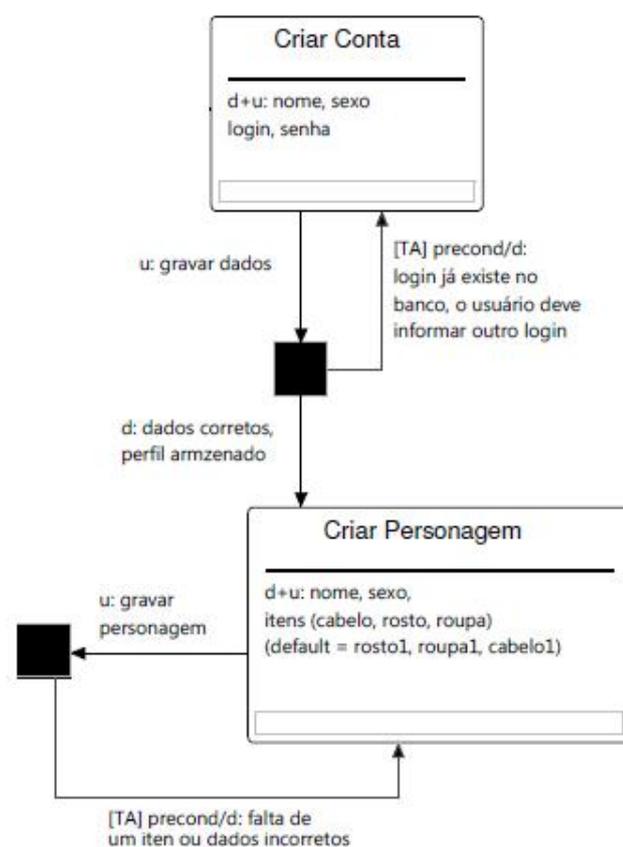


Figura 5.8: Processo para criação de personagem

A Figura 5.9 exemplifica o processo de criação de um personagem feminino implementado na interface final do jogo, além de apresentar as opções de itens que o jogador pode selecionar para compor seu personagem. Para cada parte do corpo a criança pode selecionar uma cor entre um conjunto pré-definido para atribuir a parte selecionada. Definidas todas as partes e o nome do personagem escolhido ao clicar no botão verde, opção CRIAR, o personagem será

vinculado a conta da criança e será salvo no banco de dados, sendo posteriormente utilizado em toda a interação com o jogo. O término da criação de personagem finaliza o processo de criação de conta, pois a partir da imagem salva a conta é efetivamente registrada no sistema e passa a ser possível escolher o ambiente de jogo e acessar as funcionalidades presentes na parte 3D do mesmo.

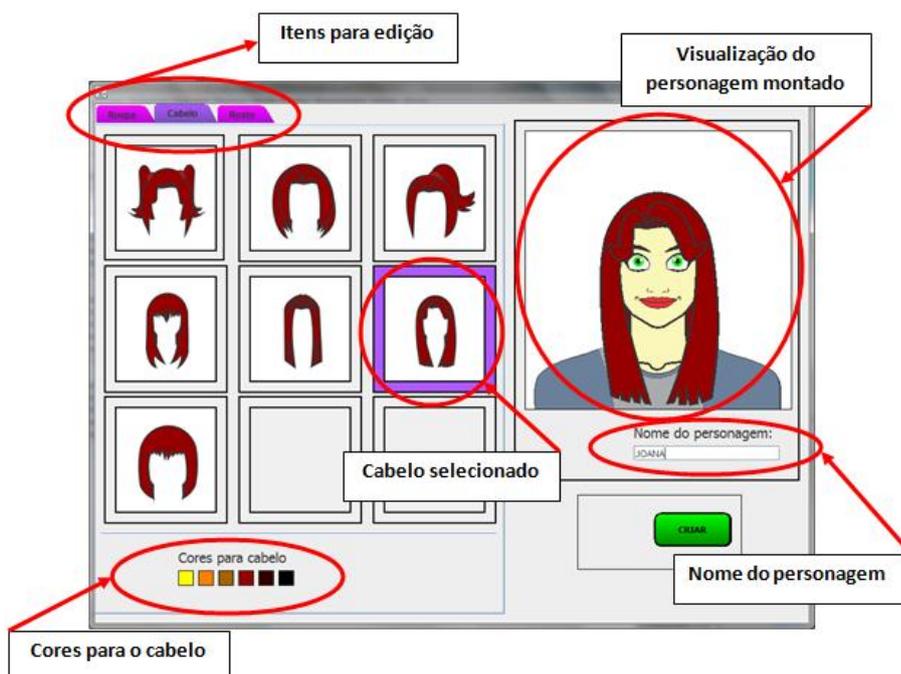


Figura 5.9: Tela para criação de personagem

5.3.5 Escolher Ambiente de Jogo

A seleção de ambientes pode ser acessada por meio de dois fluxos de execução distintos, como mostrado na Figura 5.10. Por meio do *login* e após a finalização do processo de criação de conta. A seleção de ambientes faz a ponte entre a interface 2D e a parte 3D onde efetivamente o jogador irá interagir com os objetos tridimensionais.

A ideia básica presente no diagrama da Figura 5.10 se refere à organização das telas e o processo de acesso a um ambiente, onde o jogador deverá escolher um ambiente em meio a uma lista de ambientes já liberados para ele.

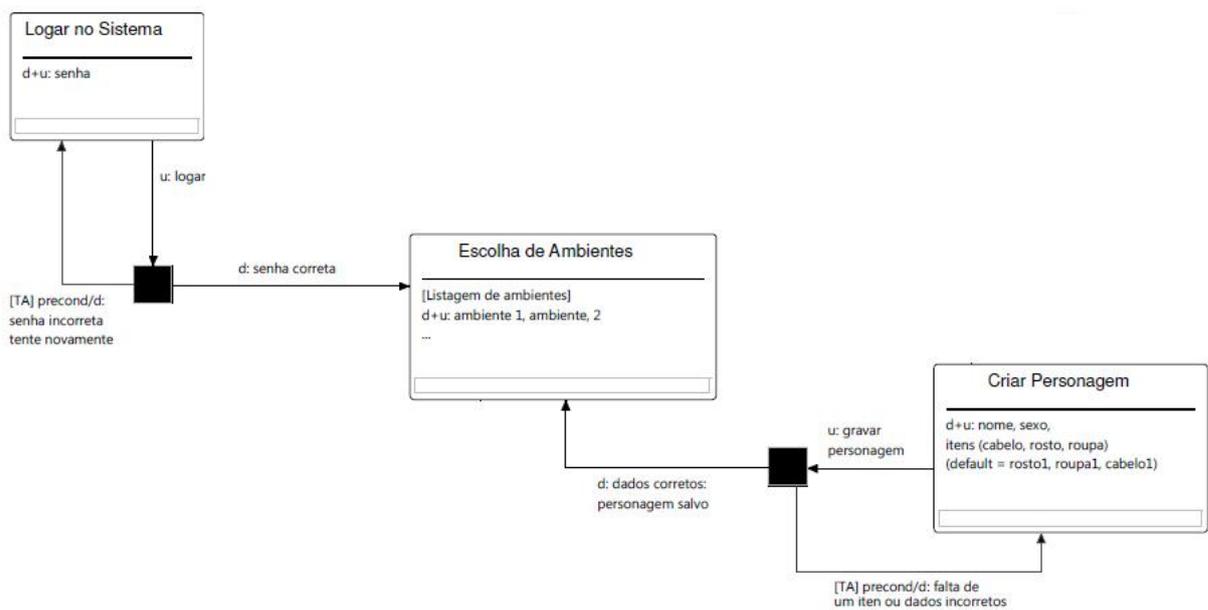


Figura 5.10: Formas de acessar a tela Escolha de Ambiente

Para o jogador acessar um novo ambiente este deve completar os ambientes já liberados. Na Figura 5.11 é possível verificar que o ambiente dois só será liberado depois de completado o ambiente um. Para um ambiente ainda não liberado para o jogador será exibido um cadeado; já um ambiente liberado é apresentado por uma imagem do próprio ambiente. Os ambientes possuem três estados relacionados ao status do mesmo, ambiente liberado, ambiente completo e estado jogando.

A interface final para seleção de ambientes possui várias informações que não foram adicionadas ao diagrama MOLIC do jogo Conceitos; informações utilizadas para expressar alguns dados tanto da conta do jogador, como do próprio ambiente. Na interface final da Figura 5.11 para cada ambiente liberado e selecionado no canto direito da mesma, informações como nome do ambiente, números de objetos encontrados, status, entre outras são exibidos à direita da tela. Na esquerda são exibidos o *login* do jogador, a imagem do personagem da conta e o nome do personagem.

Após a escolha do ambiente o jogador deve clicar no botão verde, opção JOGAR, que direcionará o jogador para uma janela OpenGL, na qual a parte 3D será exibida. A opção EDITAR CONTA, representada pelo botão azul direciona o jogador a uma tela semelhante à utilizada para criar uma nova conta, porém, no processo de edição os dados do jogador são carregados do banco e ficam aptos para edição. Caso o jogador resolva abandonar o jogo o botão vermelho, opção SAIR direciona o jogador à tela inicial.

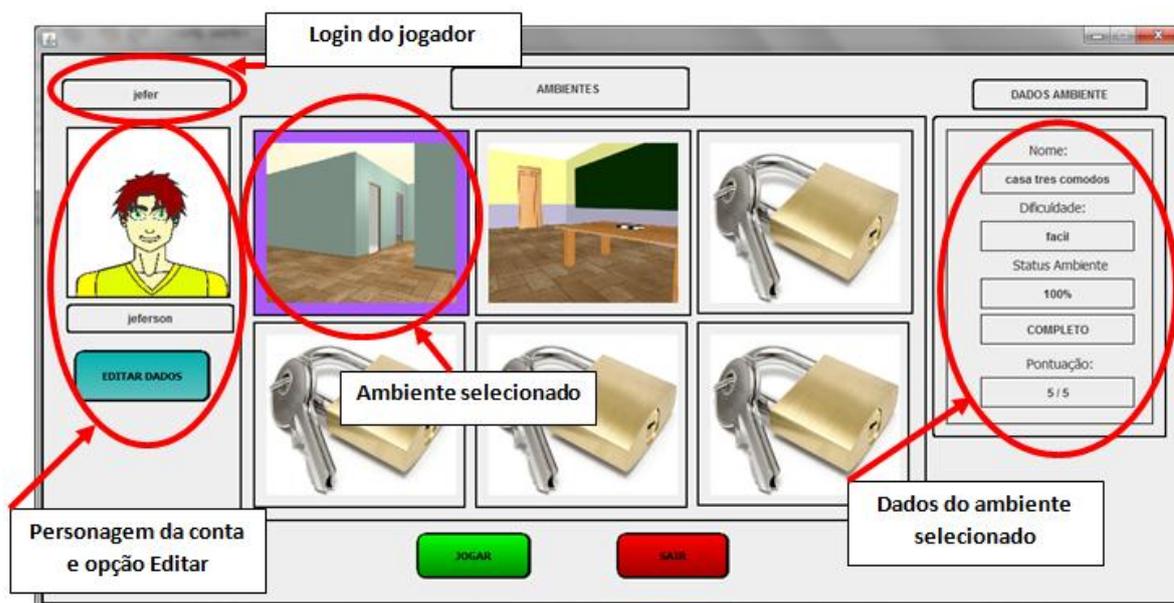


Figura 5.11: Liberando um novo ambiente

Quando o jogador selecionar um ambiente para jogar a aplicação acessa os modelos tridimensionais, telas, animações, imagens, entre outros componentes que compõem a parte 3D do jogo. Nessa fase o jogador, inserido em um ambiente tridimensional e representado por uma cápsula de jogo, um jogador no universo 3D, deve encontrar os objetos requisitados pelo seu personagem. Para cada objeto que o jogador encontrar sua pontuação aumentará; o jogo termina no momento em que o jogador encontrar todos os objetos requisitados pelo personagem.

O processo de comunicação entre o jogador e o jogo pode ser visualizado no diagrama da Figura 5.12, que descreve o funcionamento do jogo em relação à parte 3D, que tem como base a busca de objetos requisitados pelo personagem.

O modo de edição visualizado no diagrama da Figura 5.12, não foi implementado na versão atual do jogo, pois não é totalmente fundamental para interação, visto que o objetivo do jogo é encontrar os objetos requisitados, porém, será implementado futuramente quando as funcionalidades básicas do jogo estiverem totalmente validadas.

No modo de edição o objeto selecionado pelo jogador, sendo esse correto ou não, poderá ser manipulado por meio de operações de escalas, rotação e translação, além de opções para mudança de textura e cor. A tela Imagens de Objeto vinculada ao Modo de Edição irá carregar do banco de dados uma lista de imagens do objeto para que o jogador entenda que o objeto selecionado pode ser representado de várias formas.

Quando um jogador está interagindo com o ambiente 3D, o mesmo pode acessar as opções de jogo, essa funcionalidade foi implementada na interface final por meio da opção pausar

acessado pela tecla TAB, ou ainda, pode buscar o objeto requisitado pelo personagem para aumentar sua pontuação. No diagrama a pontuação pode ser acessada através do menu de opções de jogo, contudo no jogo Conceitos a pontuação pode ser visualizada durante toda a interação por meio da interface principal, como visualizado na Figura 5.13.

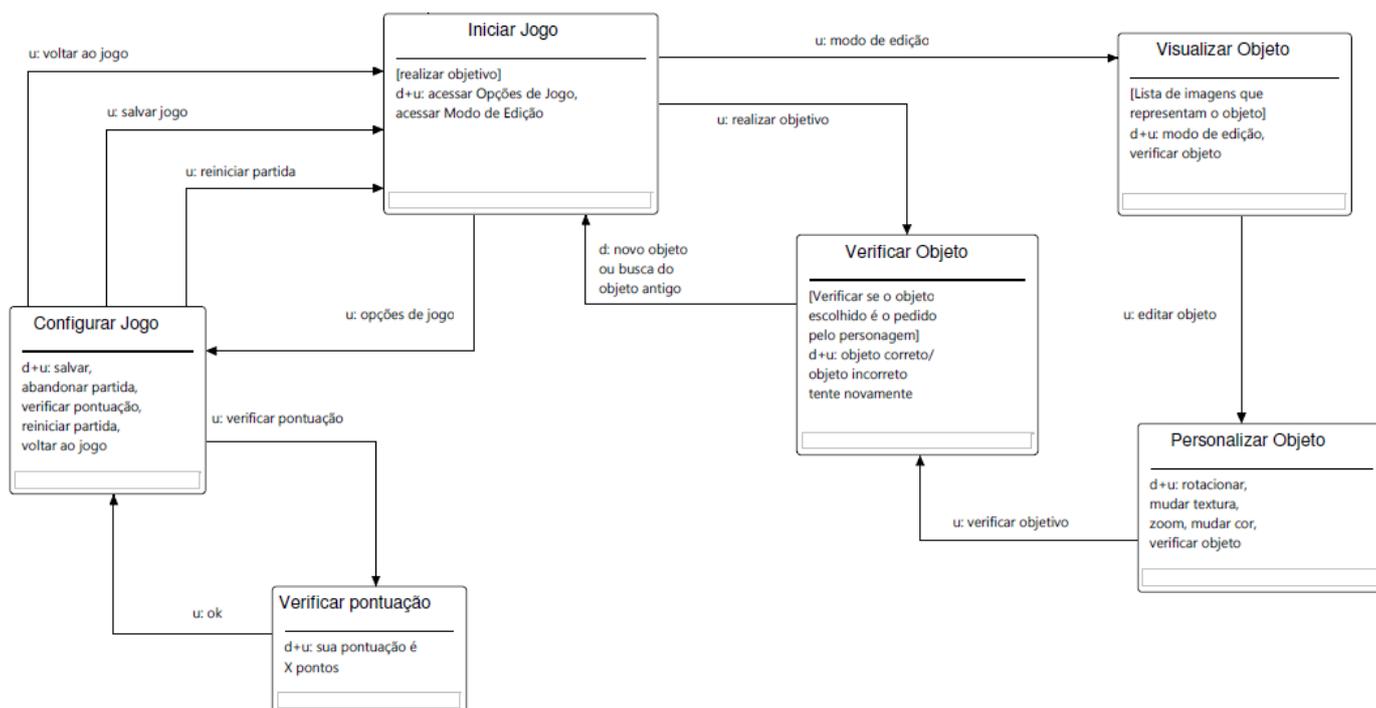


Figura 5.12: Dinâmica empregada no jogo Conceitos

5.3.6 Interação com o Ambiente 3D

Ao acessar um ambiente o personagem do jogador irá requisitar, por meio de uma mensagem de texto, que encontre certos objetos presentes no ambiente. A possibilidade de agregar a interface vídeos para tradução das mensagens transmitidas pelo personagem está sendo estudada, pois a *engine* jMonkey não oferece suporte a vídeos.

Para cada ambiente o jogador e os objetos presentes no mesmo são carregados em posições específicas. O jogador está livre para se movimentar pelo ambiente buscando o objeto. Para que o jogador consiga se guiar e para melhorar a interação, o jogador possui uma mão utilizada para selecionar os objetos presentes no ambiente. Dessa forma, por meio do pedido do personagem o jogador irá se movimentar pelo ambiente até encontrar o objeto requisitado.

Ao encontrar um objeto a pontuação do jogador é atualizada, sendo essa informada ao mesmo por meio de um *display* no canto esquerdo inferior, como mostra a Figura 5.13. O

display de pontos indica quanto objetos o jogador encontrou e também quantos objetos faltam para terminar o ambiente.



Figura 5.13: interface do jogo Conceitos em sua parte 3D

Quando um objeto é selecionado uma esfera vermelha surge acoplada ao objeto, indicando que foi selecionado. A partir da escolha de um objeto o personagem irá responder se o mesmo é o objeto esperado, pois cada objeto é armazenado no banco de dados com um nome vinculado e a lista de objetos que o personagem tem como referência. As Figuras 5.14 e Figura 5.15, mostram as tela de sucesso, quando o objeto selecionado é o correto, e a tela de fracasso, quando objeto encontrado é incorreto, respectivamente.

Ao final da interação, ou seja, quando todos os objetos são encontrados o personagem avisa que jogo acabou, pedindo ao aluno que jogue outro ambiente. Quando o ambiente é completado esse muda seu estado para completo, contudo, caso o jogador queira jogar um ambiente completo o mesmo será carregado com a pontuação zerada, porém, devido ao ambiente estar completo o jogador não poderá salvar seus avanços, pois como o ambiente está finalizado não é possível que o usuário atinja uma pontuação maior do que a já armazenada para aquele ambiente. A Figura 5.16, mostra a tela de fim de jogo.

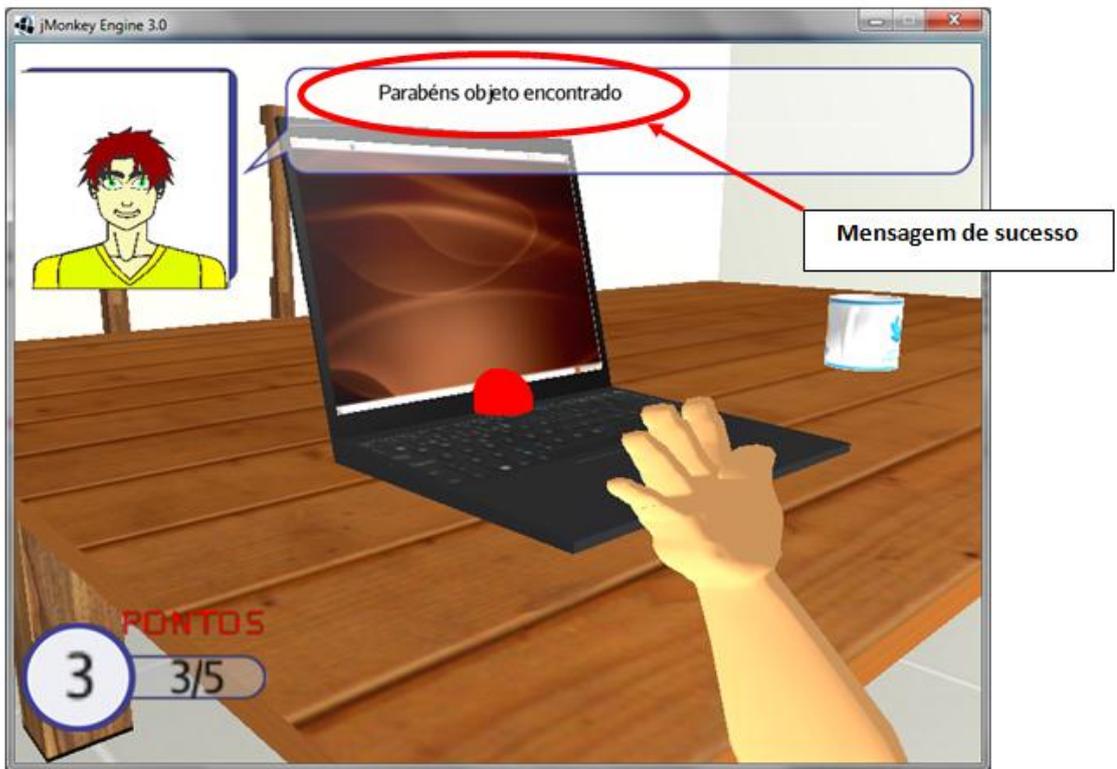


Figura 5.14: Seleção de objeto correto



Figura 5.15: Seleção de objeto incorreto



Figura 5.16: Fim de ambiente

5.3.7 Pausar e Opções de Jogo

Como a maioria dos jogos existentes, Conceitos possui um estado de pausa, onde o jogador pode parar totalmente o jogo e realizar algumas ações. Para acionar a funcionalidade pausar, o jogador deve pressionar a tecla TAB, pois a tecla ESC, naturalmente utilizada para esse propósito fecha a aplicação devido às configurações da *engine*. No estado pausado o jogador poderá salvar seu avanços, reiniciar sua partida ou abandonar o ambiente, como mostrado na Figura 5.17.

É importante ressaltar, que as imagens que representam o significado de cada botão foram apenas incorporadas na tela inicial do jogo como uma demonstração de um recurso que fará parte de interface final, sendo assim na interação 3D e em outras telas do jogo, em seu estado atual, essas imagens ainda não estão disponíveis. Pois será realizada uma reunião com o CAS para definir quais palavras são as mais adequadas para representar um conceito ou uma funcionalidade na interface e, somente a partir dessa etapa concluída, serão elaboradas as imagens correspondentes às ações de cada botão.

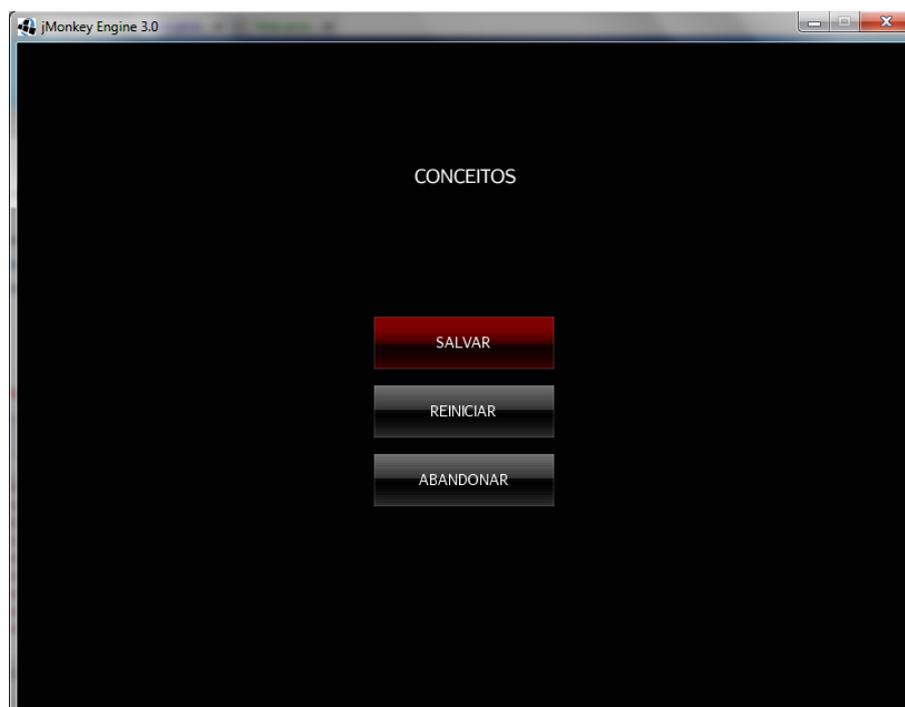


Figura 5.17: Tela de opções de jogo

Ao escolher a opção SALVAR o jogo irá verificar se pontuação obtida até o momento é maior que a última pontuação registrada; nesse caso a pontuação armazenada será substituída pela atual e o status do ambiente será recalculado e salvo com a pontuação. O status é uma variável que controla o percentual de completude de um ambiente, ou seja, se o jogador estiver com uma pontuação igual a dois e o número de objetos que devem ser encontrados é cinco, o status do ambiente será de quarenta por cento.

O estado do ambiente é controlado por seu status, ou seja, se o status for zero o estado do ambiente será liberado, e se o status for maior que zero e menor que cem o estado será jogando e se for igual a cem será completo. Toda vez que o jogador escolher a opção salvar e sua pontuação for maior do que a registrada o jogo irá direcionar o jogador para um tela de aviso, que irá indicar se a operação de persistência ocorreu de forma correta ou não.

Ao escolher a opção REINICIAR o jogador terá sua pontuação zerada e seu jogador, cápsula de jogo, será direcionado à posição inicial no espaço tridimensional do ambiente em questão, ou seja, o jogador será transladado para origem. Caso o jogador selecione a opção ABANDONAR todos os seus avanços não salvos serão perdidos, pois ao abandonar a partida entende-se que o jogador não deseja guardar seus resultados, diferentemente de uma opção sair e salvar.

5.3.8 Opções de Fim de Jogo

Quando o jogador encontra todos os objetos requisitados por seu personagem, imediatamente a mensagem de conclusão de fase é apresentada ao jogador uma tela que contém algumas opções de final de jogo. Nessa tela, Figura 5.18, o jogador pode escolher uma entre três opções disponíveis: (i) PRÓXIMA FASE: ao selecionar essa opção o ambiente que está na sequência do ambiente concluído é carregado e o jogador poderá jogar o novo ambiente sem precisar acessar a tela seleção de ambientes; (ii) JOGAR NOVAMENTE: essa opção recarrega o ambiente concluído semelhante a opção reiniciar, porém, salva a pontuação do jogador e muda o estado do ambiente para completo antes de zerar a pontuação e mover o jogador para a origem; (iii) SAIR E SALVAR: como o nome diz, diferentemente do abandonar partida, antes de sair do jogo salva a pontuação do jogador e o estado do ambiente e só então, encerra a parte 3D da interação.

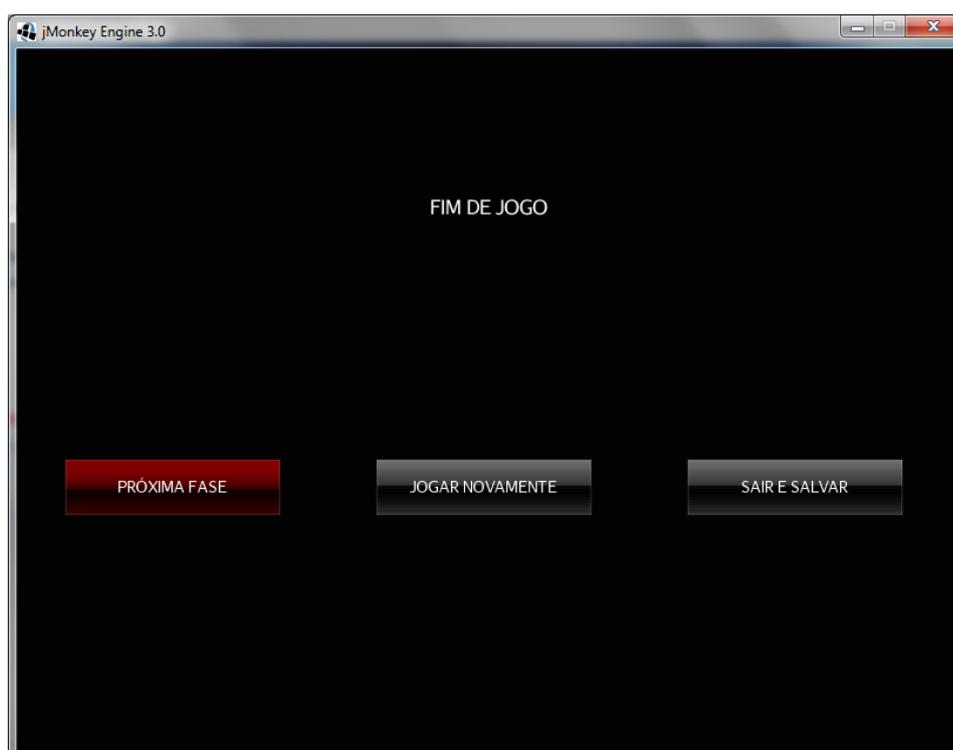


Figura 5.18: Mensagem de conclusão de salvamento

As opções e funcionalidades tanto da parte 2D como 3D do jogo poderão sofrer alterações, pois como a versão do jogo Conceitos ora apresentada ainda é um protótipo muitas mudanças podem ocorrer, principalmente por conta das validações junto ao CAS, Dependendo das ne-

cessidades dos usuários, certas peculiaridades do jogo podem ainda ser repensadas para melhorar e facilitar a interação.

5.4 MOLIC para uma Nova Versão do Jogo Conceitos

Com base na interface final desenvolvida é possível gerar um novo diagrama que expresse mais fielmente a interação entre o usuário e o jogo, além de identificar pontos críticos na interface.

Dado que o jogo Conceitos ainda é um protótipo a ser aprimorado até atingir uma versão estável adequada ao uso, a partir do protótipo do jogo terminado um novo diagrama MOLIC foi elaborado (Figura 5.19), como forma de documentar a interação entre usuário-sistema e especificar eventuais pontos problemáticos presentes na interação que podem ser melhorados. Alguns detalhes presentes no diagrama MOLIC elaborado antes do desenvolvimento da interface (Figura 5.1) foram retirados e novos detalhes foram adicionadas devido à implementação do jogo.

A grande diferença entre os dois modelos está na interação 3D do jogo e na forma como o usuário abandona a interação. Na parte 3D o usuário não precisa acessar uma nova tela para visualizar sua pontuação, que estará sempre presente em um *display*. Outra mudança ainda na interação 3D está na identificação de final de jogo. Ao realizar um objetivo o sistema verifica se o objeto é o último objeto da lista de requisição do personagem, caso seja a tela de fim de jogo é exibida. Caso contrário, o usuário deverá realizar uma nova busca pelo objeto requisitado. Diferentemente do MOLIC anterior o modo de edição pode ser acessado a partir da tela de jogo, ou seja, o usuário pode realizar um objetivo ou editar um objeto, de forma que as duas funcionalidades estão desvinculadas dando mais liberdade ao usuário, evitando a emissão de mensagens desnecessárias, uma vez que no MOLIC anterior se o usuário optasse apenas por editar um objeto ao final do processo o sistema iria validar se este foi o requisitado pelo personagem emitindo uma mensagem desnecessária de sucesso ou de fracasso.

O processo de finalização da interação no novo diagrama acontece com base no retrocesso à tela de seleção de ambientes, onde o usuário pode escolher outro ambiente de jogo, e depois para tela inicial, onde efetivamente é desvinculado do *login* saindo do jogo por meio de confirmação.

Em determinadas transições do diagrama algumas decisões de *design* devem ser tomadas para garantir que a comunicação entre *designer* e sistema seja a mais eficiente possível, favorecendo o entendimento da mensagem transmitida e a capacidade de aprendizado do usuário. Geralmente essas decisões estão vinculadas a representação da informação, ou seja, encontrar a melhor forma de expor ou requisitar uma informação ao usuário.

Como pode ser visto no diagrama da Figura 5.19, as transições em negrito representam pontos na interface que exigem uma decisão de *design* mais apurada, sendo assim, para cada uma dessas transições foi elaborada uma análise *rationale* que tem por base uma questão, representando um problema de representação de informação, uma proposta de solução e argumentos, sendo essas contribuições da solução para a interação do usuário com o sistema. As transições que agregam decisão de *design* são discutidas abaixo:

1 - Transição Logar

Devido ao público alvo, para acessar o jogo o aluno apenas deve informar seu *login*, não havendo a necessidade de senha, pois a memorização de *login* e senha pode prejudicar o uso do sistema pelo usuário.

Q1: Como permitir o acesso ao jogo sem prejudicar a interação aluno-sistema?

P1: Por meio uma tabela de *logins* com paginação ordenada em ordem alfabética, ou *combo box* → (Nova questão) **Q2:** Como facilitar a busca pelo *login*?

Arg [+] Facilita o acesso, pois o aluno deve apenas selecionar seu *login*.

Arg [-] Indução ao erro, pois o aluno pode selecionar uma conta que não é dele.

Arg [-] Aumento no tempo de busca pelo *login* caso vários usuários já estejam cadastrados no sistema.

P2: Em uma caixa de texto, filtro, onde o usuário informa seu *login* e o sistema verifica se esse existe?

Arg [+] Facilita o acesso, pois o aluno não precisa procurar seu *login* em meio a outros, evitando também acessos indevidos.

Arg [-] Impossibilidade de acesso ao jogo, caso o aluno não se lembre de seu *login*.

2 - Transição Acessar Ambientes

Tanto por meio da finalização do processo de criação de personagem quanto pelo *login* o aluno consegue acessar a tela referente à escolha de ambientes, porém, as informações devem ser representadas em uma estrutura que aumente a qualidade da interação deixando o aluno ciente de todas as informações de cada ambiente – o número de ambientes é restrito, no caso em seis, o que facilita a interação devido à redução de informação apresentada ao aluno surdo.

Q1: Como apresentar os ambientes e as informações referentes a esses ao aluno sem prejudicá-lo?

P1: Em uma lista de imagens, onde cada imagem representa um ambiente, sendo que as informações referentes a cada ambiente devem estar agrupadas em torno da imagem do ambiente, deixando a cargo do usuário selecionar o ambiente desejado → (nova questão) **Q2:** Como definir um ambiente *default*?

Arg [+] Permite ver todos os ambientes disponíveis.

Arg [+] Facilita o acesso do aluno a um ambiente específico

Arg [-] Aumenta a complexidade de cada ambiente devido à inclusão das informações desses junto suas imagens.

P2: Por meio de um marcador, seleção com cor diferenciada no ambiente, baseado no último ambiente completo pelo aluno. Caso o aluno esteja acessando a tela pela primeira vez o ambiente definido para o mesmo será o ambiente número 1.

Arg [+] *Feedback* visual do último acesso ao jogo.

Arg [+] Facilita a escolha do ambiente correto para jogo, no caso ambiente posterior ao ambiente completo.

3 - Transição Editar objetivo

O editar objeto é utilizado para expressar ao aluno que um objeto mesmo possuindo atributos diferentes, como textura, cor, tamanho, entre outros, possui sempre o mesmo conceito.

Q1: Como apresentar diferentes objetos vinculados pelo mesmo conceito ao aluno?

P1: Por meio de uma lista de imagens contendo várias representações do objeto selecionado em diferentes tamanhos, cores, texturas, etc.

Arg [+] Permite ao aluno visualizar rapidamente muitas variações do objeto selecionado.

Arg [+] Aumenta a compreensão do aluno em relação ao conceito do vinculado ao objeto.

Capítulo 6

Considerações Finais

Esse capítulo tem por objetivo apresentar as conclusões e trabalho futuros referentes ao jogo Conceitos, sendo assim o mesmo foi dividido em duas subseções no intuito de abordar de forma clara, respectivamente, esses tópicos.

6.1 Conclusões

No decorrer desse trabalho vários aspectos relacionados à interface e as ferramentas de desenvolvimento foram estudadas para garantir a implementação adequada do jogo. Porém, mesmo com um protótipo já desenvolvido, muitas melhorias e aprimoramentos devem ser agregados. O protótipo apresentado incorpora os principais requisitos elicitados junto ao CAS, estando totalmente aberto a mudanças e agregação de novos recursos, como é o caso de um módulo de edição de objetos, que mostrará ao jogador que o conceito de um objeto pode ser representado por várias imagens.

Em relação às dificuldades de desenvolvimento, a falta de experiência em programação de jogos, mais especificamente no uso da *engine* jMonkey e do modelador Blender, tornou-se um grande obstáculo para a implementação da parte 3D. Para apropriação dessas tecnologias muito tempo se fez necessário. Já em relação à elaboração da interface, levando em consideração o público-alvo, seu desenvolvimento não se mostrou trivial, pois diferentemente de outro sistema convencional a interface do jogo Conceitos deve garantir o máximo de recursos para que o usuário possa aprender e melhor interagir com o jogo.

Como forma de mensurar a dificuldade do desenvolvimento do jogo, por meio da abordagem de pontos de casos de uso, alguns critérios relacionados à experiência de programação, apropriação de tecnologias, estabilidade de requisitos entre outros, foram

avaliados no intuito de identificar critérios que poderiam interferir de forma direta no desenvolvimento de jogo. A Tabela 6.1 expressa alguns desses critérios.

Tabela 6.1: Critérios e nível da equipe de desenvolvimento

Critérios/Requisitos de Desenvolvimento	Nível da Equipe
Familiaridade da equipe com Scrum e XP	Média
Experiência da equipe	Baixa
Experiência da equipe em OO	Alta
Capacidade dos analistas da equipe	Baixa
Motivação	Alta
Estabilidade dos requisitos	Alta
Domínio da tecnologia e configuração do ambiente de programação	Baixa

Como pode ser visualizado na Tabela 6.1, a equipe¹ ao iniciar o desenvolvimento do jogo possuía algumas dificuldades, principalmente referentes a domínio das tecnologias utilizadas, falta de experiência em análise de sistemas e desenvolvimento de jogos, e tais fatores contribuíram para que o tempo de desenvolvimento fosse estimado em seis meses e meio.

Mesmo com o tempo para desenvolvimento restrito e com as dificuldades relacionadas a tecnologias e práticas de programação que surgiram durante o desenvolvimento, a implementação ocorreu de forma adequada, tanto que as funcionalidades fundamentais propostas inicialmente estão contempladas na versão atual do jogo. Porém, devido a falta de tempo para implementação duas funcionalidades consideradas não essenciais não foram desenvolvidas, sendo essas a visualização de histórico de resultados (funcionalidade 2D) e modo de edição de objetos (funcionalidade 3D), como pode ser visto na Tabelas 6.2 e 6.3.

¹ Os alunos Felipe da Silva Inacio e Guilherme Coelho, na disciplina de Processos de Engenharia de Software II auxiliaram no desenvolvimento do Jogo na parte de documentação, modelagem de objetos 3D e execução de testes.

Tabela 6.2: Funcionalidades 2D do Jogo

Requisitos	Prioridade	Status
Criar Conta Sistema	Alta	Implementado
Criar Personagem	Alta	Implementado
Escolher Ambiente	Alta	Implementado
Visualizar histórico de resultados	Baixa	Não Implementado

Tabela 6.3: Funcionalidades 3D do Jogo

Requisitos	Prioridade	Status
Jogar	Alta	Implementado
Visualizar Objetos	Média	Não implementado
Salvar jogo	Média	Implementado
Visualizar Resultado	Alta	Implementado
Terminar Partida	Alta	Implementado
Reiniciar Partida	Média	Implementado
Abandonar Partida	Baixa	Implementado

6.2 Trabalhos Futuros

Como aprimoramento do jogo certas dependências de software podem ser eliminadas como é o caso do uso do framework JMF, que exige algumas configurações de sistema que podem ser um pouco complexas para alguns usuários, além do uso do SGBD PostgreSQL, que oferece uma série de recursos que não são utilizados pelo sistema, além da dependência estabelecida pelo mesmo em relação a sua instalação no computador o usuário. Em particular, para esse caso, pretende-se adotar o SQLite que pode ser incorporado ao projeto eliminando a dependência da instalação de um software no computador do usuário sem que sejam necessárias muitas mudanças na implementação do jogo.

Segundo (SQLITE, 2012) SQLite é uma biblioteca de software que atualmente é a escolha mais popular de motor de banco de dados para celulares, PDAs, MP3 *players*, *set-top boxes* e outros aparelhos eletrônicos, pois possui um código pequeno realizando um eficiente gerenciamento de espaço de memória e de disco, além de ser altamente confiável e não requerer manutenção de um Administrador de Banco de Dados.

Além das dependências de sistema uma solução referente à execução de vídeos ou animações durante a interação 3D deve ser analisada, pois como a *engine* utilizada não tem suporte nativo a vídeos, um modelo animado representando um intérprete para o usuário pode ser uma solução, porém deve ser bem estudada, pois os movimentos executados pelo modelo devem ser os mais reais possíveis.

A partir de uma reunião recente com professores do CAS foram estipuladas algumas melhorias que devem ser agregadas na interface ou implementadas como novas funcionalidades (*re-design*), como aprimoramento:

- Adequação de palavras que representam funcionalidades na interface, como é o caso da palavra “criar conta”, presente na tela inicial que deverá ser substituída pela palavra “começar”. Essa mudança será realizada em todas as palavras presentes na interface que não possuem um significado claro para o usuário surdo.
- Para palavras que expressam substantivos, como um nome próprio, cada letra deverá ser traduzida para seu sinal em Libras, devido a não correspondência do sinal para tradução do nome em Libras.
- Reorganização de informação para a tela de escolha de ambientes, onde para cada ambiente deverá ser agregado uma imagem correspondente à sua aparência externa. O nome do ambiente e a pontuação do aluno deverão ser apresentados junto à imagem de representação do ambiente e não no lado direito da interface, sofrendo alterações para cada novo ambiente selecionado. Para ambientes completos, sobre a imagem que representa o ambiente deve ser agregada uma imagem de um sinal positivo e a cor de seleção do ambiente deve ser verde indicando sucesso, já para um ambiente incompleto a cor de seleção deve ser vermelha para chamar a atenção do aluno, visto que esse ambiente ainda está pendente e deve ser completado. Ambientes não liberados devem continuar sendo representados por cadeados fechados.
- Toda a mensagem de erro deve possuir um *gif* animado representando um intérprete, para que o aluno consiga entender o que ele fez de errado, ou quais problemas foram apontados pelo jogo.
- Todos os *gifs* devem oferecer uma forma de controle para animação, no caso o clique, ou seja, a animação só deve ser ativada quando o usuário clicar sobre o *gif*, que deve apresentar a animação uma vez; caso o usuário queira ver novamente a animação deverá clicar sobre o *gif* novamente.

- Ao clicar em um *gif* o mesmo deve se expandir para que o usuário consiga melhor visualizar a mensagem transmitida por ele.
- Para a tela de escolha de personagens cada menu de item, sendo roupa, cabelo e rosto, devem ter consigo uma imagem agregada para indicar o significado da palavra em sua forma visual.
- Ao selecionar uma cor na tela de criação de personagem, seja para a roupa, cabelo ou rosto, essa cor além de ser atualizada no item correspondente do personagem exibido, deverá também ser atualizada no menu de escolha, ou seja, todos os tipos de item disponíveis para escolha devem ter sua cor alteradas para a escolhida.
- Na parte 3D a opção de salvar o jogo deve ser agregada a funcionalidade sair do jogo, sendo assim o jogo deve apenas apresentar na tela de opções de jogo as opções “reiniciar” e “sair e salvar”. A opção abandonar deve ser retirada da tela de opções de jogo, pois além de agregar mais informação do que necessário para tela, deixa a responsabilidade de atualização de avanços para o aluno, o que pode induzir o mesmo a um erro que fatalmente irá descartar todos os avanços realizados durante a interação.
- Elaboração de uma listagem contendo todas as mensagens presentes no jogo para que os professores do CAS consigam traduzi-las para Libras em forma de vídeos, os quais serão agregados ao jogo. As mensagens passadas aos professores do CAS devem ser simplificadas, ou seja, devem ser mais objetivas possíveis, tanto para redução do tamanho do arquivo de vídeo, como para compreensão do aluno.
- Na parte 3D do jogo é importante a presença de um marcador indicando qual objeto está selecionado. Ao selecionar um objeto correto a cor do marcador de seleção deve ser verde, indicando sucesso, no caso de erro a cor do marcador deverá ser vermelha.

Outro requisito está vinculado à portabilidade do jogo, ou seja, o jogo deve ser executável tanto na plataforma Windows (plataforma ora utilizada) como na plataforma Linux. Para prover esse requisito foram utilizadas somente software e *framework* livres, como é o caso da adoção da linguagem Java, da *engine* jMonkey, entre outros. Contudo, mesmo com o uso de tecnologias livres não é possível dizer que o jogo é totalmente portátil. Dessa forma, como trabalho futuro será realizado um teste de portabilidade, visando avaliar o comportamento do jogo durante sua execução na plataforma Linux.

Quando o jogo estiver consolidado, espera-se avaliar sua usabilidade com usuários reais, para identificar eventuais problemas de uso e propor melhorias. E, por fim, espera-se também distribuir o jogo Conceitos na Internet.

Apêndice A

Tecnologias Utilizadas

Este apêndice tem por objetivo expor detalhes das configurações necessárias para desenvolver e executar o jogo Conceitos. Dessa forma serão apenas especificados detalhes sobre como configurar o ambiente de programação Netbeans para suportar as bibliotecas, ferramentas e *frameworks* utilizados no desenvolvimento do jogo. Conceitos referentes a iluminação de objetos, modelagem, aplicação de texturas, entre outros, fogem do escopo desse documento, ou seja, sendo apenas citados, caso necessário.

O presente documento se divide em três partes distintas, referentes às funcionalidades básicas providas pelo jogo Conceitos, sendo (i) Configurando o Netbeans e o jMonkey, que tem por objetivo especificar os detalhes necessários para que as funcionalidades do *Sun Development Kit* (SDK) jMonkey 3.0 Beta possam ser agregadas ao *Integrated Development Environment* (IDE) Netbeans 7.2; (ii) JMF e fobs4, que traz o detalhamento de como configurar o ambiente de programação e o sistema operacional para que aplicações Java consigam suportar a execução de vídeos; e (iii) Blender e Ogre3D, que explica como exportar e modelos tridimensionais elaborados em Blender 2.5.9 em um formato compatível com os modelos suportados pela *engine* jMokey.

A.1 Configurando o Netbeans e o jMonkey

A *engine* jMoneky segundo jMonkey (2012) é um *kit* de desenvolvimento direcionado a elaboração de jogos com base na linguagem de programação Java. Possui uma documentação rica em detalhes, além de vários exemplos e fóruns de ajuda *online* onde vários programadores compartilham suas experiências. O SDK da *engine* (uma plataforma de desenvolvimento exclusiva da *engine*, semelhante a IDE Netbeans) combina todas as bibliotecas jME3 e algu-

mas ferramentas de desenvolvimento que facilitam o desenvolvimento de jogos, principalmente a geração de código e o gerenciamento de projetos.

Devido à necessidade do jogo Conceitos de agregar em sua composição recursos não oferecidos pela *engine*, como a reprodução de vídeos, as bibliotecas jME3 foram incorporadas ao Netbeans. Essas bibliotecas oferecem todos os recursos presentes no SDK, como a manipulação de objetos 3D, gerenciamento de animações, aplicação de física em objetos, detecção de colisão, entre outros recursos que devem ser contemplados no desenvolvimento de um jogo onde a interação é baseada em objetos tridimensionais.

No jogo Conceitos desenvolvido nesse trabalho, foi utilizado para programação a IDE Netbeans 7.2, onde foram agregadas as bibliotecas do SDK jMonkey 3.0 Beta e da LWJGL 2.8.4. Dessa forma, o ambiente de programação Netbeans foi preparado para agregar no projeto Conceitos as funcionalidades básicas que o SDK jMonkey oferece para o desenvolvimento de jogos, além das especificidades da biblioteca gráfica LWJGL que permite a execução do aplicativo e sua comunicação com o sistema operacional (SO).

Abaixo serão mostrados os passos para configurar o Netbeans para que esse ofereça o suporte necessário para o desenvolvimento de jogos.

A. 1.1 Configuração do Ambiente de Trabalho

Faça o *download* das bibliotecas do jMonkey² e descompacte os arquivos em um diretório de sua preferência. Essas bibliotecas garantem a compilação do jogo desenvolvido por meio da *engine*. O mesmo deve ser feito para as bibliotecas LWJGL³, que oferecem suporte necessário para a execução do jogo.

No Netbeans selecione o *menu* arquivo e em seguida a opção novo projeto. Essa operação abrirá a janela apresentada na Figura A.1, em categorias escolha a opção Java, e em Projetos selecione Aplicação Java, ao final desse processo clique em próximo. Na nova tela (Figura A.2) defina um nome para o projeto, habilite a opção “Usar Pasta Dedicada para Armazenar Bibliotecas” e desabilite a opção “Criar Classe Principal”, e por fim clique no botão “Finalizar”. Dessa forma, será criado um novo projeto, onde serão agregadas as classes e bibliotecas do jogo.

2 <http://jmonkeyengine.com/nightly>: página web para download das bibliotecas da enginejMonkey.

3 <http://www.lwjgl.org/download.php>: página web para download das bibliotecas LWJGL.

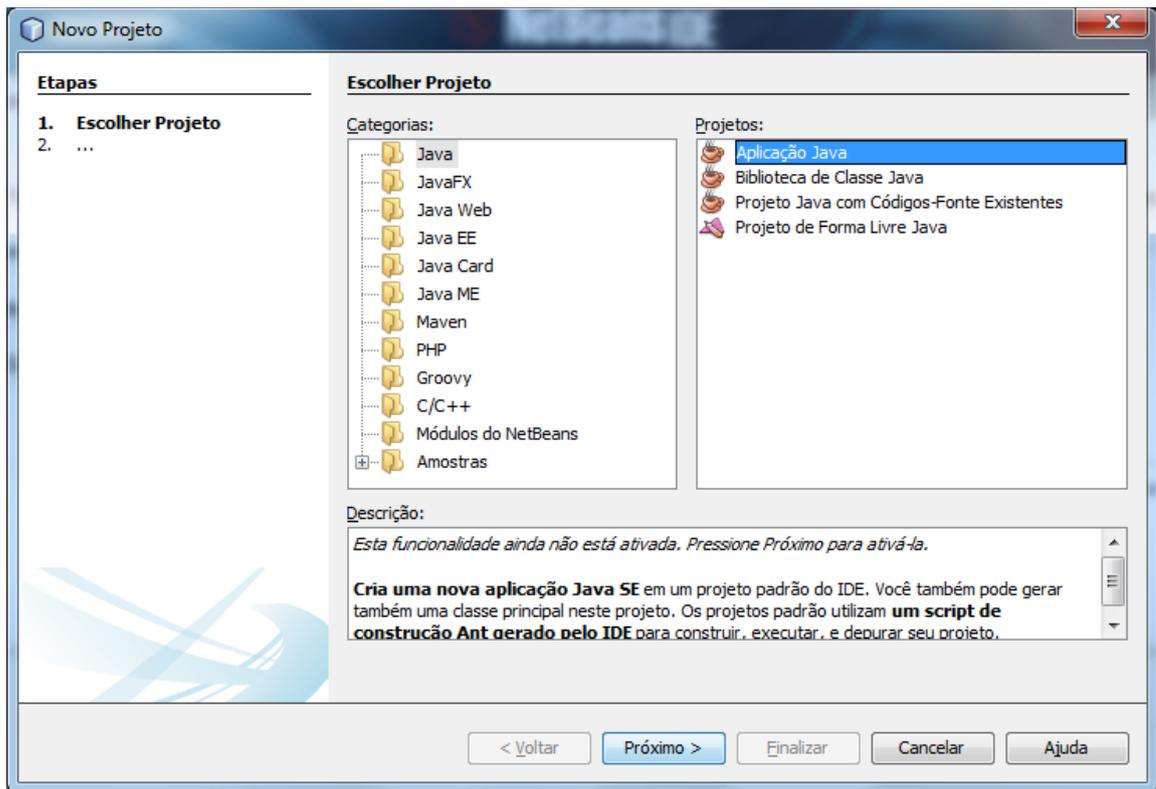


Figura A.1: Tela inicial para criação de um projeto Java

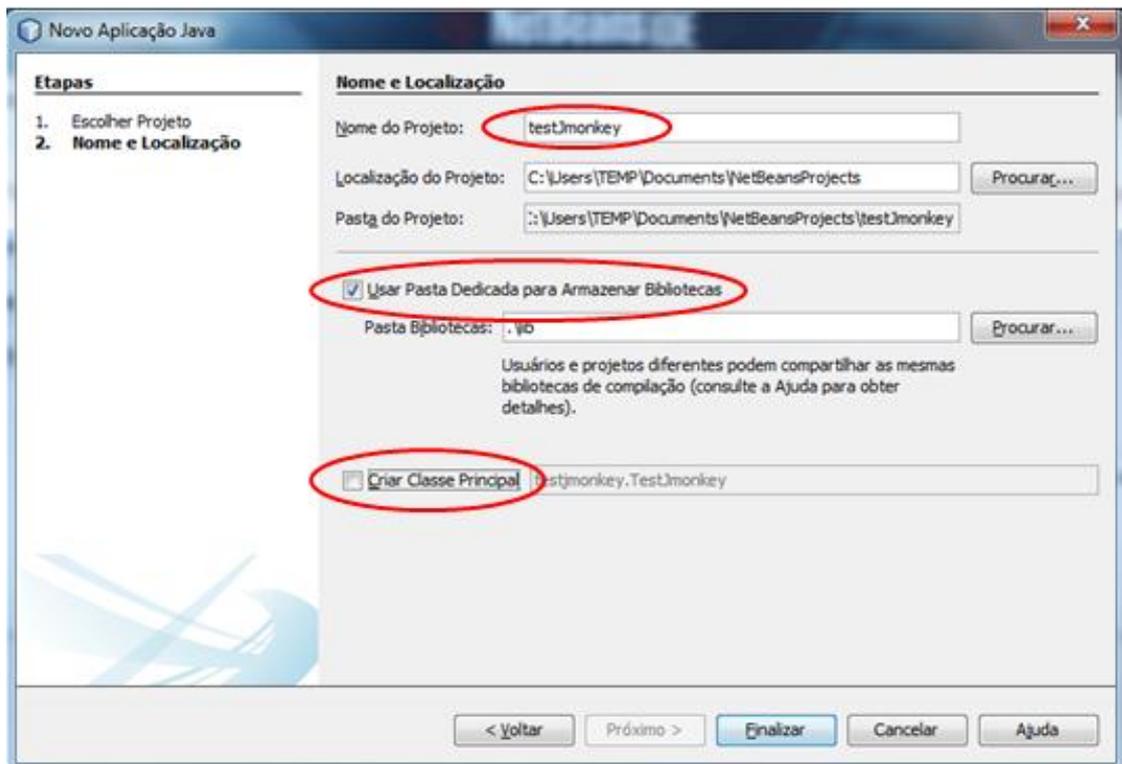


Figura A.2: Demonstra as opções que devem ser alteradas

Depois do projeto criado as bibliotecas devem ser adicionadas. Para adicioná-las selecione a aba projetos e na pasta bibliotecas, como mostra a Figura A.3, clique com o botão direito do *mouse*, e escolha a opção “Adicionar JAR/Pasta...”. Após execução dessa ação será exibida uma janela para selecionar quais bibliotecas serão adicionadas ao projeto. Nessa janela selecione as bibliotecas que foram descompactadas, primeiramente as do jMonkey e em seguida as da LWJGL. Para finalizar selecione todas as bibliotecas, marque a opção “Copiar para Pasta de Bibliotecas” e clique em abrir. As bibliotecas que devem ser adicionadas podem ser visualizadas na Figura A.4 e na Figura A.5, do jMonkey e da LWJGL, respectivamente.

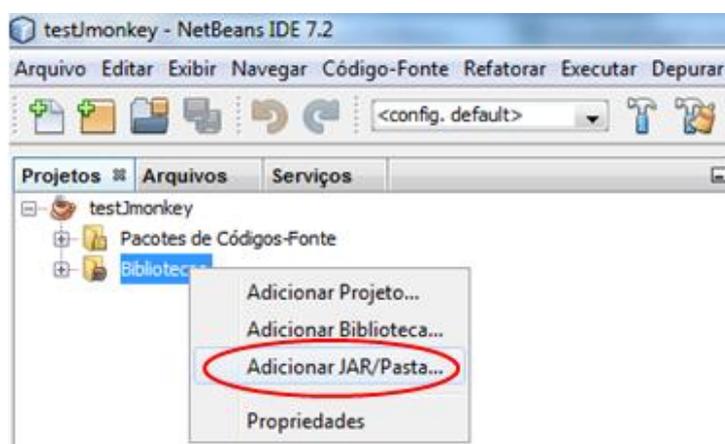


Figura A.3: Demonstra como adicionar bibliotecas externas ao projeto

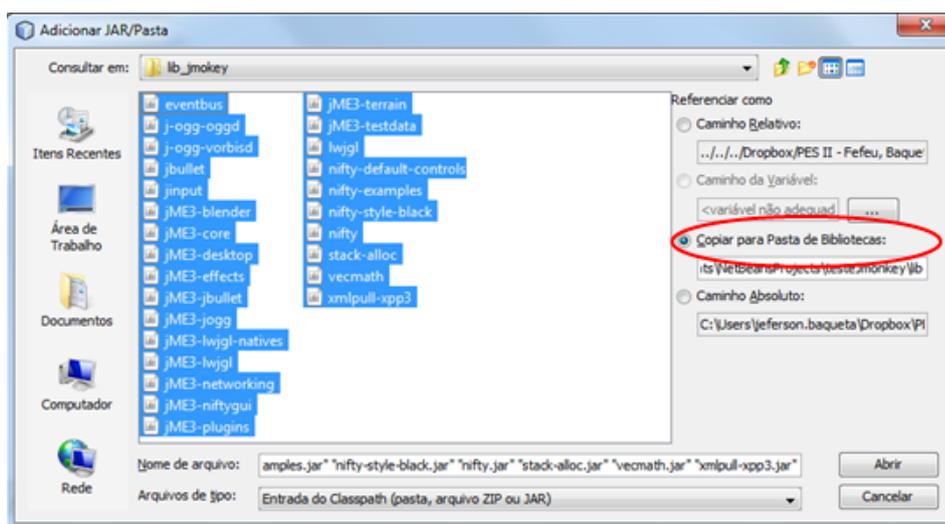


Figura A.4: Lista de bibliotecas do jMonkey

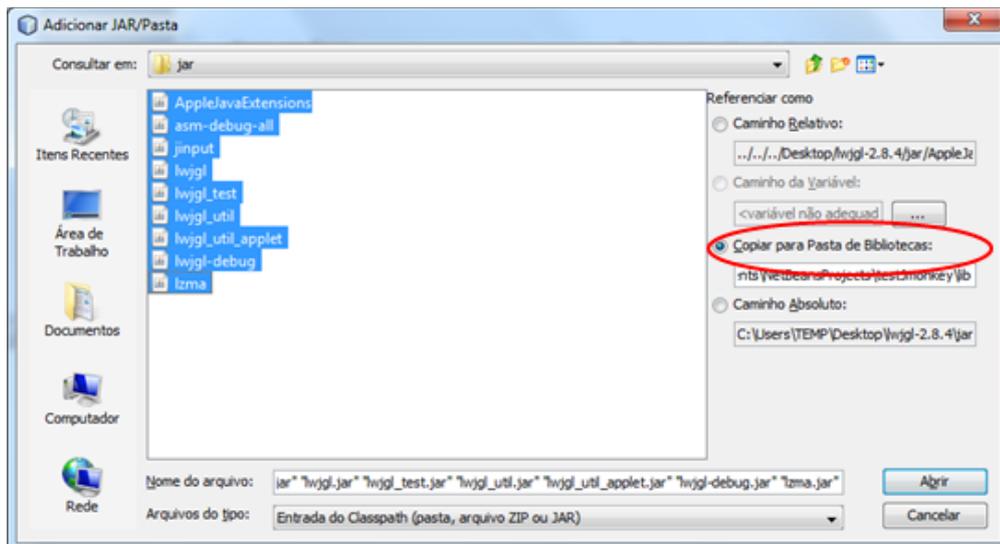


Figura A.5: Lista de bibliotecas da LWJGL

Dependendo da versão do Netbeans é necessário adicionar alguns arquivos nativos da LWJGL compatíveis com o SO, caso contrário, a execução do jogo estará comprometida. A versão 7.2 do Netbeans não necessita dos arquivos nativos da LWJGL para executar aplicações desenvolvidas com base em jMonkey, contudo caso seja necessário o processo de agregação dos arquivos nativos da LWJGL ocorre da seguinte forma:

- Clique com o botão direito sobre o projeto, na aba projetos, selecione a opção propriedades. Uma nova janela será exibida (Figura A.6), nessa tela selecione na área das categorias a opção “Executar” e na caixa de texto nomeada de “Opções de VM” informe o caminho dos arquivos nativos da LWJGL, esses arquivos estão no mesmo diretório das bibliotecas descompactadas anteriormente em uma pasta chamada *native*.
- É importante ressaltar que dentro da pasta *native* deve-se selecionar as bibliotecas referentes ao SO onde está instalado o IDE, no caso do exemplo Windows. A sintaxe utilizada para informar os arquivos nativos da LWJGL se apresenta da seguinte forma:
`-Djava.library.path=[caminho para a pasta LWJGL/native/seu sistema operacional].`

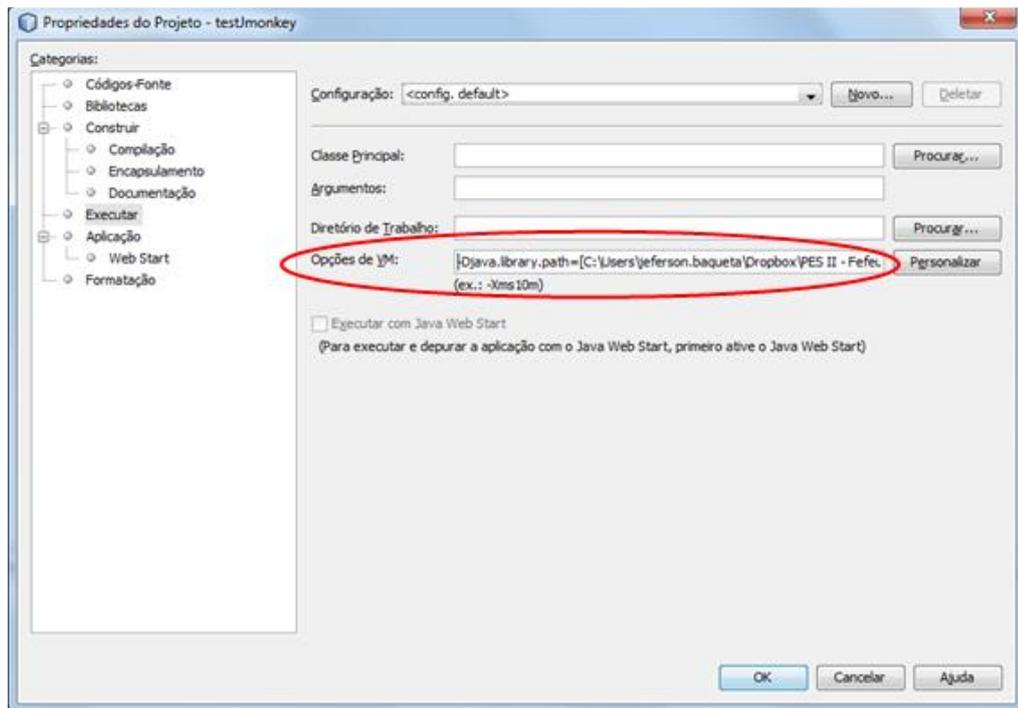


Figura A.6. Adicionando bibliotecas nativas da LWJGL

- Ao final desse processo clique no botão “OK” e o ambiente de programação estará pronto para o desenvolvimento, lembrando que esse ultimo passo só é necessário para versões do Netbeans anteriores à 7.2.

A.1.2 Teste de Execução

Para verificar se a configurações foram realizadas de forma correta clique em seu projeto com o botão direito, crie um novo pacote com o nome que desejar e clique nesse no novo pacote; crie uma classe com o nome PrimeiroJogo, e em seguida utilize o código abaixo para realizar o teste de execução:

- Bibliotecas importadas:

```

7  import com.jme3.app.SimpleApplication;
8  import com.jme3.material.Material;
9  import com.jme3.math.ColorRGBA;
10 import com.jme3.math.Vector3f;
11 import com.jme3.scene.Geometry;
12 import com.jme3.scene.shape.Box;

```

- Classe Primeiro Jogo:

```
23 public class PrimeiroJogo extends SimpleApplication {
24
25     public static void main(String[] args) {
26         PrimeiroJogo app = new PrimeiroJogo();
27         app.start();
28     }
29
30     @Override
31     public void simpleInitApp() {
32
33         Box cubo = new Box(Vector3f.ZERO, 1, 1, 1);
34         Geometry geometria = new Geometry("Box", cubo);
35         Material material = new Material(assetManager, "Common/MatDefs/Misc/Unshaded.j3md");
36         material.setColor("Color", ColorRGBA.Blue);
37         geometria.setMaterial(material);
38         rootNode.attachChild(geometria);
39     }
40 }
41 }
```

Como não foi estipulada nenhuma classe base para o projeto, basta pressionar as teclas SHIFT+F6 para executar o arquivo, o resultado dessa ação será a exibição da tela padrão do jMonkey para configuração do *display* do jogo (Figura A.7), após a confirmação das opções o resultado do código acima será uma janela com o cubo azul em seu interior como exibido na Figura A.8.

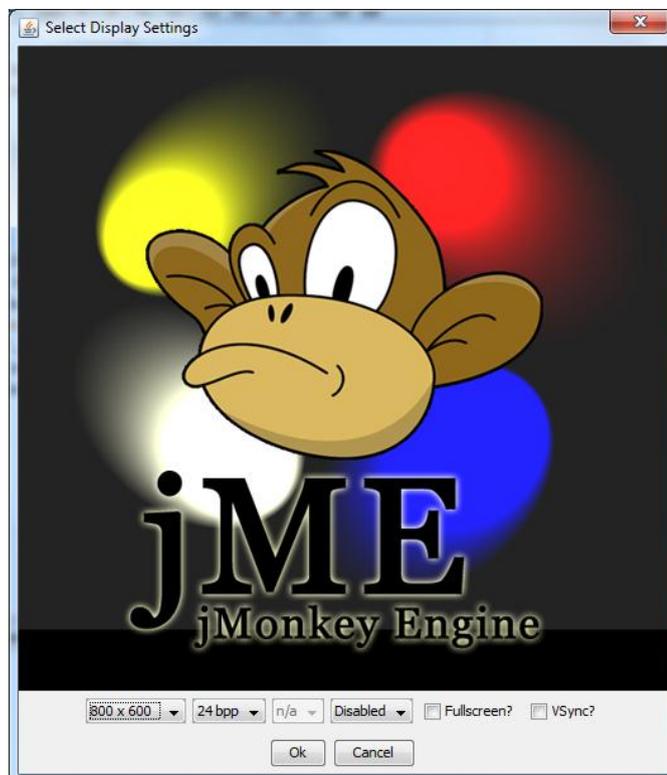


Figura A.7: Tela padrão do jMonkey para *display*

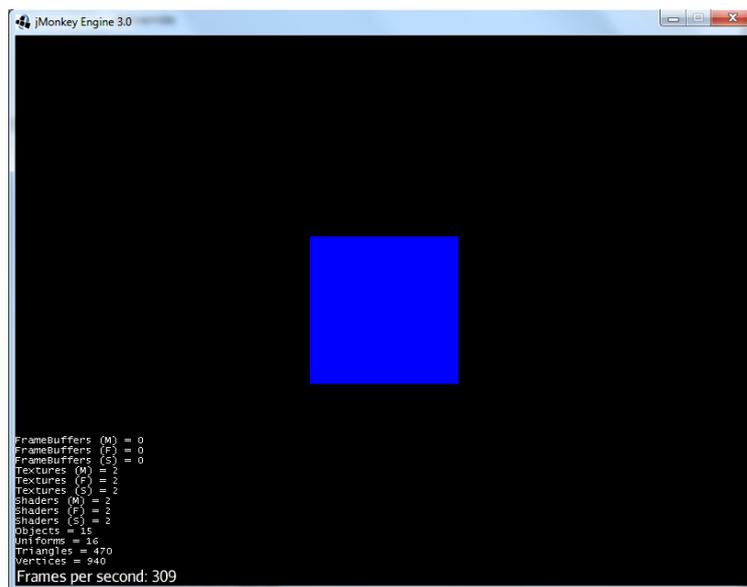


Figura A.8: Resultado do código de configuração proposto

É importante ressaltar que caso a execução não ocorra, há que verificar o tipo de erro ocorrido, porém caso o erro informado seja relacionado ao formato dos *pixels*, o vídeo do computador pode estar desinstalado ou não atualizado, o que pode ser resolvido com a instalação ou atualização do *drive* de vídeo.

A.2 JMF e fobs4

O *Java Media Framework* (JMF) e o *fobs4* são duas ferramentas que agregadas ao Java permitem o desenvolvimento de aplicações capazes de reproduzir vídeos em vários formatos, como o formato *avi*. Devido à necessidade de exibir vídeos em Libras para auxiliar na interação entre o jogo e o jogador, essas ferramentas foram agregadas a parte 2D do jogo *Conceitos*, especificamente na fase cadastro do usuário.

O JMF em sua versão 2.1.1⁴, utilizada no jogo, oferece a reprodução de alguns tipos de vídeos específicos, sendo bem limitada, pois alguns vídeos com o mesmo formato executam no *player* disponibilizado pelo *framework* e outros não. Para corrigir os problemas apresentados pelo JMF deve-se instalar o *plugin* *Fobs4jmf_0.4.2*⁵, esse que potencializa as funcionalidades

⁴<http://www.oracle.com/technetwork/java/javase/download-142937.html>: site da Oracle que disponibiliza o download do *jmf* em sua versão 2.1.1.

⁵<http://sourceforge.net/projects/fobs/files/fobs4jmf%20binaries/Fobs4JMF-0.4.2>: site para download do *plugin* *fobs4jmf_0.4.2*.

do *framework*, além de aumentar a quantidade de formatos, tanto de áudio quanto de vídeos que podem ser reproduzidos pelo JMF.

O grande problema da utilização dessas ferramentas é a grande dependência que oferecem para aplicação desenvolvida em relação a máquina e SO, pois para que essas ferramentas funcionem de forma adequada devem ser criadas variáveis de ambientes no sistema operacional que irão indicar ao aplicativo o caminho das bibliotecas utilizadas, para isso as ferramentas devem estar instaladas no computador, caso contrário a aplicação não será capaz de reproduzir os vídeos.

A.2.1 Criando uma Variável de Ambiente

Para utilizar tanto o JMF com o Fobs4, deve-se realizar o *download* dos mesmos nos endereços supracitados. Primeiramente deve-se criar uma variável de ambiente no sistema operacional instalado na máquina chamada CLASSPATH, onde serão adicionados os caminhos dos arquivos .jar referentes ao *framework* e o *plugin* que serão utilizados. O processo de criação dessa variável acontece da seguinte forma, lembrando que esse processo está sendo realizado em ambiente Windows:

- Em iniciar com o botão direito sobre “Computador” escolha a opção propriedades, como mostra a Figura A.9.

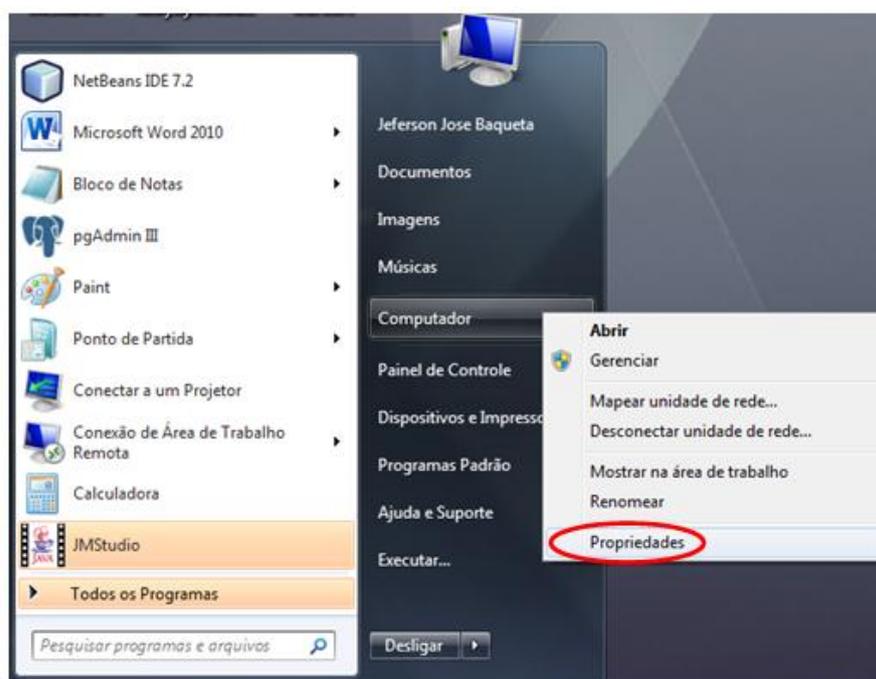


Figura A.9: Acessando as propriedades do Computador

- Após essa ação será exibida uma tela sobre as configurações básicas do computador (Figura A.10). Nessa tela, no canto esquerdo inferior, selecione a opção “Configurações avançadas do sistema”.

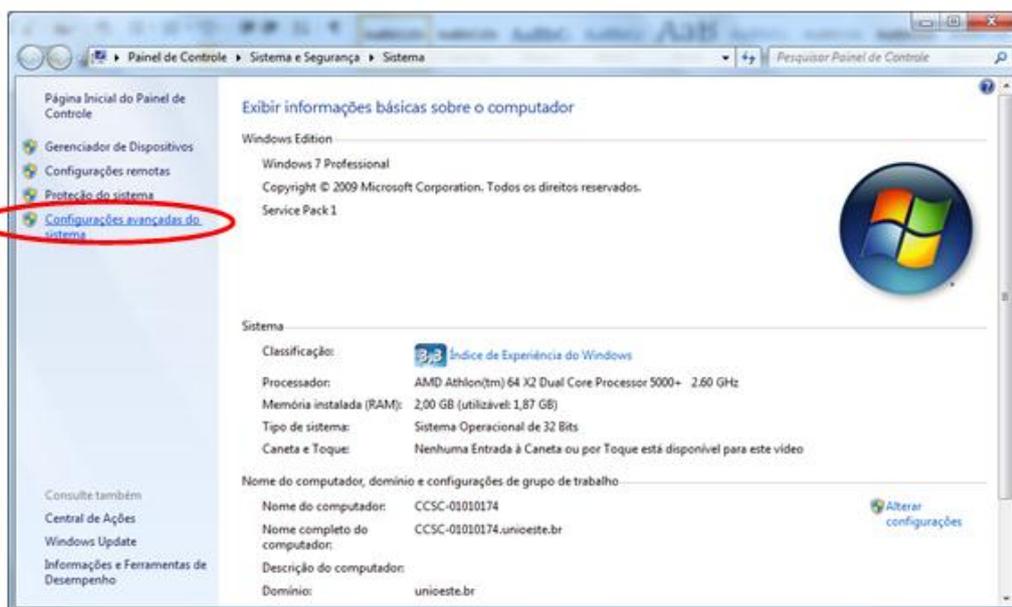


Figura A.10: Configurações do sistema

- Esse processo dará origem a uma janela de propriedades, onde se deve selecionar a opção “Variáveis de Ambiente...”, como mostrado na Figura A.11.

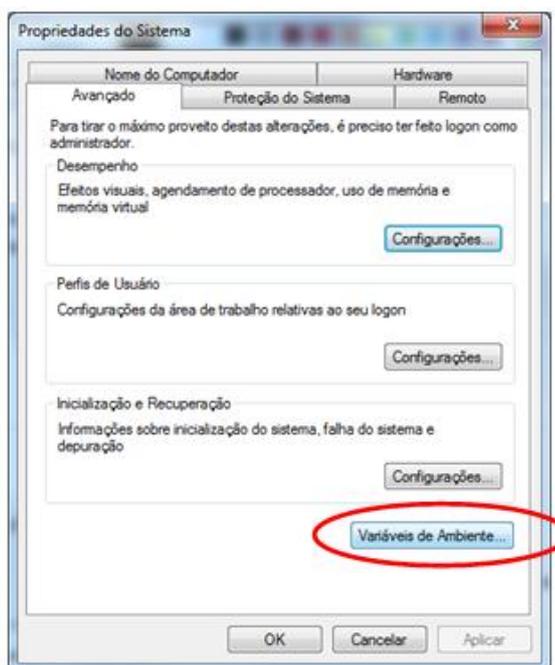


Figura A.11: Propriedades do Sistema

- A próxima tela é efetivamente a janela de configuração de variáveis de ambiente, e para criar uma nova variável, clique no botão “novo”. Em seguida será exibida uma pequena janela onde deve ser informado o nome da variável de ambiente e o valor da mesma, sendo respectivamente CLASSPATH e “;”, como mostrado na Figura A.12, após, basta clicar em “OK” e a variável estará criada.

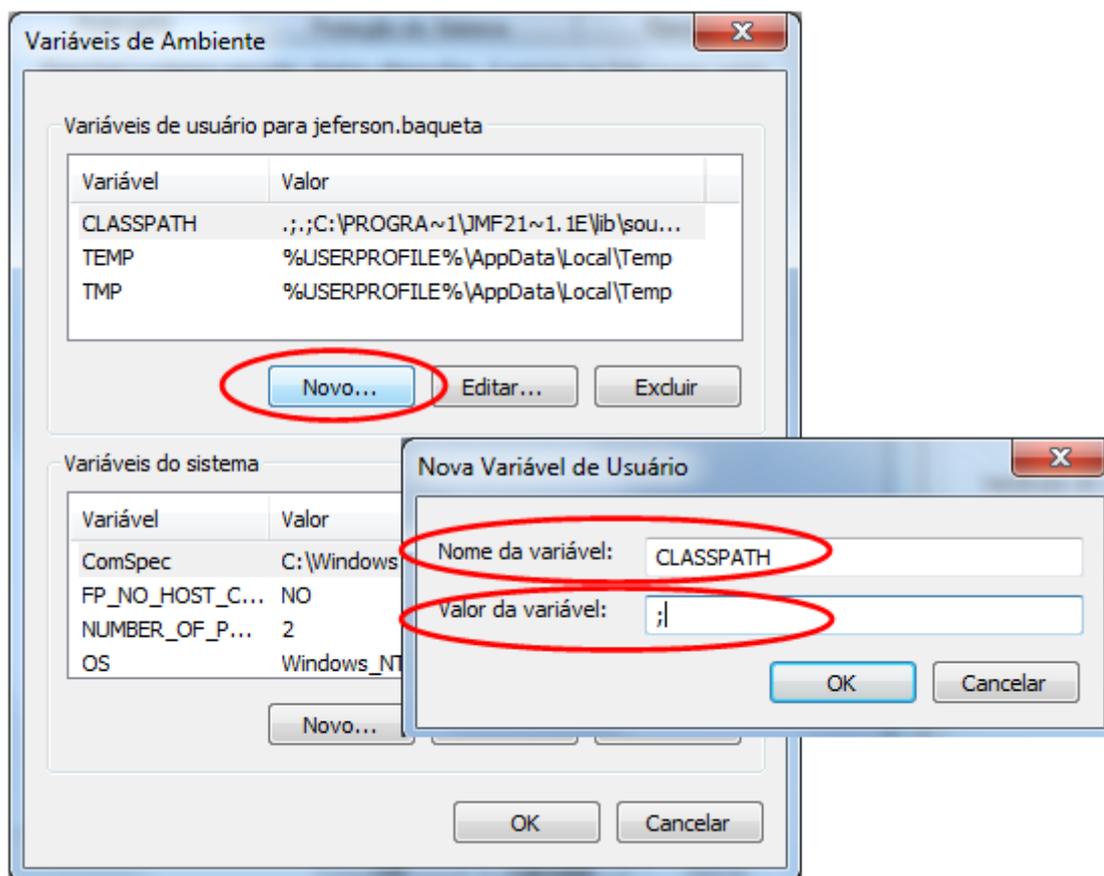


Figura A.12: Variáveis de ambiente

A.2.2 Instalação do JMF e Configuração do fobs4

Após a criação da variável CLASSPATH, descompacte ambos os arquivos, JFM e fobs4. Primeiramente instale o arquivo jmf-2_1_1e-windows-i586 .Nesse exemplo esse arquivo é específico para Windows 32bits, porém, no site da Oracle o *framework* é disponibilizado também para outras plataformas. A instalação do *framework* é relativamente simples, sendo apenas necessário avançar entre as três telas apresentadas durante o processo de instalação e reiniciar o computador ao final. Esse processo não será especificado por meio de figuras, dada a sua simplicidade.

Instalado o JMF no computador deve-se adicionar o *plugin* fobs4 ao diretório onde foi instalado o *framework*; para tal, utiliza-se dos arquivos descompactados do fobs4, ou seja, copie todos os arquivos do fobs4, mostrados na Figura A.13, e os cole no diretório do JMF, possivelmente no endereço C:\Program Files\JMF2.1.1e\lib, os arquivos já existentes devem ser substituídos. A Figura A.14 ilustra esse processo.

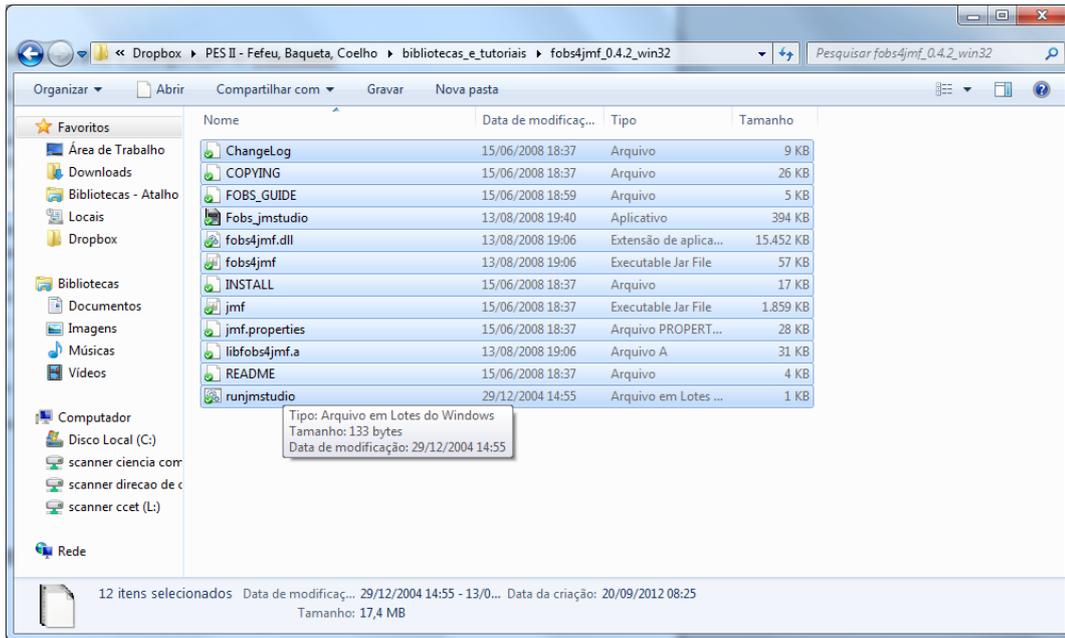


Figura A.13: Arquivos do fobs4 que devem ser copiados

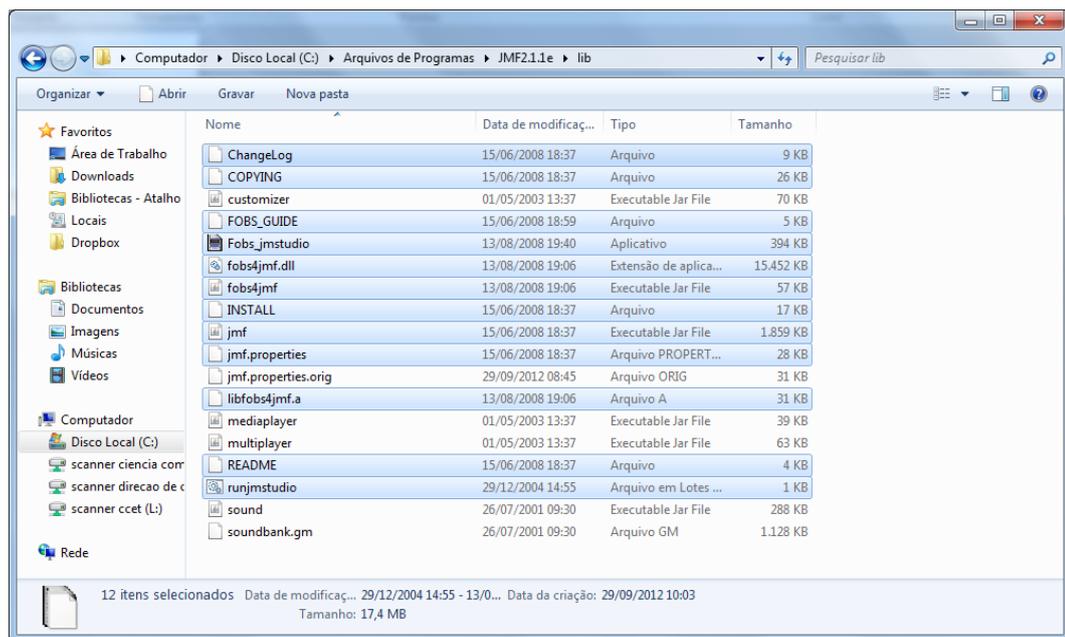


Figura A.14: *Plugin* fobs4 adicionado ao diretório do JMF

Após a adição do *plugin*, o arquivo `fobstjmf.dll` no diretório do JMF deve ser copiado para o diretório `C:\Windows\System32`. Em seguida o caminho do arquivo `fobs4jmf` no diretório do JMF deve ser informado para a variável de ambiente `CLASSPATH` já criada. Para fazer isso é só realizar a sequência de passos para adicionar uma variável de ambiente, com a diferença de que agora se deve editar o valor da variável existente especificando o caminho do arquivo `fobs4jmf` ao final da especificação do caminho do arquivo `jmf`, como apresentado na Figura A.15. O caminho do `fobs4` deve ser adicionado depois do último “;” presente na caixa de texto que representa valor da variável, e ainda no arquivo `fobs4`, deve-se adicionar a extensão `.jar`.

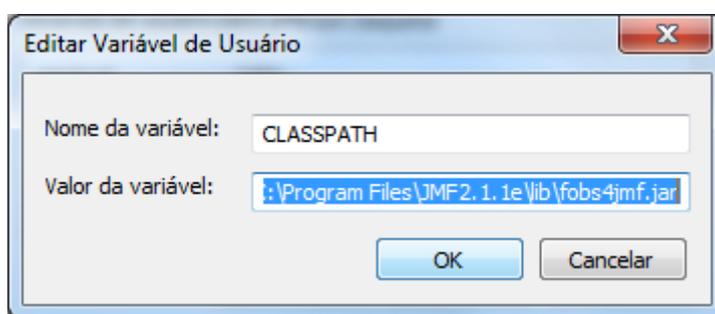


Figura A.15: Caminho do *plugin* `fobs4` adicionado a variável de ambiente `CLASSPATH`

A.2.3 Teste de Execução do JMF e do `fobs4`

No mesmo projeto utilizado para testar a *engine* `jMonkey` adicione as bibliotecas que estão no diretório do JMF, como indicado na Figura A.16, sendo que para essas bibliotecas a opção “Caminho absoluto” deve estar marcada.

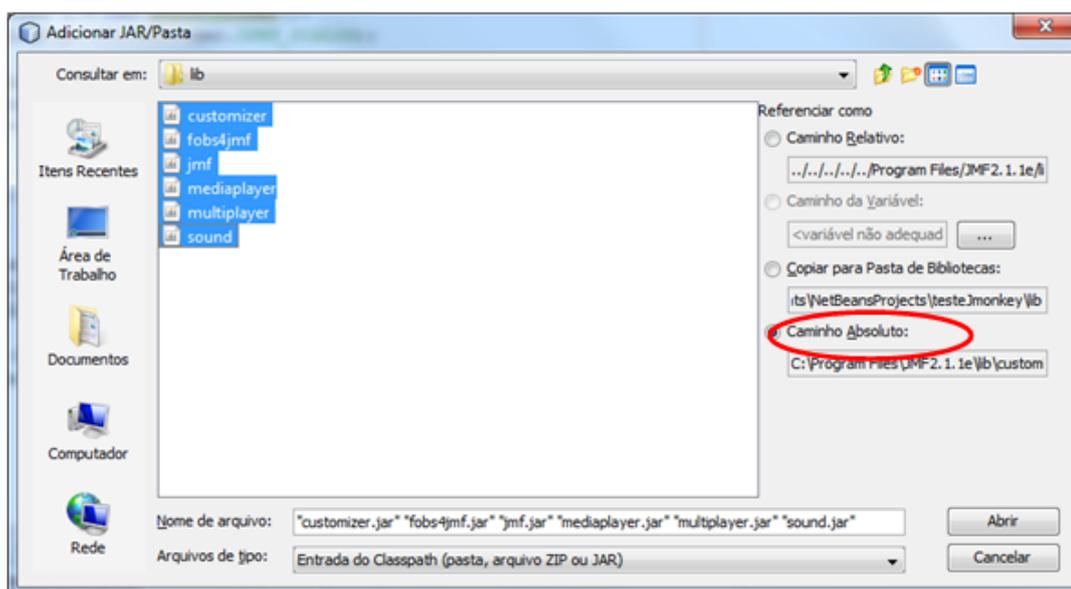


Figura A.16: Bibliotecas utilizadas para execução do `jmf` e do `fobs4`

A partir das bibliotecas adicionadas crie duas novas Classes, uma com o nome MediaPlayer (Figura A.17), que será estendida para um painel, e outra com o nome AbrirVideo (Figura A.18), sendo essa a classe principal desse exemplo.

```
5 package jogo;
6
7 import java.awt.BorderLayout;
8 import java.awt.Component;
9 import java.io.IOException;
10 import java.net.URL;
11 import javax.media.CannotRealizeException;
12 import javax.media.Manager;
13 import javax.media.NoPlayerException;
14 import javax.media.Player;
15 import javax.media.Time;
16 import javax.swing.JPanel;
17
18 /**
19  *
20  * @author Jefferson Baqueta
21  */
22 public class MediaPlayer extends JPanel{
23
24     private Player player;
25
26     public MediaPlayer(URL url) {
27         Manager.setHint(Manager.LIGHTWEIGHT_RENDERER, true);
28
29         try {
30             createPlayerAndShowComponents(url);
31         } catch (NoPlayerException noPlayerException) {
32             System.err.println("Formato de video não compatível");
33         } catch (CannotRealizeException cannotRealizeException) {
34             System.err.println("O sistema não possui os recursos necessários para reproduzir esse video");
35         } catch (IOException IOException) {
36             System.err.println("Erro na leitura do arquivo arquivo");
37         }
38     }
39
40
41     private void createPlayerAndShowComponents(URL url) throws IOException, NoPlayerException, CannotRealizeException {
42         player = Manager.createRealizedPlayer(url);
43
44         Component comp;
45         if ((comp = player.getVisualComponent()) != null) {
46             add(comp, BorderLayout.CENTER);
47         }
48         if ((comp = player.getControlPanelComponent()) != null) {
49             add(comp, BorderLayout.SOUTH);
50         }
51
52         player.start();
53         player.setStopTime(new Time(0.1));
54     }
55 }
```

Figura A.17: Classe MediaPlayer

```

7 import java.awt.Dimension;
8 import java.awt.Frame;
9 import java.io.File;
10 import java.net.MalformedURLException;
11 import java.util.logging.Level;
12 import java.util.logging.Logger;
13 import javax.swing.JFileChooser;
14
15 /**
16  *
17  * @author jeferson.baqueta
18  */
19 public class AbrirVideo {
20
21     public static void main(String[] args) {
22
23         Frame janela = new Frame();
24
25         JFileChooser j = new JFileChooser();
26         j.setDialogType(JFileChooser.OPEN_DIALOG);
27
28         j.setDialogTitle("Abrir Arquivo");
29         j.setCurrentDirectory(new File("d:/Netbeans/SimuladorGerenciamentoMemoria/tracer/"));
30         int res = j.showOpenDialog(null);
31
32         if (res == JFileChooser.APPROVE_OPTION) {
33             try {
34                 MediaPlayer player = new MediaPlayer(j.getSelectedFile().toURI().toURL());
35                 janela.add(player);
36                 janela.setSize(new Dimension(320, 340));
37                 janela.setVisible(true);
38
39             } catch (MalformedURLException ex) {
40                 Logger.getLogger(AbrirVideo.class.getName()).log(Level.SEVERE, null, ex);
41             }
42         } else {
43             System.out.println("erro ao iniciar iniciar Chooser");
44         }
45     }
46 }

```

Figura A.18: Classe principal AbrirVideo

Ao executar a classe AbrirVideo será exibida uma janela para busca do vídeo em um diretório do computador (Figura A.19), e preferencialmente, o vídeo deve estar no formato avi. O vídeo será exibido em um painel com dimensão de 320x340 (Figura A.20), sendo essa configuração editável.

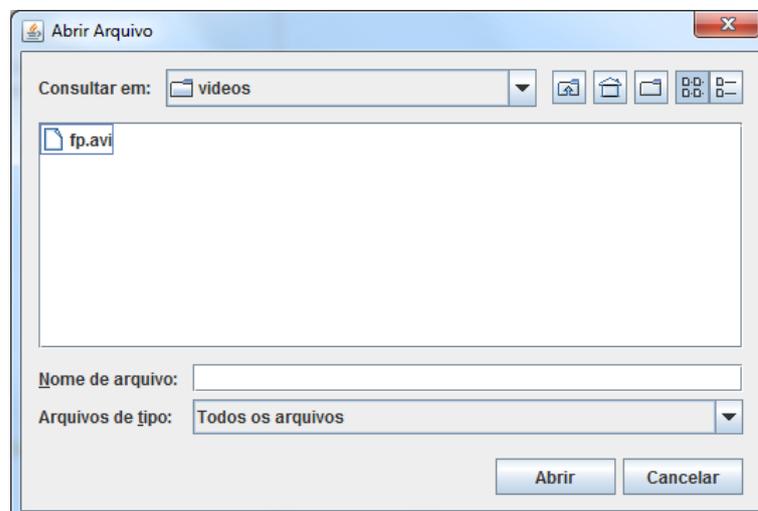


Figura A.19: Janela para seleção do vídeo

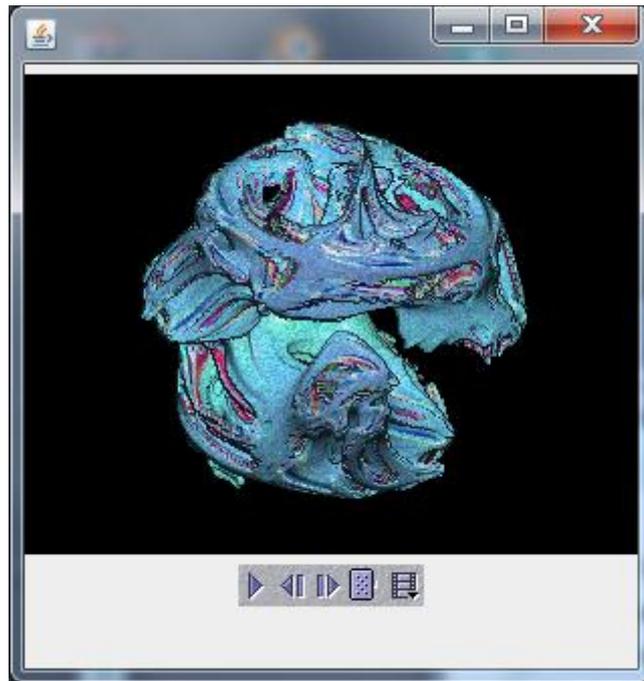


Figura A.20: Resultado da execução da classe AbrirVideo

A.3 Blender e Ogre3D

Nos jogos 3D o principal instrumento de interação entre o jogo em si e o jogador são objetos tridimensionais, ou modelos gráficos criados por meio de modeladores 3D. No caso do jogo Conceitos os modelos utilizados, objetos e cenários, foram desenvolvidos em Blender 2.59. Esse modelador foi escolhido justamente por ser completo e oferecer várias funcionalidades que facilitam a modelagem, como aplicação e edição de mapas UV (mapeamento de imagens para faces de malhas poligonais) juntamente com editores de imagens, edição de normais, controles de animações, entre outras funcionalidades, além de ser uma ferramenta *open source*.

Porém, há a desvantagem de que um modelo feito em Blender não pode ser puramente exportado para jMonkey, pois o formato `.blender` não é compatível com os formatos suportados pela *engine* jMonkey.

A.3.1 Configurando o Ogre3D

Para realizar a comunicação entre o modelador e a *engine* é necessário que seja utilizado o *plugin* Ogre3D desenvolvido em Python e vinculado ao Blender permitindo a exportação de modelos elaborados no mesmo para a *engine* jMonkey.

Para habilitar as opções de importações do Ogre3D no Blender é necessário configurá-lo. No caso do Blender 2.59 o *puglin* deve ser adicionado junto à pasta de *plug-ins* do modelador. O processo de configuração e os detalhes da exportação estão descritos abaixo.

- Faça o *download* [pluginblender2ogre-0.5.56](http://code.google.com/p/blender2ogre/downloads/list), que é adequado às versões do Blender com prefixo 2.5, e em seguida descompacte o arquivo `io_export_ogreDotScene.py`. Esse arquivo é efetivamente o *plugin* desenvolvido em Python que irá realizar a exportação.
- Copie e cole o arquivo extraído no seguinte diretório: “C:\Program Files\Blender Foundation\Blender\2.59\scripts\addons”. Este será agregado aos demais *plug-ins* nativos do Blender o *plugin* Ogre3D, que permitirá exportar os modelos para jMonkey, pois irá mapear todas as informações da malha, vértices, material e textura para arquivos especiais que serão lidos e interpretados pela *engine*.
- Abra o Blender instalado em sua máquina, e clique no *menu File* e escolha a opção “*User Preferences*” como na Figura A.21. Finalizada esta operação será aberta a janela de preferências do Blender. Nessa janela na aba *add-Nos* faça uma busca pela palavra “ogre”, e serão listados todos os *plug-ins* instalados no Blender que possuem alguma relação com a palavra consultada. No caso desse exemplo aparecerá apenas um *plugin* chamado “OgreExporter” que deve ser selecionado, como mostrado na Figura A.22, após selecione a opção “*Save as Default*” e o Blender voltará à tela inicial. Agora ao acessar o *menu File*, no sub-item “*export*” aparecerá a opção de exportação “ogre3d (scene and mesh)”, assim como na barra superior do Blender aparecerá um botão para habilitar algumas opções de gerenciamento das funcionalidades do formato Ogre3D, como visto na Figura A.23.

⁶ <http://code.google.com/p/blender2ogre/downloads/list> - site para *download* do *puglin* Ogre3D para Blender 2.59.

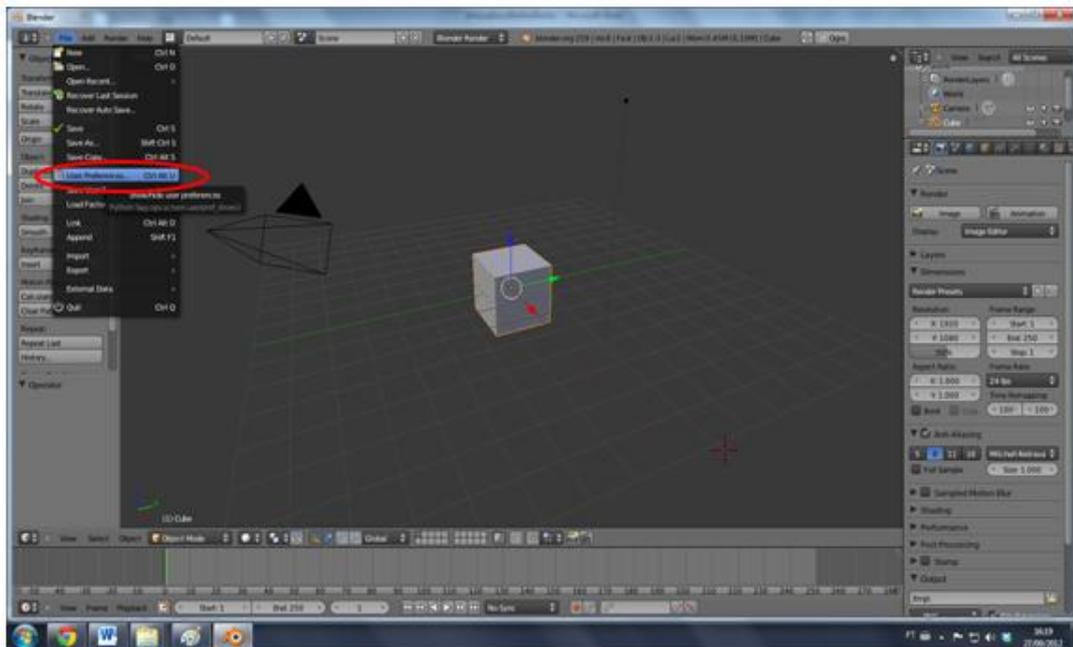


Figura A.21: Acessando a opção *User Preference*

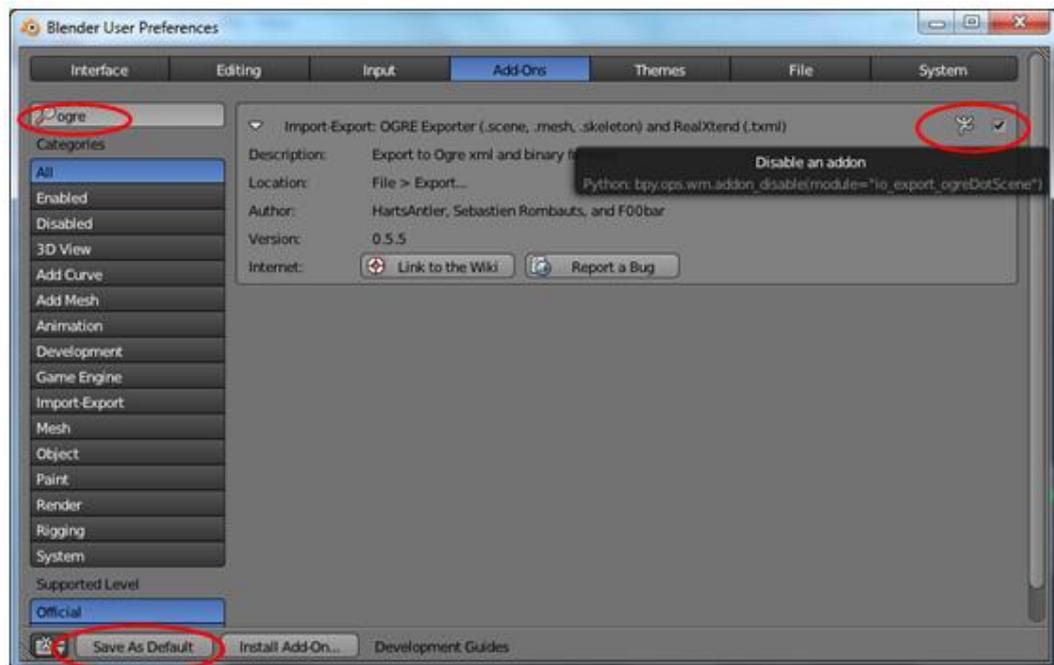


Figura A.22: Janela para adicionar *plug-ins*

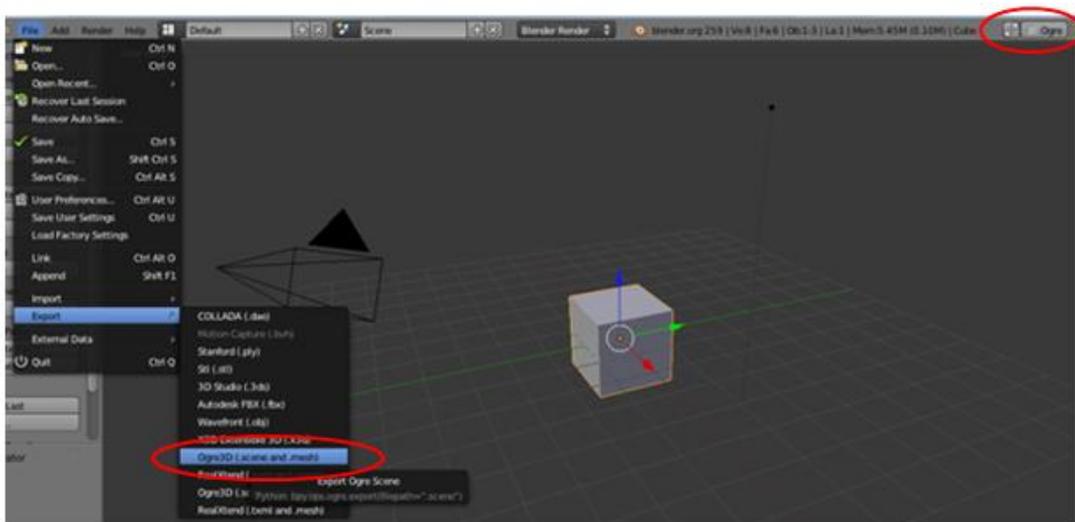


Figura A.23: Janela do Blender já configurada para exportação Ogre3D

A.3.2 Como Exportar Objetos

É importante ressaltar que todo *plugin* Ogre3D é desenvolvido para uma versão específica do Blender sendo assim, caso os modelos sejam feitos em Blender 2.6 e mapeados para o Blender 2.5, e posteriormente exportados em formato ogre3D, possivelmente quando o modelo for carregado para *engine* jMonkey problemas relacionados a malha e a textura do modelo poderão ocorrer. Por esse motivo é recomendado que escolhida a versão do modelador e verificada a existência do *plugin* Ogre3D para a mesma, não se deve mudar a versão do modelador durante o processo de criação dos modelos tridimensionais.

Para realização de um teste de exportação vá ao *menu Add*, escolha a opção *Mesh* e em seguida clique sobre a opção *Monkey*, ao fazer isso o Blender irá carregar um modelo pronto, uma cabeça de macaco chamada Suzy, mascote da marca Blender.

Com o modelo gerado, para realizar a exportação basta escolher a aba *File* e selecionar o sub *menu exporte*, e clique sobre a opção “ogre3d”. Caso o modelo exportador possua alguma textura, essa deve ser vinculada ao modelo por meio do mapa UV, pois devido à limitação do jMonkey em suportar apenas um textura por malha as texturas compostas não são renderizadas de forma adequada. A utilização de mapas de imagens UV⁷ é uma forma de mapear uma imagem composta para várias faces de um objeto.

Após selecionar a opção “export ogre3d”, uma tela semelhante à apresentada na Figura A.24 será exibida. Nessa tela serão definidos o nome do arquivo .scenee o local no computa-

⁷ <http://jmonkeyengine.org/wiki/doku.php/jme3:terminology> – Aqui são relacionados conceitos básicos sobre Computação Gráfica que devem ser entendidos antes de qualquer implementação.

dor que o modelo será exportado. A título de exemplo o modelo utilizado receberá o nome de monkey, e será salvo na pasta monkey dentro do projeto anteriormente criado no Netbeans, e será salvo no seguinte diretório:

C:\Users\jeferson.baqueta\Documents\NetBeansProjects\testeJmonkey\src\assets\Models\Monkey

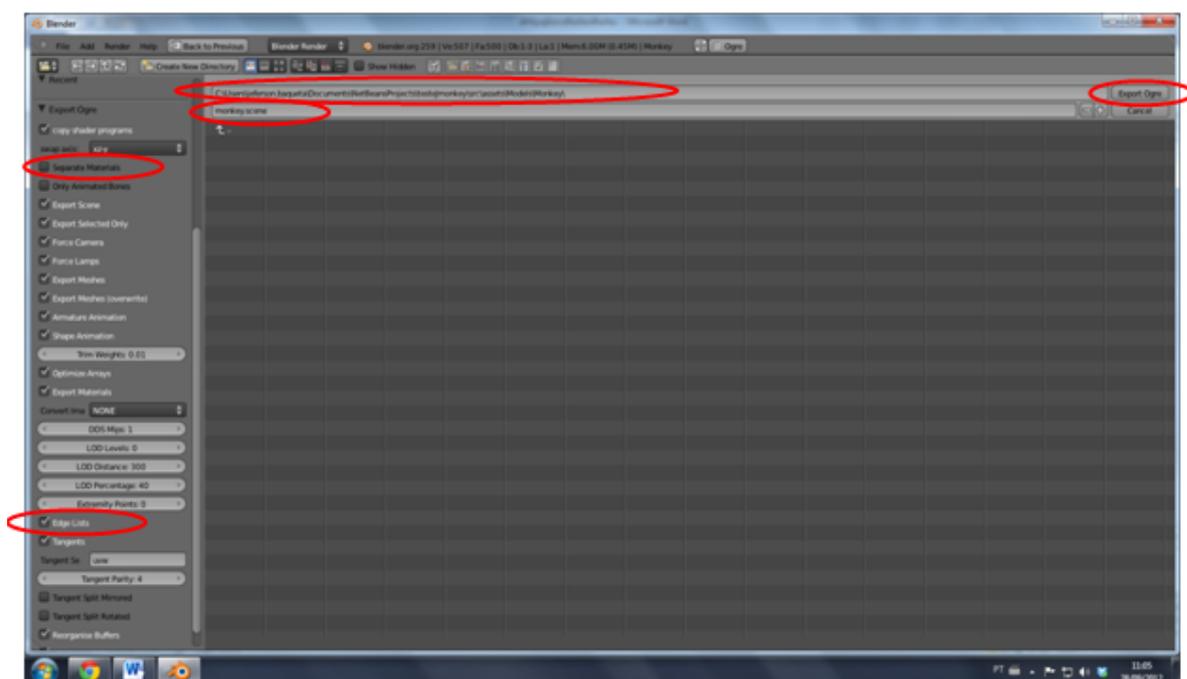


Figura A.24: Janela do Blender para definir configurações de exportações

No canto esquerdo da tela de exportação existem algumas configurações que devem ser verificadas para que exportação esteja de acordo com as especificidades da *engine* jMokey. Para tal, a opção “*Separate Materials*” deve ser desmarcada e a “*Edge List*” deve ser marcada (Figura A.24). Para não precisar refazer essa operação selecione o *menu File* e escolha a opção “*Save User Setting*”, assim as opções de exportações serão salvas, como na Figura A.25.

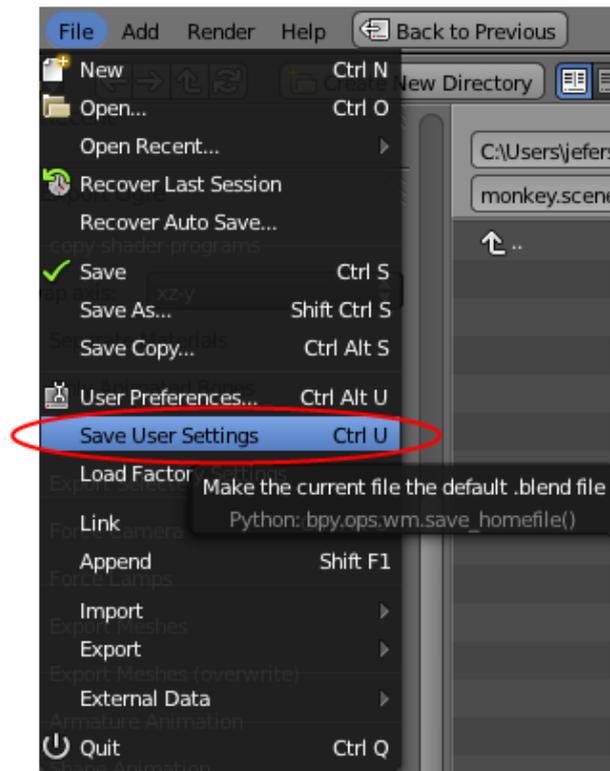


Figura A.25: Salvando as preferências do usuário

Após definida a opção de exportação é só clicar no botão “exportOgre”, como apresentado na Figura A.24, e o modelo será exportado para a pasta definida pelo usuário. Ao final do processo de exportação, no canto superior direito da janela base do Blender, 3DView, aparecerá uma janela temporária, Figura A.26, indicando algumas informações sobre o modelo que foi exportado por meio do Ogre3D, como o número de malhas, triângulos, vértices, câmeras, luz, entre outras propriedades tanto do modelo, como da cena onde esse foi inserido.

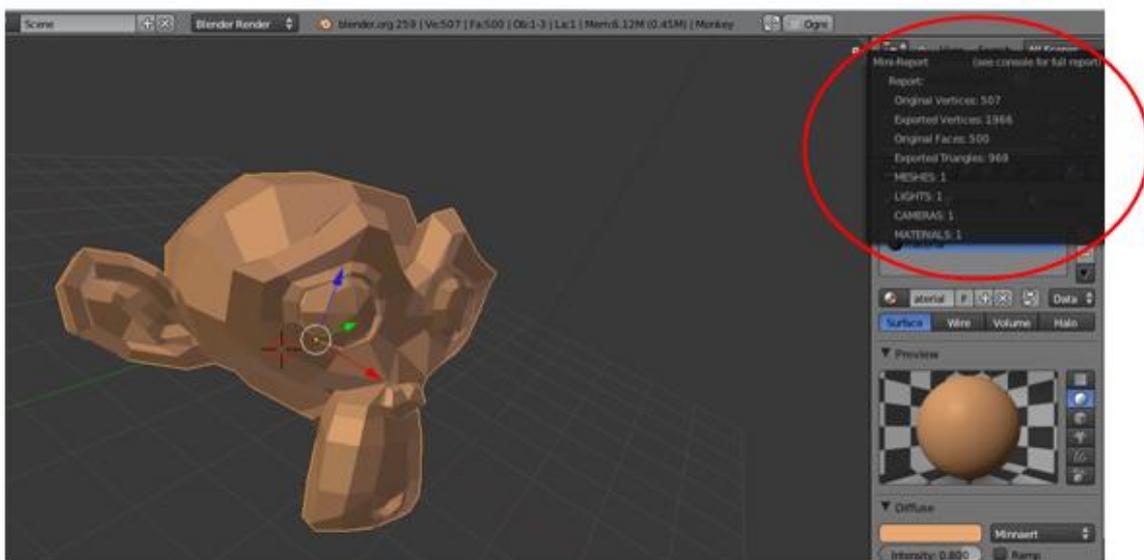


Figura A.26: 3DView e janela de confirmação de exportação

A.3.3 Como Carregar Modelos Exportados por Ogre3D

No diretório onde o modelo foi exportado aparecerão alguns arquivos, dentre os quais devem estar pelo menos um arquivo `.scene` com o nome do modelo e informações sobre o mesmo, um arquivo `.mesh` que guarda as configurações da malha e um arquivo `.material` que informa as propriedades do material e textura. Caso o modelo possua alguma textura vinculada (não é o caso do exemplo) no diretório também aparecerá um arquivo com formato de imagem, geralmente `.png` ou `.jpg`, que será o mapa UV.

Ainda em relação à exportação, caso o modelo exportado possua um esqueleto, estrutura utilizada para realizar animações, um arquivo `.skeleton`, contendo informações referentes a movimentação e peculiaridades dos ossos do modelo aparecerá no diretório.

No diretório onde o modelo foi exportado o arquivo `.material` deve possuir o mesmo nome que o arquivo `.mesh`, pois no caso de aplicação de texturas podem ocorrer alguns problemas por conta de referência de arquivos. Os arquivos gerados podem ser visualizados na Figura A.27.

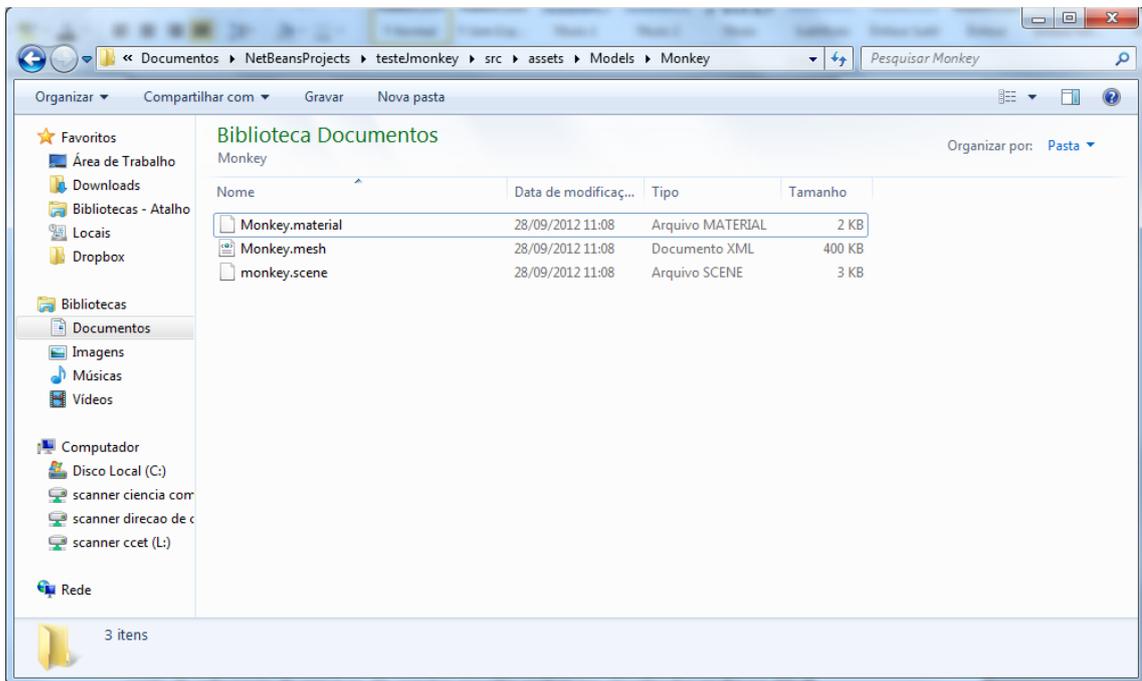


Figura A.27: Arquivos gerados pela exportação

Após a verificação dos arquivos deve-se carregar o modelo gerado em Blender para a aplicação jMonkey. Para realização dessa ação pode-se usar o projeto já criado no Netbeans para configuração do jMonkey e JMF juntamente com o fosb4.

Na classe PrimeiroJogo substitua as importações pelas seguintes:

```

7 | import com.jme3.app.SimpleApplication;
8 | import com.jme3.light.AmbientLight;
9 | import com.jme3.light.DirectionalLight;
10 | import com.jme3.math.Vector3f;
11 | import com.jme3.scene.Spatial;

```

No corpo do código deve-se realizar o carregamento do modelo 3D criado, adicionar uma luz ambiente para que o mesmo possa ser visualizado, e em seguida adicioná-lo à árvore de renderização para que esse seja exibido em tela, sendo assim, substitua o código-fonte do método simpleInitApp() pelo seguinte:

```

32  @Override
33  public void simpleInitApp() {
34      flyCam.setDragToRotate(true);
35      Spatial modelo = assetManager.loadModel("assets/Models/Monkey/monkey.scene");
36      modelo = modelo.rotate(0, -45, 0);
37      AmbientLight luz = new AmbientLight();
38      rootNode.addLight(luz);
39      DirectionalLight direcaoLuz = new DirectionalLight();
40      direcaoLuz.setDirection(new Vector3f(23, 44, 12).normalizeLocal());
41      rootNode.addLight(direcaoLuz);
42      rootNode.attachChild(modelo);
43  }

```

Na linha trinta e cinco, onde o modelo recebe `assetManager.loadModel`, o caminho do modelo criado deve ser especificado para que o `jMonkey` consiga carregá-lo. O resultado desse código pode ser visualizado na Figura A.28, onde é exibida a cabeça de macaco criada no Blender 2.5.9.

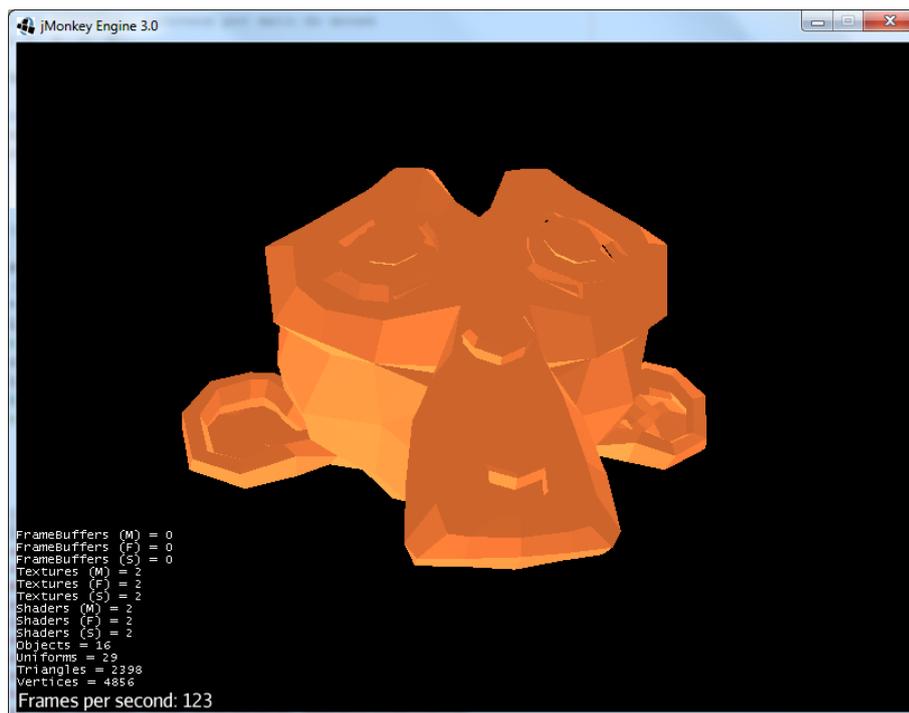


Figura A.28. Cabeça de macaco renderizada em `jMonkey`

Após da execução de todos os passos apresentados nesse anexo o ambiente estará pronto para programação, contudo vários detalhes devem ser pesquisados, principalmente os relacionados a conceituação, em especial alguns conhecimentos sobre computação gráfica são muito úteis para o desenvolvimento.

Recomenda-se que antes de dar início a programação do jogo seja realizado um estudo prévio sobre a *engine* jMokey, que possui uma vasta documentação disponibilizada na internet, além de um estudo sobre o modelador Blender, pois por meio deste é possível elaborar vários modelos que poderão ser exportados para o jogo, contudo, quanto mais detalhados os modelos maior a quantidade de processamento gráfico exigido e, conseqüentemente, mais recursos serão exigidos do computador para a execução do jogo.

Referências

ARCOVERDE, R. D. L. *Tecnologias Digitais: Novo Espaço Interativo na Produção Escrita dos Surdos*. Cadernos Cedes, São Paulo - SP, vol. 26, n. 69, p. 251-267, 2006.

BAQUETA, J. B.; BOSCARIOLI. C. *Uma Discussão Sobre o Papel das Tecnologias no Ensino Aprendizagem de Alunos Surdos*. In: Segundo Encontro Nacional de Informática e Educação (II ENINED). Paraná, 2010. p.113-141.

BAQUETA, J. B.; BOSCARIOLI. C.; BIDARRA, J. *Tecnologias Assistivas como Instrumento de Auxílio à Educação Inclusiva: Uma Discussão Inicial*. In: Segundo Simpósio Nacional de Educação. Paraná, 2010.

BARBOSA, S. D. J; SILVA. B. S, *Interação Humano-Computador*. 1. ed. Editora: Campus, 2010.

BARCELOS, R. J. S. *Ambiente Virtual de Aprendizagem 3D: proposta de Objeto de Aprendizagem para o Ensino de Algoritmos*. In: Segundo Encontro de Educação a Distância. Rio de Janeiro, 2008. p. 48-61.

BARROS, M. S. *Estudo de Usabilidade em Jogos Educativos 3d: Um Estudo de Caso*. Monografia - UFLA - Universidade Federal de Lavras, Lavras, 2011.

CARBONERO, F. C; G. M. ZAGO; CAMPOS. D. *Tecnologia Assistiva na Distrofia Muscular de Duchenne: Aplicabilidade e Benefícios*. 1. Ed. Neurocienc, São Paulo, v. 20, p. 109 - 116, 2012.

CLUA, E. W. G.; BITTENCOURT, J. R. *Desenvolvimento de Jogos 3D: Concepção, Design e Programação*. In: XXIV Jornada de Atualização em Informática (JAI). Rio Grande do Sul, 2005. p. 1313-1357.

CORREIA, L. M. In: *Inclusão e Educação Doze Olhares Sobre a Educação Inclusiva*. 1. ed. Editora: Summus. cap.: 10. *Dez Anos de Salamanca, Portugal e os Alunos com Necessidades Educativas Especiais*. 2006. p. 239 - 274.

- COUTO, R. M. S. *Processo de Projeto do Jogo Multi-Trilhas: Um exemplo de Design Participativo*. In: 8ª Congresso Brasileiro de Pesquisadores e Desenvolvimento em Design. São Paulo, 2008. p. 4107 - 4111.
- DAMÁZIO, M. F. M. *Atendimento Educacional especializado Pessoa com Surdez*. 1. ed. Editora: Cromos, 2007.
- DEFICIENTENET. Disponível em: <http://www.deficiente.net/forum/viewtopic.php?f=16&t=3008>. Último consulta em: 01/05/2011.
- DEITEL, P; DEITEL, H. *Java como programar*. 8. ed. Editora: Pearson Prentice Hall, 2010.
- FERREIRA, J. R. In: *Inclusão e Educação Doze Olhares Sobre a Educação Inclusiva*. 1. ed. Editora: Summus, 2006. Cap. 4. *Educação Especial, inclusão e políticas educacionais: notas brasileiras*. p. 85 - 114.
- FIGUEIREDO, A. M. O; LELES, F. J. F; SANTANA, L. L. S; JÚNIOR, N. A. S; CAZETTA, P. P; SANTOS, P. C; DUTRA, S. F. E. *Uma Proposta de Jogo Educacional 3D com Questões Didáticas*. In: 22º Simpósio Brasileiro de Informática na Educação e 17º Workshop de Informática na Escola. Sergipe, 2011.
- FILHO, T. A. G. *Tecnologia Assistiva para uma Escola Inclusiva: Apropriação, Demandas e Perspectivas*. Tese de Doutorado - Universidade Federal da Bahia, Bahia, 2009.
- GESSER, A. *Libras? Que Língua é Essa? Crenças e Preconceitos em torno da Língua de Sinais e da Realidade Surda*. 1. ed. Editora: Parábola Editorial, 2009.
- GIL, M. *Educação Inclusiva: O que o Professor tem a ver com Isso?*. 1. ed. Editora: Imprensa Oficial, 2005.
- HACK, J. R.; NEGRI, F. *Capacitação Docente para o Uso da Midia como Ferramenta Didática: Um Espaço de Reflexão e Ação*. In: 14º Congresso Internacional ABED de Educação a Distância. São Paulo, 2008. p. 1 - 8.
- JÚNIOR, F. K. *Formação Continuada para Instrutores de Informática Educacional da Rede Pública de Ensino Municipal. Tecnologias para Surdos*. Curso Técnico – CAS, Cascavel, PR, 2010.
- JMONKEYENGINE. Disponível em: <http://jMonkeyengine.com>. Último consulta em: 22/02/2012.

KUBASKI, C; MORAES, V. P. *O bilinguismo como proposta educacional para crianças surdas*. In: IX Congresso Nacional de Educação (EDUCERE) e III Encontro Sul Brasileiro de Psicopedagogia. Paraná, 2009. p. 3413 - 3419.

LEITÃO, U. A; SIMÃO, T. D; NEVES, J. A. *Desenvolvimento de Jogos 3d para a Educação a Distância*. In: VIII Congresso Brasileiro de Ensino Superior a Distância (ESUD). Ouro Preto, 2011. p. 1 - 13.

LENCASTRE, J. A; CHAVES, J. H. *Ensinar pela Imagem*. Revista Galego-Portuguesa de Psicoloxía e Educación, Coruña, v. 10, n.8, p. 2100-2105, julho, 2003.

LIBRASNET. Disponível em: <http://www.librasnet.com>. Último acesso em: 02/07/2012.

LIMA, N. M. *Legislação Federal Básica na área da pessoa portadora de Deficiência*. Brasília: Secretaria Especial dos Direitos Humanos, Coordenadoria Nacional para Integração da Pessoa Portadora de Deficiência, 2007. Disponível em http://portal.mj.gov.br/corde/arquivos/pdf/Legislação_F_Basica.pdf. Último acesso: 02/01/2012.

MANTOAN, M. T. E. In: *Inclusão e Educação Doze Olhares Sobre a Educação Inclusiva*. 1. ed. Editora: Summus, 2006. cap: 8. *O direito de ser, sendo diferente, na escola*. p. 183-210.

MEDEIROS, N. L; GIANINI, E; GOMES, M. J; BATISTA, W. B. *Desenvolvimento de brinquedos pedagógicos para crianças surdas*. In: Anais do III Congresso Internacional de Pesquisa em Designer. Rio de Janeiro, 2005.

NETO, M. A. C; LEITE, J. C. *Empregando modelo de interação na representação de padrões de usabilidade*. In: VII Simpósio Brasileiro de Sistemas de Informação. Bahia, 2011. p. 286 - 297.

OHIRA, L. M. *Identificação de Requisitos para Usabilidade de Software Assistivo*. Dissertação de Mestrado - UFPR - Universidade Federal do Paraná, Paraná, 2009.

PERLIN, G; STROBEL, K. *Fundamentos da Educação de Surdos*. Monografia - UFSC - Universidade Federal de Santa Catarina, Santa Catarina, 2006.

PRATES, R.O.; SOUZA, C.S.; BARBOSA, S.D.J. *A Method for Evaluating the Communicability of User Interfaces*. Interactions, Nova York, v.7, n.1, p.31-38, 2000.

QUADROS, R. M. Aquisição de L1 e L2: *O Contexto da Pessoa Surda*. In: Seminário de Desafios e Possibilidades na Educação Bilíngue para Surdos. Rio de Janeiro, 1997. p. 70 - 87.

RYBENÁ. Disponível em: <http://www.rybena.org.br/default/index.jsp>. Último acesso em: 26/06/2012.

SCHIRMER, C. R.; BROWNING, N. R.; BERSCH, R.; MACHADO, R. *Atendimento educacional Especializado Deficiência Física*. 2007. Disponível em: http://portal.mec.gov.br/seep/arquivos/pdf/eee_df.pdf. Último acesso em: 20/05/2012.

SQLITE. Disponível em: <http://www.sqlite.org>. Último acesso em: 24/11/2012.

SURDEZ. Disponível em: <http://www.surdez.org.br>. Último acesso em: 29/06/2012.

TAROUCO, L. M. R; ROLAND, L. C; FABRE, M. C. J. M; KONRATH, M. L. P. *Jogos Educacionais*. In: Anais do III Novas Tecnologias na Educação (CINTED). Porto Alegre, 2004, p. 1-8.

UNESCO. *Declaração de Salamanca e Linha de Ação sobre Necessidades Educativas Especiais*. Brasília: CORDE, 1994.

X3DOM. Disponível em: <http://www.x3dom.org>. Último acesso em 13/07/2012.