



**ESCOLA DE ARTES, CIÊNCIAS E HUMANIDADES
EACH- USP**



**Programa Institucional de Bolsas de Iniciação Científica em
Desenvolvimento Tecnológico e Inovação – PIBITI/CNPq/USP**

**Registro de dados de interação usando Programação Orientada a
Aspectos em um sistema para avaliação da aquisição de conhecimento
em Ambientes Virtuais de Aprendizagem tridimensionais**

Relatório Final

Aluno:

Rafael Luiz Testa

Orientadora:

Fátima de Lourdes dos Santos Nunes Marques

Agosto de 2013

Glossário

AVA - Ambiente Virtual de Aprendizagem

API - Interface de Programação de Aplicativos (*ou Application Programming Interface*)

BD - Banco de Dados

DER - Diagrama Entidade-Relacionamento

POA - Programação Orientada a Aspectos

POO - Programação Orientada a Objetos

RS - Revisão Sistemática

RV - Realidade Virtual

VIDA - *Virtual and Interactive Distance-Learning on Anatomy*

SGBD – Sistema Gerenciador de Banco de Dados.

Resumo

Com a popularização dos Ambientes Virtuais de Aprendizagem (AVAs) o processo de aquisição de conhecimento (cognição) dos estudantes tem sido alvo de pesquisas. Uma discussão que é delineada nesse contexto se refere à contribuição que os AVAs efetivamente oferecem para a aquisição de conhecimento do aprendiz e como é possível avaliar tal aprendizado. Estudos têm demonstrado que os métodos amplamente aplicados para avaliação da cognição nos AVAs têm feito uso intensivo de preenchimento de questionários, aplicados fora dos AVAs, favorecendo respostas subjetivas que podem não condizer com a realidade. O presente projeto está inserido em um escopo mais abrangente que visa a definir uma metodologia para avaliar a aquisição de conhecimento em AVAs tridimensionais. A fim de tornar o processo automático e útil, é necessário que tal metodologia não requeira alterações no AVA que está sendo avaliado. Uma das possíveis soluções é o emprego de Programação Orientada a Aspectos (POA). Assim, neste projeto foi empregado os conceitos de POA para implementar um módulo de registro de dados de interação em um sistema destinado a avaliar a aquisição de conhecimento em AVAs.

Palavras-chave: monitoramento das interações, avaliação da aprendizagem, Ambientes Virtuais de Aprendizagem Tridimensionais.

Lista de ilustrações

Figura 1 – Fluxograma das informações desde a identificação até a inclusão dos estudos	8
Figura 2 – JINSI - captura as chamadas de métodos de saída através da inserção de vários testes (BURGER e ZELLER, 2011)	10
Figura 3 – Funcionamento do processo de registro dos dados (BATEMAN <i>et al.</i> , 2009)	13
Figura 4 – Modelo Teórico de Avaliação da Aquisição de Conhecimento (NUNES <i>et al.</i> , 2012)	16
Figura 5 – Arquitetura do sistema (Nunes <i>et al.</i> , 2012)	17
Figura 6. DER do banco de dados do Sistema de Avaliação da Aprendizagem	19
Figura 7. Exemplo de Aspecto.....	21
Figura 8. Projeto VIDA (fases do feto durante a gestação)	23
Figura 9. Sistema CONSTRUFIG 3D (versão 2.0)	24
Figura 10. Módulo de Monitoramento das Interações dos Usuários.....	25
Figura 11. Ponto de corte do método principal do AVA 3D.....	27
Figura 12. Ponto de corte de controle da visibilidade do AVA 3D.....	27
Figura 13. Invocação do método principal que exibe o AVA 3D.....	28
Figura 14. Sistema de Avaliação da Aprendizagem (perfil avaliador) – cadastro nova interação.....	29
Figura 15(a) – Movimento realizado no AVA 3D VIDA.....	30
Figura 15(b) – Registro da interação realizada no AVA 3D.....	30
Figura 16. Sistema de Avaliação da Aprendizagem (perfil avaliador) – gerenciamento das interações.....	30
Figura 17. Sistema de Avaliação da Aprendizagem (perfil avaliador) – cadastro nova interação no CONSTRUFIG 3D.....	31
Figura 18(a) – Evento botão selecionado na interface.....	31
Figura 18(b) – Registro da interação realizada no AVA 3D.....	31
Figura 19. Sistema de Avaliação da Aprendizagem – tela de autenticação.....	34
Figura 20. Sistema de Avaliação da Aprendizagem – tela de administrador	34
Figura 21(a) – Tela de autenticação no sistema (avaliador)	35
Figura 21(b) – Tela inicial (perfil avaliador).....	35

Figura 22. Tela menu participante (perfil avaliador).....	35
Figura 23(a). Tela adicionar ou editar turma (perfil avaliador).....	36
Figura 23(b). Tela adicionar participantes na turma (perfil avaliador).....	36
Figura 24. Tela menu treinamento (perfil avaliador).....	37
Figura 25. Tela gerar avaliação (perfil avaliador).....	37
Figura 26. Tela relatório das turmas cadastradas (perfil avaliador).....	38
Figura 27. Tela detalhes da turma (perfil avaliador).....	38
Figura 28. Cadastro do participante (perfil participante).....	39
Figura 29. Interface do sistema (perfil participante).....	39

SUMÁRIO

1	Introdução	3
1.1	Objetivos.....	4
1.2	Justificativa	5
1.3	Organização do relatório.....	6
2	Estado da Arte.....	6
2.1	Condução da Revisão Sistemática	8
2.2	Resultados e Discussões	9
2.2.1	Métodos/Técnicas de Monitoramento das Interações (invasivas ao código da aplicação)	11
2.2.2	Métodos/Técnicas de Monitoramento das Interações (não invasivas ao código da aplicação).....	12
2.3	Considerações Finais sobre a Revisão Sistemática	15
3	Metodologia	16
3.1	Tecnologias Utilizadas.....	19
3.2	Programação Orientada a Aspectos (POA)	21
3.3	Estudo de Caso	23
4	Resultados	25
4.1	Módulo de Monitoramento das Interações do Usuário.....	26
4.2	Testes de Validação do Módulo de Monitoramento das Interações	30
4.3	Funcionamento do Sistema de Avaliação da Aprendizagem	33
	Considerações Finais	41

Referências	42
-------------------	----

1 Introdução

O uso do computador e das tecnologias da informação e comunicação tem crescido em diversas áreas do conhecimento. No âmbito educacional o desenvolvimento de ferramentas de apoio ao ensino, como simuladores e jogos sérios por meio de Ambientes Virtuais de Aprendizagem (AVAs) está se tornando uma alternativa cada vez mais comum. Essas ferramentas buscam motivar e contribuir com a aquisição de conhecimento, principalmente em relação a conteúdos difíceis de serem absorvidos.

Com o advento da Web 2.0, os AVAs evoluíram com os recursos proporcionados pelas tecnologias de Realidade Virtual (RV) e Realidade Aumentada (RA). Surgiram, assim, os AVAs tridimensionais (3D), que estão sendo amplamente adotados em diversos domínios de conhecimento, destacando-se a área de educação médica (TORO-TROCONIS *et al.*, 2010; SOLER *et al.*, 2008; COULTER *et al.*, 2007).

Um exemplo de ambiente virtual 3D que tem sido empregado como AVA é o *Second Life* que, apesar de ter sido desenvolvido inicialmente para relacionar as diversas pessoas do mundo usando uma interface intuitiva, evoluiu e passou a ser amplamente adotado no meio acadêmico, profissional e social. Este ambiente tem sido empregado, por exemplo, em experiências ligadas à Biologia (como genética) e Medicina, aproveitando o poder de simulação da ferramenta (VALENTE e MATTAR, 2007).

Considerando as três principais características proporcionadas pelos sistemas de Realidade Virtual e Realidade Aumentada (imersão, interação e envolvimento), acredita-se que a evolução no processo de ensino-aprendizagem por meio dos AVAs 3D está relacionada, principalmente, à intensidade das interações dos participantes no ambiente, pois a interação leva a uma sensação de imersão e, conseqüentemente, maior envolvimento do aprendiz. Nesse contexto, registrar os dados das interações dos usuários durante a exploração no AVA 3D pode resultar em informações relevantes para avaliar a contribuição de tais ambientes para a evolução do aprendizado.

Encontram-se na literatura muitas pesquisas relacionadas ao monitoramento das interações dos usuários em sistemas computacionais, uma vez que são gerados dados que podem auxiliar em diversas áreas da computação, como também em todas as áreas que fazem uso de sistemas computacionais, incluindo os AVAs 3D.

Dessa forma, o presente projeto está inserido em um projeto maior, que visa a definir uma metodologia para avaliar a aquisição de conhecimento em AVAs tridimensionais. Pretende-se que esta metodologia seja automatizada, ou seja, os dados sobre a interação do usuário com o AVA devem ser registrados durante a execução no ambiente.

Porém, a fim de tornar o processo automático e útil, é necessário que tal metodologia não requeira alterações no código fonte do AVA sob avaliação. Uma das possíveis soluções é o emprego de Programação Orientada a Aspectos (POA). A POA tem como objetivo implementar separadamente os níveis de interesses-base, que referem-se às funcionalidades principais do sistema e os interesses transversais, que se referem às restrições globais e requisitos não funcionais, como persistência de dados, distribuição, autenticação, controle de acesso e concorrência, entre outros. A proposta é desenvolver as partes do sistema sem se preocupar com as demais partes (RESENDE e SILVA, 2005).

Assim, visto que este projeto visa empregar os conceitos de POA para programar um módulo de registro de dados de interação em um sistema destinado a avaliar a aquisição de conhecimento em AVAs 3D, a seção 2 apresenta o Estado da Arte sobre o monitoramento das interações dos usuários em sistemas computacionais, realizado por meio de uma Revisão Sistemática (RS) aprofundada na literatura, buscando identificar métodos e técnicas que implementam o registro das interações dos usuários. Dessa forma, é possível verificar se o uso de POA é o método/técnica de programação mais adequado ao propósito da pesquisa.

1.1 Objetivos

Este trabalho tem como objetivo principal desenvolver um módulo de monitoramento das interações do usuário em um sistema destinado a avaliar a aquisição do conhecimento em AVAs 3D, utilizando conceitos de Programação Orientada a Aspectos. Este módulo permitirá o registro das interações do usuário em um AVA 3D de forma não intrusiva, ou seja, sem alterar o código da aplicação.

Para tal, são definidos os objetivos específicos:

- pesquisar conceitos de Programação Orientada a Aspectos;
- implementar um banco de dados para armazenar adequadamente os dados obtidos a partir da interação do usuário com o AVA 3D, considerando a necessidade de flexibilidade em diversos contextos;

- definir e implementar um banco de dados para cadastro de usuário, considerando diferentes níveis de acesso, como também armazenamento, consulta e alteração de perfil.

1.2 Justificativa

A exploração de Ambientes Virtuais de Aprendizagem Tridimensionais a partir de simulações de situações reais vem sendo uma prática cada vez mais comum em diversos campos de conhecimento, pois ampliam as possibilidades de estudo e prática de conceitos e técnicas, além de oferecer ao aprendiz a possibilidade de manipular e analisar diretamente o alvo de estudo (NUNES *et al.*, 2012).

Por meio dos AVAs 3D, o processo de ensino-aprendizagem tem potencial para tornar-se mais ativo, dinâmico e personalizado. Porém, uma discussão que é delineada neste contexto se refere à contribuição que tais ambientes efetivamente oferecem para a aquisição de conhecimento do aprendiz. (NUNES *et al.*, 2010; NUNES *et al.*, 2012).

Uma das possíveis formas de verificar a contribuição dos AVAs 3D no processo de ensino-aprendizagem é medir o nível de interações do aprendiz durante a exploração no ambiente virtual e verificar se a intensidade das interações no ambiente tem relação com a evolução do aprendizado. Portanto, obter o registro das interações dos usuários durante a exploração no AVA 3D, em conjunto com outros parâmetros de avaliação, é uma forma de averiguar se tais ambientes estão contribuindo para a evolução da aprendizagem.

Monitorar as interações dos usuários em sistemas computacionais é um campo de pesquisa que tem sido investigado pela comunidade científica em diversos contextos, considerando desde o auxílio na tomada de decisão em tempo real até a análise do comportamento dos usuários para algum propósito específico.

Salienta-se ainda que de acordo com Dolunay e Akgunduz (2008) registrar as interações dos usuários em sistemas de RV e RA é um tema de investigação que não foi abordado suficientemente na literatura. Portanto, o presente projeto torna-se uma contribuição para as áreas de RV, RA, AVAs, além de áreas correlatas.

1.3 Organização do relatório

Este relatório possui além desta Introdução, quatro seções, a saber:

Seção 2 – Estado da Arte, que apresenta a revisão bibliográfica sobre os métodos e técnicas utilizados para implementar o monitoramento das interações dos usuários em sistemas computacionais.

Seção 3 – Metodologia; esta seção apresenta os materiais e métodos utilizados na pesquisa, como tecnologias, métodos e técnicas de programação.

Seção 4 – Resultados, nesta seção são mostrados os resultados obtidos com a realização do projeto, em relação ao processo de implementação do registro das interações dos usuários nos AVAs 3D e do Sistema de Avaliação da Aprendizagem em AVAs 3D, o qual o módulo de registro de interações está relacionado.

Por fim, têm-se as Considerações Finais e Referências utilizadas nesta pesquisa.

2 Estado da Arte

A metodologia de investigação empregada no levantamento do estado da arte foi o processo de Revisão Sistemática (RS). A RS é uma metodologia de pesquisa que objetiva apresentar uma avaliação da pesquisa disponível acerca de um determinado tema (KITCHENHAM, 2004) e, diferentemente das revisões bibliográficas exploratórias, busca ser mais rigorosa, utilizando uma metodologia de pesquisa empírica e passível de generalização. Deste modo, o pesquisador pode fazer uma análise crítica dos dados coletados, resolvendo conflitos encontrados na literatura e identificando as possíveis lacunas do tema (BIOLCHINI *et al.*, 2005).

Uma RS em geral é dividida em três fases (Planejamento, Condução e Análise/Interpretação de Resultados). Na fase de Planejamento são identificadas as necessidades da revisão e desenvolve-se um protocolo que é a base para a próxima fase, que trata da Condução da revisão. Na fase de Condução se faz a identificação dos estudos, a seleção dos estudos primários, a extração e a síntese dos dados. Por fim, na fase de Análise/Interpretação de Resultados são formuladas conclusões sobre o estado da arte do problema em questão.

A RS apresentada nesta seção tem como objetivo principal identificar estudos relacionados a métodos e técnicas que estão sendo utilizados para monitorar e/ou capturar as interações dos usuários em sistemas computacionais.

Para conduzir esta revisão de literatura, primeiramente foi estabelecido o protocolo de RS elaborado com base nos modelos propostos por (KITCHENHAM, 2004; BIOLCHINI *et al.*, 2005). Seus principais tópicos são:

Objetivo: Identificar métodos e técnicas existentes para monitorar as interações dos usuários em sistemas computacionais.

Questão de pesquisa: Quais são os métodos e técnicas utilizados para monitorar e/ou capturar as interações dos usuários em sistemas computacionais?

Recursos para Busca e Seleção de Estudos – os recursos e estratégias para busca e seleção de estudos preliminares foram definidos e selecionados com base em três itens fundamentais:

Identificação de fontes de busca: considerando que os estudos primários devem estar disponíveis na *web*, as fontes de busca identificadas foram as bases de dados eletrônicas indexadas (IEEE¹, ACM² e Scopus³).

Palavras-chave: “monitoring”, “interaction capturing”, “interaction”, “usability”, “usage”, “sensor”, “network”, “mobile”, “system”, “software”, “application”, “user monitoring”, “user interaction”, “monitoring user”, “user action”, “user activity”, “user behavior”. A escolha das palavras-chave e suas relações se justificam pelo objetivo principal da RS.

Data dos trabalhos: foram considerados trabalhos a partir de 2008, pois se buscou novas abordagens de monitoramento das interações.

CrITÉRIOS de inclusão: (I) Abordam sobre monitoramento das interações do usuário.

CrITÉRIOS de exclusão: (E1) Não abordam sobre o monitoramento das interações; (E2) Abordam sobre sistemas para dispositivos móveis; (E3) Abordam sobre sistemas de monitoramento em redes de computadores; (E4) Abordam sobre sistemas de sensores; (E5) Não abordam sobre sistemas computacionais.

¹ <http://ieeexplore.ieee.org/Xplore/>

² <http://dl.acm.org/>

³ <http://www.scopus.com/home.url>

Seleção Preliminar – para esta seleção o pesquisador utilizou as *strings* de busca formadas para cada uma das bases de dados selecionadas. Os trabalhos recuperados das bases foram inicialmente armazenados, e depois, documentados na fase de condução da revisão e selecionados com base nos critérios de inclusão e exclusão previamente estabelecidos no protocolo. Vale ressaltar que essa seleção foi executada mediante a leitura dos títulos dos artigos e seus *abstracts*. Obras repetidas foram documentadas uma única vez.

Seleção Final e Extração de Resultados – o processo de seleção final consistiu na leitura completa dos artigos incluídos. Os resultados foram sumarizados e serão apresentados na seção de Resultados.

2.1 Condução da Revisão Sistemática

Na fase de Condução da RS foi aplicado o processo de seleção preliminar, definindo-se os trabalhos a serem incluídos e excluídos da RS, de acordo com os critérios de inclusão e exclusão estabelecidos no protocolo e documentados nos formulários de condução da RS.

Considerando as especificidades de cada base de dados em relação aos mecanismos de busca, foram utilizadas combinações diferentes de palavras-chave, com exceção da string “*interaction capturing*” que foi aplicada nas três bases de dados. As *strings* de busca aplicadas em cada base de dados encontram-se listadas a seguir.

ACM

1. (*monitoring AND "user interaction"*) AND (*"user action*" OR "user activity" OR usage OR "user behaviour" OR usability*) AND (*NOT mobile AND NOT sensor* AND NOT network*)
2. (*"user interaction"*) AND (*"monitoring user" OR "user monitoring"*) AND (*NOT mobile AND NOT sensor* AND NOT network*)
3. (*capturing AND "user interaction"*) AND (*"user action*" OR "user activity" OR usage OR "user behavior" OR usability*) AND (*NOT mobile AND NOT sensor* AND NOT network*)
4. “*interaction capturing*”

IEEE

1. (*monitoring AND "user* interaction"*) AND (*software OR system OR application*) AND *NOT mobile AND NOT sensor* AND NOT network*
2. (*interaction*) AND (*"monitoring user" OR "user monitoring"*)
3. “*interaction capturing*”

Scopus

1. (*monitoring AND "user interaction"*) AND (*"user action*" OR "user activity" OR usage OR "user behavior" OR usability*) AND (*NOT mobile AND NOT sensor* AND NOT network*)
2. (*capturing AND "user interaction"*) AND (*"user action*" OR "user activity" OR usage OR "user behavior" OR usability*) AND (*NOT mobile AND NOT sensor* AND NOT network*)
3. *"interaction capturing"*

A Figura 1 apresenta um fluxograma das informações advindas das diferentes fases da Revisão Sistemática, baseado em Moher *et al.* (2009). Este fluxograma tem como propósito apresentar de forma quantitativa o processo da RS, desde a fase de identificação dos estudos por meio das buscas nas bases de dados até a seleção final dos artigos incluídos na análise dos dados.

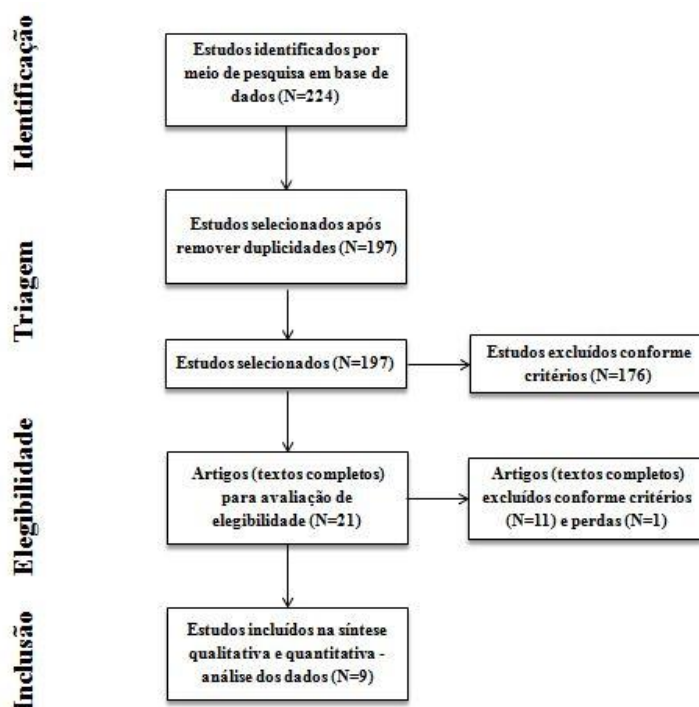


Figura 1. Fluxograma das informações desde a identificação até a inclusão dos estudos

2.2 Resultados e Discussões

Após a fase de seleção preliminar da RS, na qual resultou em 21 trabalhos selecionados para avaliação da elegibilidade, conforme ilustrou a *Figura 1*, passou-se a fase de seleção final e extração de resultados, que consistiu na leitura completa dos artigos. Ao concluir a fase de seleção final percebeu-se que dos trabalhos incluídos na seleção preliminar,

apenas 45% deles apresentavam algum tipo de técnica ou método para o monitoramento das interações dos usuários, enquanto os outros 55% se dividiam em trabalhos diversos, que apesar de abordarem o monitoramento das interações, não apresentaram explicitamente uma técnica ou método.

Deste modo, considerando que o objetivo da RS visa identificar, de forma aprofundada, métodos e técnicas para o monitoramento das interações, foram excluídos durante a fase de extração dos resultados 10 artigos que não os contemplavam. Além disso, houve perda de 1 artigo selecionado na fase de seleção preliminar por não estar disponível na web o texto completo, resultando-se ao final em 9 artigos incluídos na análise dos dados da RS.

A Tabela 1 apresenta os métodos e/ou técnicas de monitoramento das interações dos usuários identificados nos trabalhos incluídos na RS. São também apresentadas respostas a dois questionamentos considerados relevantes para cada método e/ou técnica: i) *Se o Método/Técnica pode ser usado em mais de uma aplicação?* e ii) *Se o Método/Técnica é intrusivo ao código da aplicação?*

Tabela 1. Artigos incluídos na RS durante a fase de seleção final - apresentam métodos/técnicas de monitoramento das interações

ID	Referência	Abordagem dos Métodos/Técnicas de Monitoramento	Método/Técnica pode ser usado em mais de uma aplicação?	Método/Técnica invasiva ao código da aplicação?
1	(Nakamura e Igarashi, 2008)	<i>Proxy</i> entre o gerenciamento de eventos Java e a aplicação	Sim	Não
2	(Ramsay <i>et al.</i> , 2008)	Modificação da API Java para interceptar os eventos	Sim	Não
3	(Bateman <i>et al.</i> , 2009)	Programação Orientada a Aspectos (POA)	Sim	Não
4	(Herbold <i>et al.</i> , 2011)	<i>Hook</i> de mensagens – usa API do Windows para interceptação de mensagens	Sim	Não
5	(Burger e Zeller, 2011)	<i>Proxy</i> de métodos	Sim	Sim
6	(Moore <i>et al.</i> , 2011)	Rastreamento ocular e vídeo capturado da tela	Sim	Não

7,8	(Vasconcelos e Baldochi, 2012; Azzopardi <i>et al.</i> , 2012)	Acoplamento de um código <i>Javascript</i> à página <i>web</i>	Sim	Sim
9	(Scholze <i>et al.</i> , 2012)	Implementa uma arquitetura de monitoramento Orientada a Serviços	Sim	Sim

As subseções 2.2.1 e 2.2.2 apresentam os trabalhos incluídos na RS e relacionados na Tabela 1.

2.2.1 Métodos/Técnicas de Monitoramento das Interações (invasivas ao código da aplicação)

Esta subseção apresenta os trabalhos que utilizaram um método e/ou técnica para monitorar as interações dos usuários, porém de forma invasiva ao código da aplicação. Apresenta ainda, uma breve explicação do método/técnica, bem como suas principais características, vantagens e limitações.

Em Burger e Zeller (2011) foi desenvolvido o sistema JINSI na área de Teste de Software, com o propósito de gravar as interações dos usuários em sistemas computacionais que estão em teste, buscando reproduzir as falhas do sistema para minimizar as situações que poderiam levar a ocorrência de erros. Para monitorar as interações, foi agregado ao JINSI o *Proxy* de métodos, que intercepta o sistema em teste e captura as interações dos usuários refazendo o fluxo de execução de todos os métodos observados, de forma a envolvê-los.

A Figura 2 mostra um exemplo do uso de *Proxy* de métodos, o qual substitui todos os métodos observados no código original do sistema em teste, para reproduzir um momento de falha quando detectado.

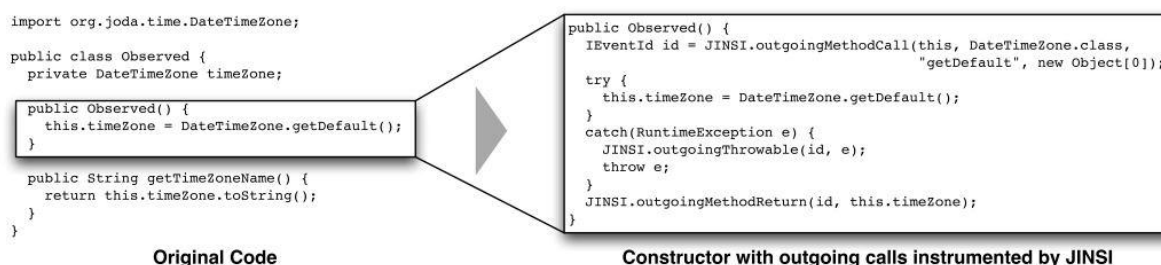


Figura 2. JINSI - captura as chamadas de métodos de saída através da inserção de vários testes (BURGER e ZELLER, 2011)

Dessa maneira é possível obter as interações do usuário utilizando os dados da invocação dos métodos, tendo em vista que são registrados os dados da chamada, dos atributos e também do retorno do método. A abordagem é aplicável apenas em sistemas de código aberto implementados na linguagem de programação Java.

Para avaliar a usabilidade de aplicações *Web*, Vasconcelos e Baldochi (2012) desenvolveram uma ferramenta para capturar as interações do usuário. A ferramenta automatiza o processo dos dados com o intuito de detectar problemas de usabilidade em tarefas pré-definidas. O acoplamento de um código *Javascript* à página *Web* é o meio utilizado para extrair tais informações. A aplicação captura os eventos relacionados às interações com os componentes da interface da aplicação, além de capturar quando se interage com os atributos *CSS* (*Cascading Style Sheet*), o que permite identificar com maior precisão as interações no ambiente.

Scholze *et al.* (2012) propõem uma solução para aprimorar as informações disponibilizadas em um Ambiente Virtual Colaborativo, agregando mais conhecimento ao ambiente ao relacionar as interações do usuário com o feedback implícito. Assim é possível agregar mais conhecimento ao sistema de colaboração sem pedir explicitamente ao usuário o seu conhecimento sobre o assunto em questão. Para monitorar as interações do usuário foi desenvolvida uma arquitetura de monitoramento Orientada a Serviços.

Dessa forma, tem-se uma estrutura com boa modularidade, fácil adaptação do serviço de monitoramento das interações e genérica a sistemas legados. A cada aplicação que necessitar armazenar os dados da interação do usuário, será adicionado o módulo de serviço. Porém, esta técnica exige sistema de código aberto para acrescentar o módulo.

2.2.2 Métodos/Técnicas de Monitoramento das Interações (não invasivas ao código da aplicação)

Nesta seção são apresentados os trabalhos que utilizaram um método e/ou técnica para monitorar as interações dos usuários, porém de forma não invasiva ao código da aplicação. Apresenta ainda, uma breve explicação do método/técnica, bem como suas principais características, vantagens e limitações.

Herbold *et al.* (2011) apresentam uma ferramenta genérica e não invasiva, capaz de armazenar as interações do usuário durante os testes de software, com propósito de recriar

as circunstâncias ocorridas no momento de falha do sistema e, assim, reproduzi-las automaticamente para perceber o erro e corrigi-la. O mecanismo utilizado na construção dessa ferramenta foi o gancho de mensagem⁴, fornecido por meio de uma API do sistema operacional Windows, capaz de interceptar as chamadas de métodos, mensagens ou eventos de uma aplicação. Ao interceptá-los, adiciona-se uma sub-rotina do registro das interações na fila de processamento de mensagens da aplicação monitorada.

Um método para visualizar o histórico das operações realizadas pelo usuário por meio da interação com a interface gráfica é apresentado por Nakamura e Igarashi (2008). Para tal, implementaram um *Proxy* entre o gerenciamento de eventos da linguagem Java e a aplicação. O método faz uso da estrutura da linguagem Java, na qual os eventos gerados a partir das interações do usuário com a *GUI (Graphical User Interface)* são gerenciados por uma fila de eventos (*EventQueue*) antes de serem enviados para a aplicação. Então, a fim de gravá-los, coloca-se um *Proxy* entre o gerenciamento de eventos da linguagem Java e a aplicação, capaz de interceptar esses eventos, sem modificar o código da aplicação.

Por meio do *Proxy*, o sistema de monitoramento grava as sequências de operações executadas. Este mesmo sistema captura uma imagem da tela com a configuração atual da interface gráfica paralelamente à sequência de eventos gravados, e relaciona-os por meio de uma marca de tempo e pela estrutura hierárquica dos componentes pertencentes à *GUI* (NAKAMURA e IGARASHI, 2008). Esta técnica, por aproveitar a estrutura de gerenciamento de eventos da linguagem Java, mostra como limitações a dependência de se utilizar apenas ferramentas de monitoramento desenvolvidas na linguagem de programação Java.

Uma ferramenta com a finalidade de comparar automaticamente os dados de interação dos usuários com modelos arquitetônicos de interfaces, já existentes para o mesmo propósito, é apresentada por Ramsay *et al.* (2008). Portanto, coleta informações relevantes da arquitetura da interface para auxiliar os designers, buscando desenvolver interfaces mais centradas no usuário. A ferramenta que desenvolve essa tarefa se chama *Inquisitor* e tem como uma das funções principais atuar como um mecanismo de extração de eventos. Baseado em *jrapture* de Steven *et al.* (2000), uma ferramenta com o mesmo objetivo, a tarefa de extração é executada fazendo modificações na API da linguagem Java.

⁴ <http://msdn.microsoft.com/en-us/library/ms632589.aspx>

A ferramenta *Inquisitor* atua também como um *wrapper* (empacotador) da classe principal e se registra na classe *Java's AWT Toolkit* para receber os eventos AWT gerados durante a execução (RAMSAY *et al.*, 2008). Esta abordagem é vulnerável às mudanças que ocorrem na API original. Portanto, se alguma classe for adicionada ou modificada, a ferramenta que extrai os eventos não mais funcionará (HERBOLD *et al.*, 2011). Porém, como todos os demais sistemas voltados para a linguagem *Java*, a ferramenta *Inquisitor* está limitada apenas às aplicações geradas pela linguagem *Java*.

Uma abordagem interativa para instrumentação dos dados de uso de *softwares* é proposta por Bateman *et al.* (2009). O registro das interações do usuário com o sistema é gerado de forma personalizada sem a necessidade de programação. O objetivo da proposta é alcançado usando-se Programação Orientada a Aspectos (POA). O paradigma da Programação Orientada a Aspectos permite aumentar a modularidade de diferentes funções pela separação dos interesses entrecortantes (BATEMAN *et al.*, 2009). Assim pode-se criar um módulo de interesse independente para instrumentar as interações do usuário, sendo este módulo o responsável por capturar os eventos.

A Figura 3 representa o funcionamento do processo de registro dos dados. Na qual os dados da aplicação instrumentada (monitora os dados de usabilidade) são capturados e armazenados em um arquivo de registro com as informações sobre a usabilidade dos dados do sistema, para posterior análise dos dados coletados (BATEMAN *et al.*, 2009).

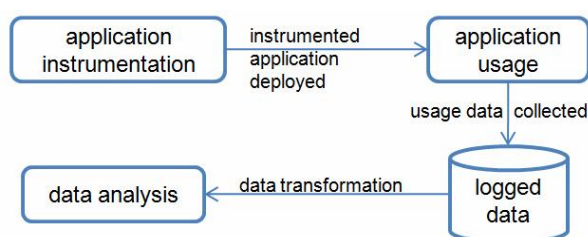


Figura 3. Funcionamento do processo de registro dos dados (BATEMAN *et al.*, 2009)

Esse mecanismo conta com um recurso que permite adicionar os elementos a serem monitorados apenas selecionando-os na interface, excluindo, portanto, a necessidade de programação para adicioná-los. Entretanto, a principal desvantagem da POA é apresentar um conjunto restrito de linguagens de programação com bibliotecas consolidadas para aspectos.

Outro trabalho no contexto de coleta de dados das interações é apresentado por Azzopardi *et al.* (2012), com propósito de compreender como o usuário interage com o um sistema Web e, assim, otimizar as buscas na Web. A técnica de acoplamento de uma biblioteca *Javascript* é aplicada nesse trabalho. Contudo, não é discutida a interação com os elementos CSS, mas os dados coletados são enviados para o servidor para serem registrados ou modelados. Uma desvantagem da técnica é o fato de ter que adicionar script a cada página Web que se deseja capturar as interações.

Um estudo apresentado por Moore *et al.* (2011) monitora as interações dos usuários em mecanismos de busca na Web usando a técnica de captura de vídeo da tela associada ao rastreamento ocular (*eye-tracking*). O rastreamento dos olhos preenche a lacuna entre os dados do vídeo e as entradas do usuário, compostas por sequências de teclas e acionamento do mouse. O processo de organização sequencial para reparar a questão de busca é comparado com o reparo via interações do usuário, identificando as similaridades e diferenças. Reparar a questão (*query*) de busca é um processo importante para tornar os resultados encontrados mais relevantes. Entretanto, esta abordagem necessita de um grande armazenamento e processamento de informações.

Considerando o exposto nas seções 2.2.1 e 2.2.2 existem diversos métodos/técnicas para realizar o monitoramento dos usuários em sistemas computacionais, inclusive em ambiente Web, sendo que a maioria utiliza a interceptação de eventos, fazendo a comunicação entre os diversos dispositivos usados pelos usuários (como *mouse* e teclado) e a interface gráfica do *software*, podendo assim capturar os eventos ocorridos a fim de obter as interações dos usuários durante a exploração no ambiente.

2.3 Considerações Finais sobre a Revisão Sistemática

A partir da RS realizada, foi possível identificar na literatura quais os principais métodos e técnicas que possibilitam monitorar os dados de interação do usuário em sistemas computacionais e armazená-los para posterior análise ou tomada de decisão em tempo real. Os métodos e técnicas encontrados permitiram uma melhor compreensão de como ocorre a interceptação das interações humanas em sistemas computacionais.

Verificou-se que 50% dos trabalhos selecionados na fase de seleção preliminar, não apresentaram uma técnica ou método para monitoramento das interações dos usuários,

apesar de abordarem o monitoramento das interações em sistemas computacionais. Nos demais trabalhos, que correspondem a 45% dos artigos selecionados na seleção preliminar, encontrou-se algum método e/ou técnica de monitoramento das interações. Destaca-se que na maioria deles, os recursos da linguagem de programação Java foram utilizados. Isto leva a induzir que seja pelo fato da linguagem Java oferecer técnicas de monitoramento das interações independente do sistema operacional, considerando que quanto maior a flexibilidade da ferramenta de monitoramento, maior será o número de aplicações passíveis de serem monitoradas.

É importante ressaltar ainda que mais de 50% dos trabalhos incluídos na fase de seleção final e extração de resultados da RS apresentaram técnicas de monitoramento que não são invasivas ao código fonte da aplicação, o que é essencial em um cenário no qual o código da aplicação não está disponível.

Verificou-se que existem diversas pesquisas que fazem uso do monitoramento das interações do usuário. Sendo assim, uma ferramenta de monitoramento das interações, não invasiva, adaptável a qualquer tipo de aplicação e independente de sistema operacional favorece a comunidade científica da área de AVAs, especialmente AVAs 3D. Porém, é um objetivo difícil de ser cumprido completamente, visto que as diferentes tecnologias que apoiam as linguagens de programação e os sistemas operacionais não oferecem suporte nativo para o problema em questão.

Salienta-se que foi encontrado na RS o trabalho de Bateman *et al.* (2009) que merece um destaque por tratar o problema de monitoramento das interações do usuário em sistemas computacionais com o uso da POA. E, portanto, se aproxima do objetivo deste estudo, o que pode vir a auxiliar na construção das classes de monitoramento das interações. Entretanto, está inserido em um contexto de Teste de Software, o que traz algumas particularidades da área.

3 Metodologia

Este trabalho faz parte de um contexto maior que vai além do registro dos dados das interações, pois está agregado à automatização do Modelo Teórico de Avaliação da Aquisição do Conhecimento proposto por Nunes *et al.* (2012) e ilustrado na Figura 4. Logo,

além do módulo do registro das interações dos usuários, também foi desenvolvido a automatização do modelo, conforme arquitetura do sistema apresentada na Figura 5.

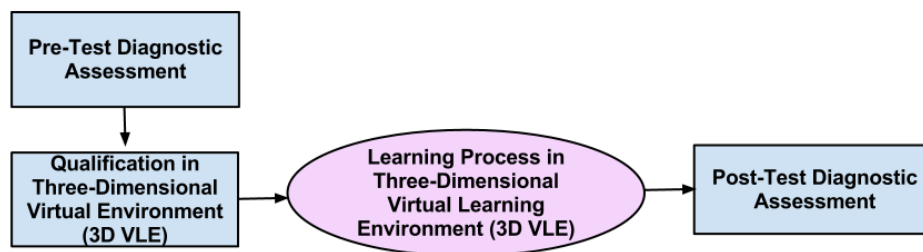


Figura 4. Modelo Teórico de Avaliação da Aquisição de Conhecimento (NUNES *et al.*, 2012)

Para compreender as partes do modelo de avaliação e em qual contexto o módulo de monitoramento das interações dos usuários se encontra inserido, segue uma breve descrição de cada módulo que compõe o Modelo Teórico de Avaliação da Aquisição de Conhecimento, ilustrado na Figura 4.

Avaliação Diagnóstica Pré-Teste – tem como objetivo avaliar o quanto o aprendiz conhece sobre o tema abordado, antes do mesmo passar pelo processo de aprendizagem por meio do AVA 3D, para depois comparar com os resultados obtidos na fase de Avaliação Diagnóstica Pós-Teste.

Qualificação no AVA 3D – tem como propósito preparar o aprendiz para usar os recursos do AVA 3D que será aplicado como meio de aprendizagem. A capacitação pode ocorrer de diversas formas, como por exemplo, por meio de um cenário genérico com manipulação direta no ambiente; orientação textual do cenário do ambiente que poderá ser entregue aos participantes para conhecimento ou apresentação de um vídeo acompanhado dos comandos básicos específicos do ambiente virtual em questão.

Processo de Aprendizagem no AVA 3D – tem como propósito realizar o processo de aprendizagem no campo de estudo por meio do AVA 3D. Durante a exploração no Ambiente Virtual as interações do aprendiz são capturadas e registradas, de tal forma que identifiquem como foi o comportamento do indivíduo durante o processo de aquisição de conhecimento, para posteriormente correlacionar as ações/interações do aprendiz com os resultados obtidos nas fases de Avaliação Diagnóstica Pré-Teste e Pós-Teste.

Avaliação Diagnóstica Pós-Teste – tem como objetivo avaliar o conhecimento do aprendiz após a exploração no ambiente virtual, com propósito de verificar a sua evolução. Para tanto, utiliza dos mesmos instrumentos de coleta de dados aplicados na fase de Avaliação Diagnóstica Pré-Teste.

Considerando o exposto, é possível verificar que o objetivo principal deste projeto, que visa desenvolver um módulo de monitoramento das interações dos usuários em um sistema destinado a avaliar a aquisição do conhecimento em AVAs 3D, está relacionado diretamente ao módulo **“Processo de Aprendizagem no AVA 3D”**, que compõe o Modelo Teórico de Avaliação da Aprendizagem.

Para automatização do Modelo Teórico de Avaliação da Aprendizagem em AVAs 3D, Nunes *et al.* (2012) propõem a arquitetura do sistema ilustrada na Figura 5, de forma que o modelo possa ser aplicado mais genericamente possível em diversos contextos de AVAs 3D. A arquitetura do sistema apresenta dois componentes principais – i) componente de Interface do AVA 3D; ii) componente do Sistema de Avaliação da Aprendizagem que implementa o Modelo Teórico de Avaliação da Aprendizagem. O componente “Sistema de Avaliação da Aprendizagem”, além de incluir os instrumentos de avaliação diagnóstica, também inclui o módulo de monitoramento das interações do usuário durante a exploração no AVA 3D, com propósito de verificar se tais ambientes estão de fato colaborando com a aquisição de conhecimento dos aprendizes.

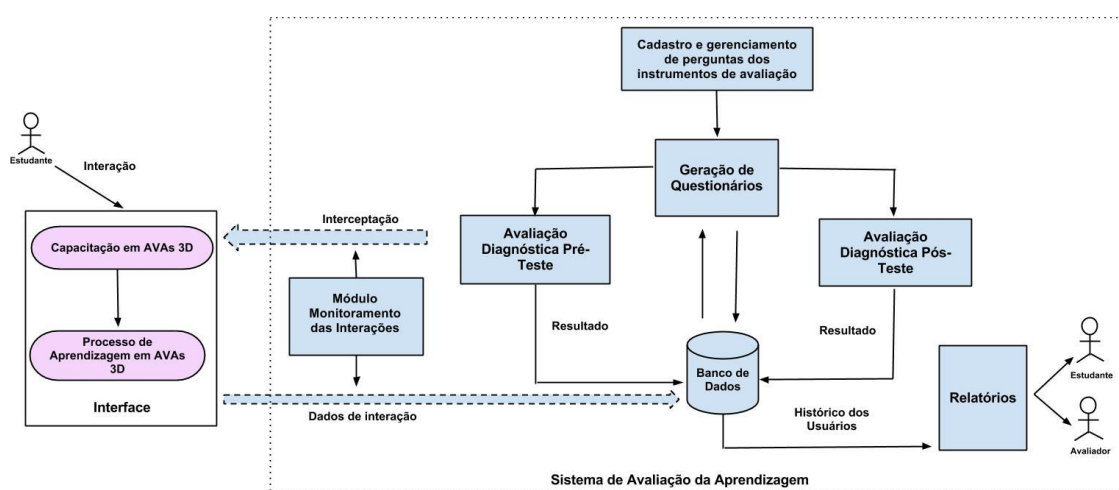


Figura 5. Arquitetura do sistema (NUNES *et al.*, 2012)

Contudo, a proposta desta arquitetura é fazer com que o Modelo Teórico de Avaliação da Aprendizagem seja implementado de maneira independente da aplicação (AVA

3D), tornando-o mais genérico possível, de forma que o componente do Sistema de Avaliação da Aprendizagem intercepte a aplicação para registrar os dados de interação sem a necessidade de se ter um especialista em programação para interligar os sistemas, desde que a plataforma de programação seja a mesma, como pode ser observado na seta tracejada da Figura 5 que apresenta a arquitetura do sistema (NUNES *et al.*, 2012).

Para implementação da arquitetura do sistema considerando todas as suas especificidades, apresenta-se na seção 3.1 as tecnologias utilizadas.

3.1 Tecnologias Utilizadas

A partir da Revisão Sistemática realizada, foi possível identificar na literatura quais os principais métodos e técnicas que possibilitam monitorar os dados de interação do usuário em sistemas computacionais e armazená-los para posterior análise ou tomada de decisão em tempo real. Os métodos e técnicas encontrados permitiram uma melhor compreensão de como ocorre a interceptação das interações humanas em sistemas computacionais.

Dessa forma, considerando as especificidades da arquitetura do sistema e os métodos e técnicas para monitoramento das interações do usuário que foram identificados com a realização da RS, desenvolveu-se o módulo de registro dos dados das interações dos usuários com base na Programação Orientada a Aspectos, por meio da extensão *AspectJ* (ASPECTJ, 2013). A seção 3.2 apresenta os principais conceitos de POA.

O Sistema Gerenciador de Banco de Dados (SGBD) escolhido para implementação da base de dados do Sistema de Avaliação da Aprendizagem, incluindo o módulo de monitoramento das interações dos usuários, foi *h2 Database Engine*⁵ por ser um sistema de banco de dados embarcado, o qual permite o gerenciamento dos dados de forma integrada à aplicação e não requer instalações nem configurações especiais.

A Figura 6 apresenta o modelo lógico do Banco de Dados construído. Este modelo apresenta as funcionalidades a serem consideradas no sistema, como também os níveis de usuários e os dados a serem armazenados. O modelo construído prevê todas as entradas (formas de interação e parâmetros) e saídas (consultas e relatórios necessários).

⁵ H2 Database Engine é um banco de dados de código livre escrito em Java, e pode ser encontrado em <http://www.h2database.com>.

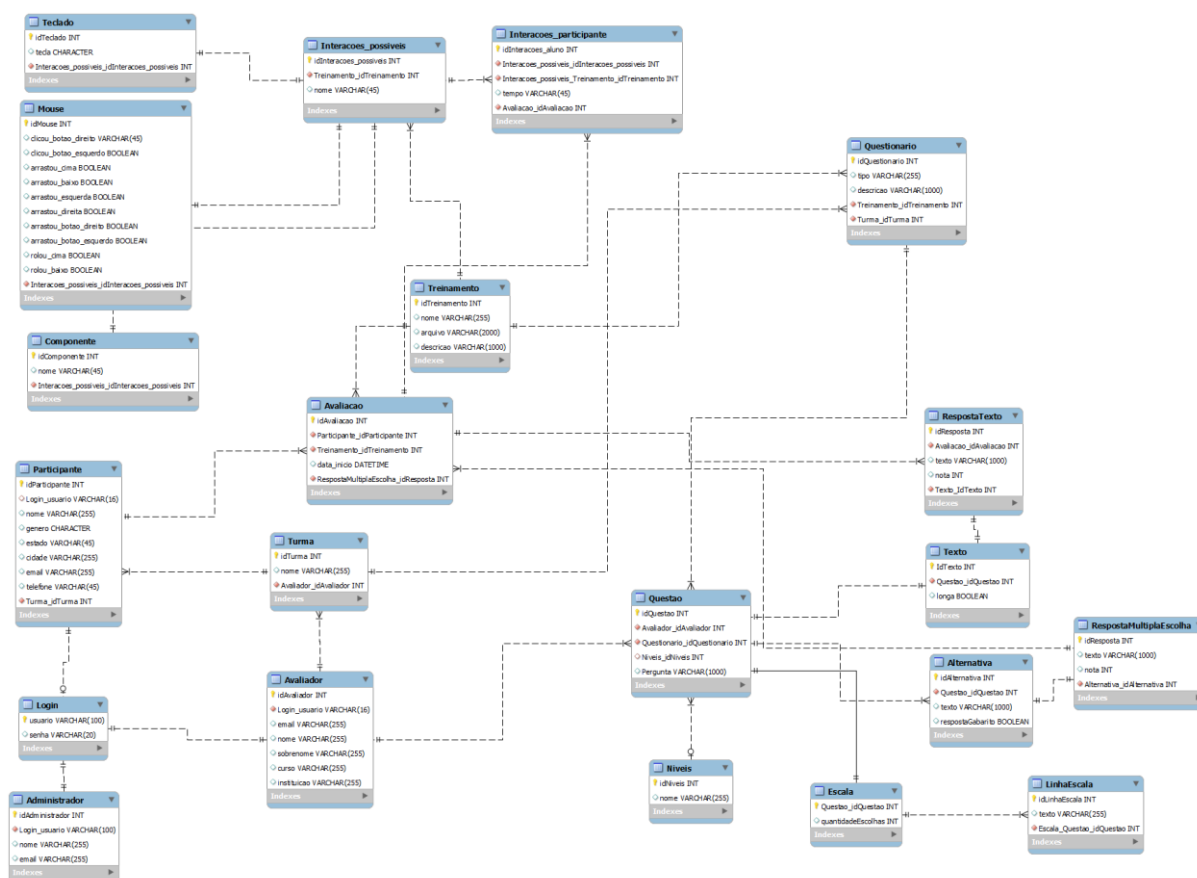


Figura 6. DER do banco de dados do Sistema de Avaliação da Aprendizagem

Para implementação do sistema optou-se pela linguagem de programação Java, por ser uma linguagem de programação que se integra diretamente com a extensão AspectJ, que implementa a Programação Orientada a Aspectos na plataforma Java. Os dados a serem analisados são obtidos por meio do SGBD, que também armazena os registros das interações dos usuários durante a exploração no AVA 3D.

Como estudo de caso para validar a arquitetura e implementação do sistema, incluindo o módulo de registro das interações dos usuários usando POA utilizou-se o AVA 3D do Projeto VIDA (*Virtual and Interactive Distance-Learning on Anatomy*), desenvolvido pelo Interlab/USP e pelo LAPIS (Laboratório de Aplicações de Informática em Saúde) da Escola de Artes Ciências e Humanidades da USP e o AVA Construfig 3D desenvolvido por um grupo de pesquisadores da Universidade Severino Sombra em Vassouras-RJ, com propósito de apoiar o ensino de Geometria Plana e Espacial (MENDES, CARVALHO e CARVALHO, 2007), a serem apresentados na seção 3.2. Os resultados do estudo de caso serão apresentados na seção de Resultados (seção 4.1).

As etapas de implementação da arquitetura do Sistema de Avaliação da Aprendizagem, como também o seu funcionamento serão apresentadas na de Resultados (seção 4.2).

3.2 Programação Orientada a Aspectos (POA)

A POA tem como objetivo modularizar interesses transversais, que com a Programação Orientada a Objetos (POO) ficam misturados e distribuídos dentro do próprio código orientado a objetos. Assim, a POA permite que os módulos sejam organizados de forma independente, devido ao conceito de aspecto, permitindo que o código seja encapsulado e modularizado (CAMARGO e MASIERO, 2006).

De acordo com Camargo e Masiero (2006), a POA permite implementar separadamente os interesses-base, que referem-se à funcionalidade principal do sistema e os interesses transversais que se referem às restrições globais e requisitos não funcionais, como persistência, distribuição, autenticação, controle de acesso e concorrência.

Em um sistema computacional é possível identificar diversos interesses, geralmente divididos em classes. Com a POA é possível modularizar os interesses transversais espalhados nas classes, ou seja, os interesses que estão ao longo de muitas classes podem ser encapsulados em um único módulo chamado Aspecto, o que não era possível utilizando a POO.

De acordo com Camargo e Masiero (2006) a separação de interesses (*concerners*) é um conceito utilizado pela Engenharia de Software, que tem o intuito de separar fisicamente (em termos de código) os interesses do sistema e declaram que:

“Aspecto é o termo usado para denotar a abstração da POA que dá suporte a um melhor isolamento de interesses transversais. Em outras palavras, um aspecto corresponde a um interesse transversal e constitui uma unidade modular projetada para afetar um conjunto de classes e objetos do sistema” (CAMARGO e MASIERO, 2006).

A separação de interesses tem como objetivo melhorar a inteligibilidade do sistema, como também sua manutenção, reúso e evolução (CAMARGO e MASIERO, 2006), que é justamente o propósito da arquitetura do Sistema de Avaliação da Aprendizagem.

Em POA são definidos os seguintes conceitos:

- **Pontos de junção (*join points*):** são pontos identificáveis do sistema, como por exemplo, chamada de um método, execução de um método, lançamento de uma exceção, construtores, entre outros.
- **Ponto de corte (*pointcuts*):** os pontos de corte são agrupamentos de um conjunto de pontos de junção (MASUHARA e KAWAUCHI, 2003).
- **Adendos (*adivices*):** é a estrutura que permite adicionar comportamento dinâmico em um ponto de junção. Existem 3 tipos de adendos: anteriores (*before*), posteriores (*after*) e substituição (*arround*). Um adendo do tipo anterior é executado antes, o posterior depois e o de substituição no lugar de um ponto de junção.

Para exemplificação desses conceitos a Figura 6 exibe um código que grava os dados de interação do usuário cada vez que este executa um evento na aplicação, como por exemplo, um clique com o botão esquerdo do *mouse*. Neste aspecto chamado *Mouse*, definiu-se um ponto de corte chamado *botaoEsquerdo*, depois dos dois pontos definiu-se o ponto de junção, e por fim um adendo no restante do código.

```
public aspect Mouse {
    pointcut botaoEsquerdo(MouseEvent e) : call(void *(MouseEvent)) && args(e);

    before(MouseEvent e) : botaoEsquerdo(e) {
        if (e.getID() == MouseEvent.MOUSE_RELEASED)
        {
            if (e.getButton() == MouseEvent.BUTTON1) // Botão esquerdo do mouse
            {
                // Grava interação
            }
        }
    }
}
```

Figura 7. Exemplo de Aspecto

O aspecto apresentado na Figura 7 intercepta a chamada do método que ocorre na API de gerenciamento de eventos do dispositivo *mouse* da linguagem Java, e então o adendo adicionado irá gravar o movimento antes do método ser invocado.

Para o registro das interações é preciso identificar o ponto de junção desejado, ou seja, precisa-se saber a assinatura dos métodos que serão interceptados. Com o objetivo de concretizar essa tarefa é preciso saber dentro do código de cada AVA 3D o trecho de código que ocorre a interação desejada, o que se torna inviável, tendo em vista que cada implementação segue um modo diferente para utilizar uma mesma interação. Outro motivo

que torna essa tarefa de difícil de executar é o fato de nem todos os AVAs 3D terem o código fonte disponível para análise de onde se encontra uma interação específica.

O problema de definir o ponto junção é contornado identificando trechos de códigos das *APIs* de manipulação de eventos do *JAVA*, já que, por ser de código aberto, se conhece as assinaturas dos métodos para interceptação, e também por ser utilizada por todas as aplicações escritas em *JAVA* que precisem manipular esses eventos gerados pelos dispositivos.

Vale ressaltar que a POA permite a obtenção dos dados sem adicionar complexidade ao código da aplicação entrecortada (BATEMAN *et al.*, 2009). Ou seja, essa operação pode ser feita de forma não intrusiva ao código da aplicação, visto que o módulo que agrupa o interesse entrecortante está separado da aplicação monitorada.

3.3 Estudo de Caso

Como estudo de caso para verificar a viabilidade do uso de POA na implementação do registro das interações do usuário em AVAs 3D, foram selecionados dois ambientes virtuais implementados na plataforma Java, sendo o Projeto VIDA (*Virtual and Interactive Distance-Learning on Anatomy*) (TORI *et al.*, 2009) e o CONSTRUFIG 3D (MENDES, CARVALHO e CARVALHO, 2007).

Os testes foram iniciados usando o Projeto VIDA, desenvolvido pelo Interlab (Laboratório de Tecnologias Interativas) da Escola Politécnica da USP e pelo LApIS (Laboratório de Aplicações de Informática em Saúde) da Escola de Artes Ciências e Humanidades da USP, que visa o desenvolvimento de um AVA 3D.

Na versão utilizada neste estudo, o Ambiente Virtual VIDA apresenta as diferentes fases de um feto (3D) durante sua gestação, com propósito de identificar as etapas da evolução, período de gestação, entre outras informações de interesse dos alunos dos cursos da área de saúde. O ambiente suporta apenas a interação com dispositivos comuns, como *mouse* e teclado. Na Figura 8 pode ser observada a interface do sistema no qual o participante pode interagir com um modelo tridimensional de um feto.

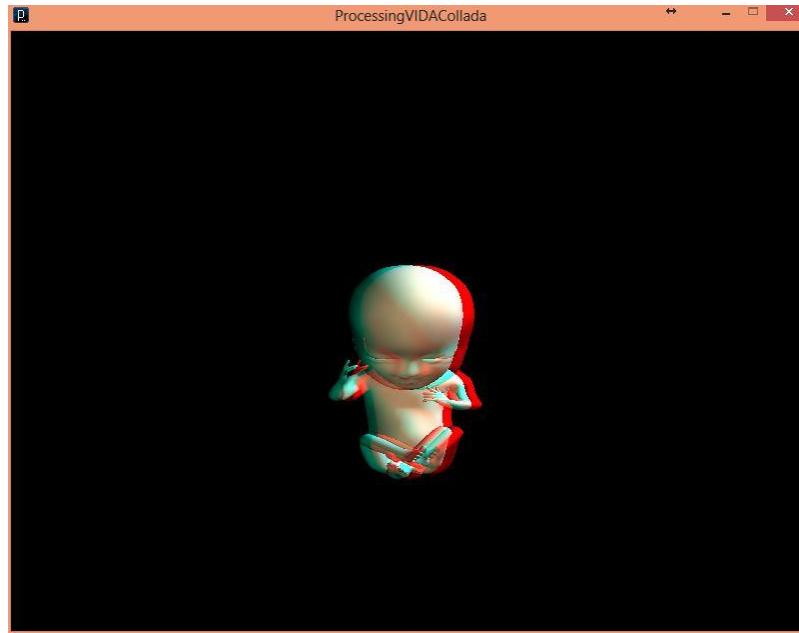


Figura 8. Projeto VIDA (fases do feto durante a gestação)

O sistema se baseia na visão estereoscópica para oferecer maior realismo na visualização dos objetos tridimensionais. Os parâmetros da projeção estereoscópica são calibrados de forma que cada usuário tenha a sensação de estar visualizando o feto no espaço físico entre o seu corpo e o monitor, produzindo um efeito de profundidade (TORI *et al.*, 2009).

A visualização estereoscópica pode ser gerada e visualizada por diversos dispositivos e técnicas. Neste caso a forma de visualização estereoscópica foi o anaglifo, que não exige monitores ou projetores especiais, apenas um par de óculos descartável de duas cores com filtros coloridos (cada lente com uma cor diferente), produzindo o efeito 3D desejado (TORI *et al.*, 2009).

Como segundo estudo de caso, foram realizados testes do módulo de monitoramento das interações do usuário usando a ferramenta computacional voltada para o ensino de Geometria Plana e Espacial, denominada CONSTRUFIG3D (MENDES, CARVALHO e CARVALHO, 2007).

A ferramenta foi desenvolvida para auxiliar alunos que estão iniciando o estudo de Geometria. O sistema auxilia na identificação de figuras geométricas planas e espaciais, e permite a composição de figuras espaciais a partir da seleção de figuras planas. Por meio de uma abordagem jogo/simulação o aluno desenvolve conceitos sobre figuras espaciais a partir

de figuras planas. Com uma interface simples, a ferramenta utiliza conceitos de Computação Gráfica, apresentando áreas de trabalho 2D e 3D e permite rotacionar a figura espacial criada, visualizando vértices, arestas e faces.

O ambiente apresenta os seguintes componentes: i) botão para selecionar o número de figuras planas que irão compor a figura espacial (área 2D), ii) figuras planas que serão usadas para compor a figura espacial (área 2D), iii) botões como limpar, rodar, sair, sobre e o botão 3D que tem como finalidade criar a figura espacial a partir das informações enviadas (área 2D).

Para utilizar o ambiente o usuário escolhe de forma livre figuras geométricas planas, que irão compor uma figura espacial. Se as figuras planas escolhidas forem coerentes, o usuário irá visualizar e interagir com a figura espacial gerada na área de trabalho 3D, na qual é possível interagir com a figura. Como pode ser visto na Figura 9.

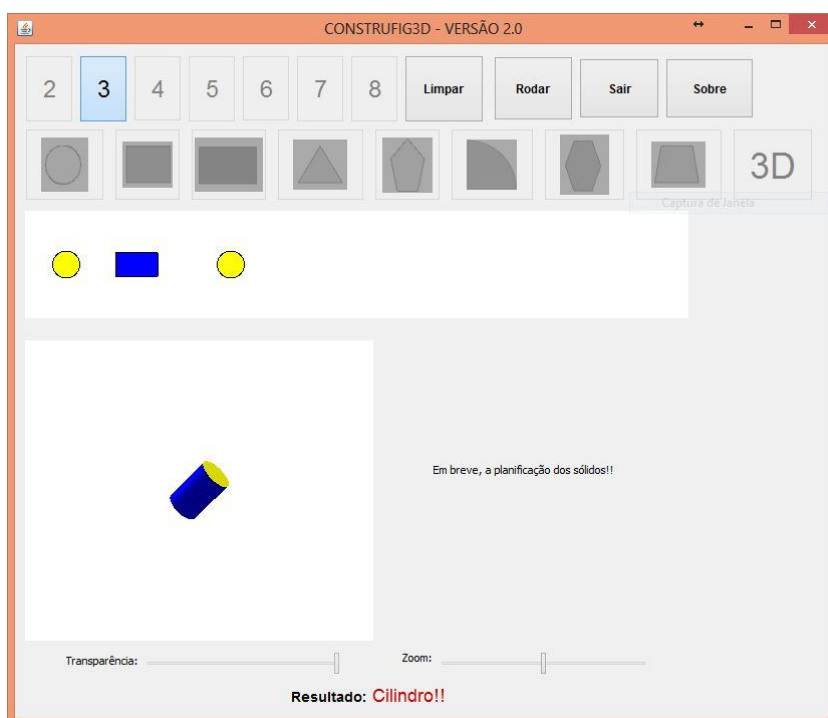


Figura 9. Sistema CONSTRUFIG 3D (versão 2.0)

4 Resultados

Como resultados da pesquisa, temos a Revisão Sistemática realizada e apresentada na seção do Estado da Arte, tendo sido submetido o artigo “Uma Revisão Sistemática sobre o Registro das Interações do Usuário em Ambientes Virtuais de

Aprendizagem Tridimensionais” ao SBIE – Simpósio Brasileiro de Informática na Educação, o qual está aguardando o resultado.

Além da RS, tem-se os resultados referentes à implementação do Sistema de Avaliação da Aprendizagem a ser apresentada na seção 4.2, incluindo o módulo de monitoramento das interações do usuário, a ser apresentado na seção 4.1.

4.1 Módulo de Monitoramento das Interações do Usuário

Esta subseção apresenta o Módulo de Monitoramento das Interações dos Usuários e seus componentes, tendo a sua implementação baseada em POA. O módulo de monitoramento é ilustrado na Figura 10.

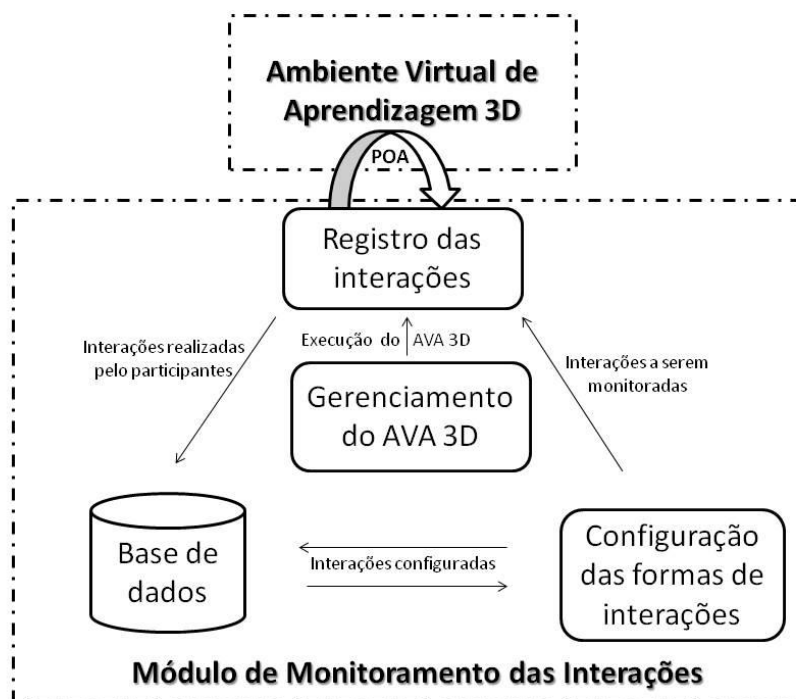


Figura 10. Módulo de Monitoramento das Interações dos Usuários

O módulo de monitoramento das interações está inserido no Sistema de Avaliação da Aprendizagem e compartilha o mesmo banco de dados. Todos os componentes do sistema de monitoramento são descritos a seguir:

- **Registro das interações** - neste módulo são interceptadas as interações realizadas pelo participante durante a exploração no AVA 3D. Tais interações são registradas, somente, se tiverem sido previamente cadastradas pelo avaliador no

Sistema de Avaliação da Aprendizagem. Durante a interceptação também são registrados o tempo de cada interação e o tempo total de exploração no AVA 3D.

- **Gerenciamento do AVA 3D** - este módulo tem como objetivo cadastrar, editar ou excluir um AVA 3D. Porém, este módulo não foi completamente desenvolvido, ou seja, na versão atual do sistema não é possível o avaliador cadastrar um AVA 3D por meio da interface do Sistema de Avaliação da Aprendizagem, devido a complexidade do problema e curto prazo para implementação. Dessa forma, este módulo foi adaptado, inserindo-se os AVAs 3D diretamente na IDE (*Integrated Development Environment*).

Portanto, para iniciar o Sistema de Avaliação da Aprendizagem, executa-se o arquivo (.jar) referente ao AVA 3D desejado. A partir da execução do AVA 3D, é exibida a tela de *login* do sistema. Uma vez realizada a autenticação no perfil desejado, todas as atividades executadas no sistema serão referentes ao AVA 3D executado inicialmente.

Quando se executa pela primeira vez o AVA 3D, as informações sobre o mesmo são coletadas, a fim de gerar uma identificação única. Ao iniciar pela segunda vez o ambiente, o sistema identifica que este já foi cadastrado e as informações que foram associadas ao AVA 3D são recuperadas.

- **Configuração das interações** - responsável pelo cadastro/edição/exclusão dos possíveis tipos de interações a serem monitoradas no AVA 3D, ou seja, quais os tipos de interações serão possíveis de serem capturadas quando o participante estiver explorando o AVA e quais os eventos associados. O avaliador cadastra os tipos de interações a serem monitoradas, executando-as no próprio AVA 3D.

Para iniciar o módulo de monitoramento das interações, precisa-se executar o Sistema de Avaliação da Aprendizagem (aplicação Java), selecionando-se o AVA 3D e executá-lo juntamente com o compilador do AspectJ.

Quando o AVA 3D selecionado é iniciado, a função apresentada na Figura 11 intercepta a execução do método principal por meio de um ponto de corte, e então armazena em variáveis uma referência à classe e ao método, para que futuramente esse método seja executado novamente.


```
pointcut callMainMethod(String[] args) : execution(public static void main(String[])) && args(args);

before(String[] args): callMainMethod(args)
{
    Aplicacao ava = Aplicacao.getInstance();
    if(!ava.isArgsAtribuido())
    {
        ava.setClasseAplicacao(thisJoinPointStaticPart.getSignature().getDeclaringType());
        ava.setArgs(args);
        new Fluxo().start();
    }
}
```

Figura 11. Ponto de corte do método principal do AVA 3D

Apesar do AVA 3D estar em execução, a sua exibição é impedida como ilustra a Figura 12. Na sequência é executado e exibido o Sistema de Avaliação da Aprendizagem. Assim o usuário do sistema depara-se com a tela de login, que lhe dará acesso ao Sistema de Avaliação da Aprendizagem no perfil desejado.

```
pointcut visibilidade() : call (void Frame.setVisible(boolean));

void around(Frame atual, boolean b) : visibilidade() && args(b) && target(atual)
{
    Aplicacao ava = Aplicacao.getInstance();
    if(ava.isprimeiraVisualizacao() && !(atual==Fluxo.getInstance().getAtual()))
    {
        //Não permite a exibição da aplicação
        b=false;
        //controle
        ava.setprimeiraVisualizacao(false);
        ava.setAplicacao(atual);
        TreinamnetoDAO t = new TreinamnetoDAO();
        t.inserir(atual.toString());
    }
    proceed(atual, b);
}
```

Figura 12. Ponto de corte de controle da visibilidade do AVA 3D

No caso do perfil de avaliador, quando o usuário executar a opção de Gerenciamento das Interações que se encontra disponível no menu Treinamento, seja para gerenciar as interações já cadastradas ou para cadastrar uma nova interação, o método principal referente ao AVA 3D será invocado para exibir o ambiente virtual na tela do sistema. No caso do perfil de participante do treinamento, ao executar a opção Iniciar Treinamento disponível no menu Treinamento, o método principal do AVA 3D será novamente invocado, com objetivo de iniciar o ambiente com sua configuração inicial. A função para invocação do método é ilustrada na Figura 13.

```
public void inicia()
{
    if(this.principal!=null)
    {
        try {
            principal.invoke(null, this.args);
        } catch (IllegalAccessException | IllegalArgumentException
            | InvocationTargetException e) {
            e.printStackTrace();
        }
    }
}
```

Figura 13. Invocação do método principal que exhibe o AVA 3D

Enquanto o participante do treinamento está interagindo com o ambiente suas interações são monitoradas. Essas interações podem ser movimentos/cliques realizados com o *mouse*, teclas pressionadas ou manipulação dos botões da interface. O registro das interações é de extrema importância para o avaliador analisar o comportamento do participante durante o processo de aprendizagem. O armazenamento das interações é efetuado com um certo intervalo de tempo para não sobrecarregar o ambiente, porque muitas requisições ao disco rígido o deixaria lento, e consequentemente, poderia prejudicar o desempenho do sistema.

O monitoramento é realizado adicionando-se pontos de corte aos métodos da API do *Java* que manipulam os eventos do *mouse* (*MouseEvent*), eventos do teclado (*KeyEvent*) e os eventos relacionados aos componentes da interface (*ActionEvent*). Nos pontos de corte são analisados os tipos das ações realizadas pelo usuário, buscando verificar se a interação realizada é equivalente a uma das interações cadastradas previamente pelo avaliador, antes que a mesma seja armazenada no banco de dados.

No entanto, para cadastrar os eventos do *mouse* (*MouseEvent*), eventos do teclado (*KeyEvent*) e os eventos relacionados aos componentes da interface (*ActionEvent*), foi necessário estabelecer, inicialmente, o que poderia ser monitorado em cada evento, tendo em vista que o sistema pode estar lidando com qualquer AVA 3D (construído na plataforma *Java*).

Dessa forma, para os eventos do *mouse* as interações possíveis de serem monitoradas são: cliques com botão direito ou esquerdo; arrastar para cima, para baixo, direita e esquerda com o botão direito ou esquerdo; rolar para cima ou para baixo. Para os eventos com teclado, é possível monitorar as teclas pressionadas. E quanto aos eventos com os botões da interface, o monitoramento acontece verificando-se o evento associado ao componente da

interface. Tais especificações de monitoramento são implementadas no código fonte do sistema.

Para comparar uma interação realizada pelo participante com uma já cadastrada pelo avaliador, um meio de identificar a interação foi necessário. Para as ações desempenhadas com o *mouse* são gravados booleanos das ações possíveis descritas acima. Para o teclado é armazenado o caractere correspondente à tecla. Por fim, para os botões da interface são armazenados os dados referentes à tela que ele pertence, sua posição na mesma e seu tamanho em pixels.

4.2 Testes de Validação do Módulo de Monitoramento das Interações

Para validar o funcionamento do módulo de monitoramento das interações, que compõe o Sistema de Avaliação da Aprendizagem, de forma a não interferir no código fonte da aplicação, foram usados como estudo de caso os Projetos VIDA e CONSTRUFIG 3D já descritos na seção 3.3, além de outros ambientes virtuais simples desenvolvidos para os testes iniciais.

O módulo de monitoramento das interações do usuário é utilizado tanto no perfil de avaliador quanto no perfil de participante. Como avaliador, o módulo de monitoramento é utilizado para cadastrar novas interações e gerenciar as interações já cadastradas. Para cadastrar uma nova interação, o AVA 3D é exibido na tela, conforme ilustra a Figura 14 que mostra o sistema sendo executado usando o Projeto VIDA (AVA 3D).

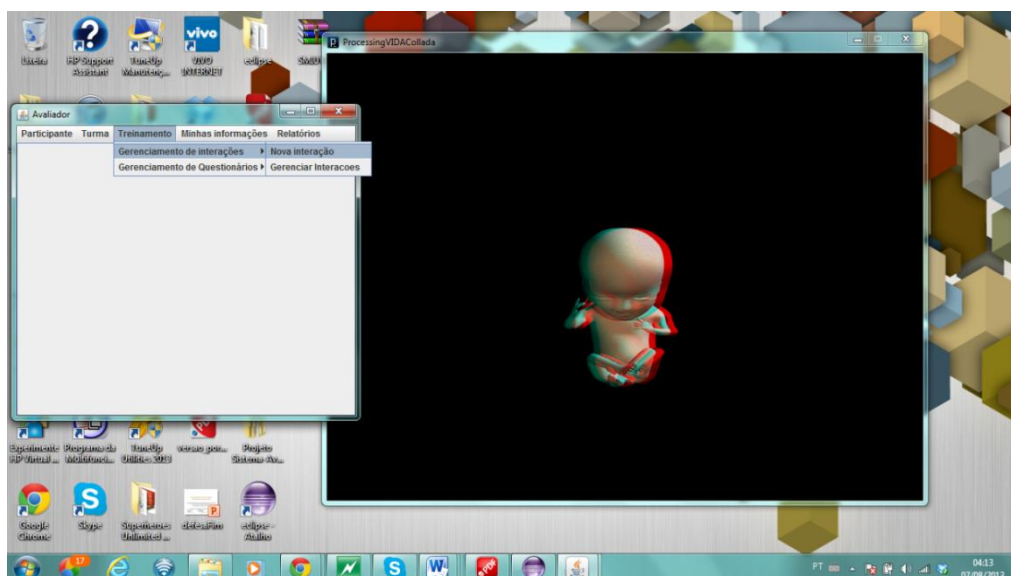


Figura 14. Sistema de Avaliação da Aprendizagem (perfil avaliador) – cadastro nova interação

O registro das interações é armazenado em tempo real durante a execução da interação no próprio ambiente, como ilustram as Figuras 15(a) e 15(b). Logo após a interação realizada, aparecerá a tela para confirmar ou refazer a interação (Figura 15b). Antes de qualquer treinamento ser iniciado o avaliador deve cadastrar todas as interações possíveis, as quais ele gostaria que fossem monitoradas quando o participante realizar o treinamento.

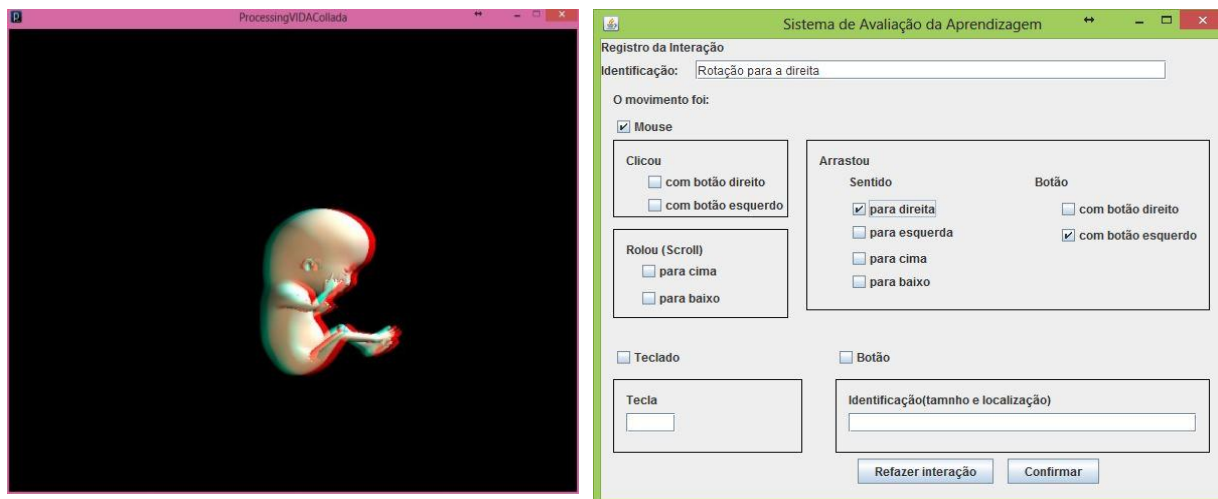


Figura 15(a). Movimento realizado no AVA 3D VIDA

Figura 15(b). Registro da interação realizada no AVA 3D

Em relação às interações já cadastradas, o avaliador pode buscar uma interação armazenada no banco de dados, como ilustra a Figura 16. Dessa forma, é possível executar as ações de editar, refazer ou excluir a referida interação. Para tal, ao selecionar a interação desejada, será exibida uma tela similar à ilustrada na Figura 15(b), que inclui o botão excluir.

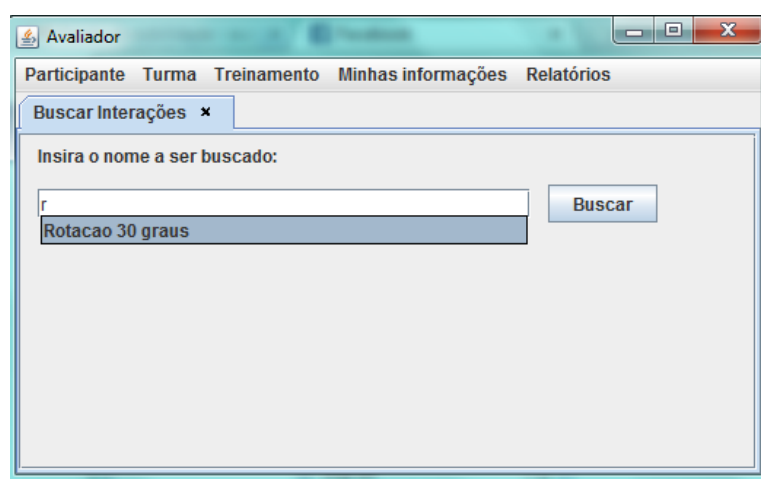


Figura 16. Sistema de Avaliação da Aprendizagem (perfil avaliador) – gerenciamento das interações

Ainda buscando validar o módulo de monitoramento das interações do usuário, o Sistema de Avaliação da Aprendizagem foi executado com o ambiente CONSTRUFIG 3D.

A Figura 17 exibe a interface do Sistema de Avaliação da Aprendizagem, no perfil de avaliador, buscando cadastrar uma nova interação no referido ambiente.

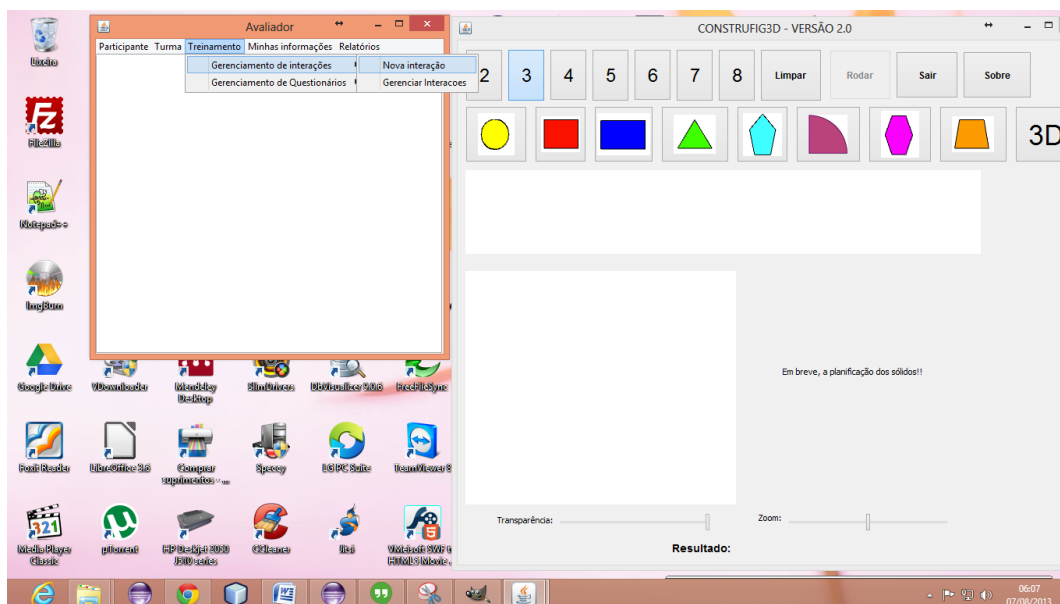


Figura 17. Sistema de Avaliação da Aprendizagem (perfil avaliador) – cadastro nova interação no CONSTRUFIG 3D

O registro das interações é armazenado em tempo real durante a execução da interação no próprio ambiente, como ilustram as Figuras 18(a) e 18(b).

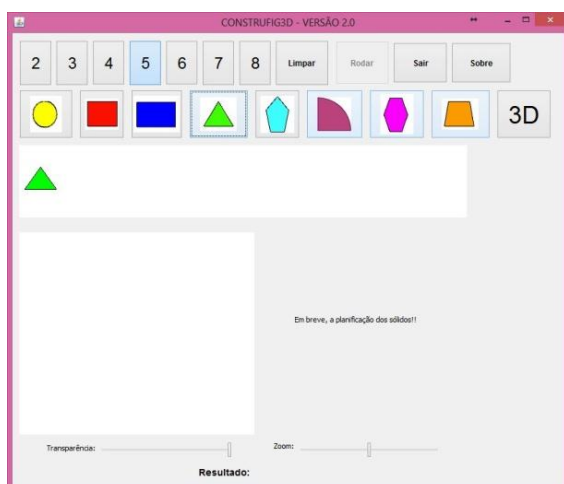


Figura 18(a). Evento botão selecionado na interface

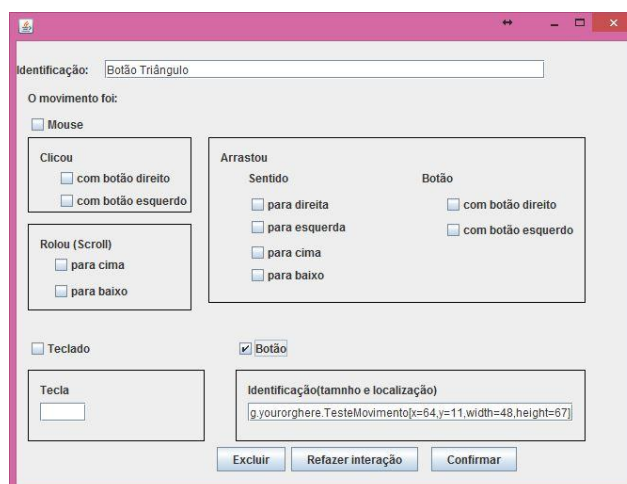


Figura 18(b). Registro da interação realizada no AVA 3D

Acessando o Sistema de Avaliação da Aprendizagem no nível de participante, é possível registrar as interações do usuário, enquanto o mesmo realiza o treinamento no ambiente virtual. Tal ação é disparada, a partir do momento que o participante executar a opção Iniciar treinamento disponível no sistema, por meio do menu Treinamento.

Vale salientar que o Projeto VIDA tem suas interações focadas na manipulação do ambiente usando os dispositivos *mouse* e teclado para interagir diretamente com o objeto virtual (feto), sendo possível realizar ações como rotação, translação e escala, não apresentando nenhum outro componente na interface, a não ser o próprio objeto virtual. Já o CONSTRUFIG 3D, além de apresentar a área 3D para manipulação do objeto virtual, disponibilizando ações de rotação e escala, apresenta também uma carga expressiva de botões no componente da interface (área 2D), o que pode conduzir o usuário a uma maior frequência de interações com o ambiente, especialmente em relação às tentativas de criar diferentes figuras geométricas espaciais.

A partir dos testes realizados, é possível afirmar que o módulo de monitoramento das interações do usuário foi capaz de interceptar todos os eventos previamente definidos, obtendo sucesso no registro das interações do usuário.

4.3 Funcionamento do Sistema de Avaliação da Aprendizagem

Apesar do desenvolvimento do Sistema de Avaliação da Aprendizagem apresentado na Figura 5 não estar previsto, inicialmente, no plano de atividades do projeto de pesquisa em questão, este se tornou um objetivo secundário devido a sua proximidade com o Módulo de Monitoramento das Interações, uma vez que tendo o Sistema de Avaliação da Aprendizagem implementado, é possível validar o módulo de monitoramento das interações dentro do contexto que ele se encontra inserido. Dessa forma apresenta-se nesta seção uma breve descrição do funcionamento do Sistema de Avaliação da Aprendizagem.

Inicialmente, a primeira versão do sistema foi desenvolvida para *desktop*. Dessa forma, é necessário que todos os participantes da turma a ser avaliada, executem o sistema no mesmo computador, buscando garantir que o registro das interações de cada participante durante a exploração no AVA 3D, como também os dados coletados por meio dos instrumentos de avaliação sejam todos armazenados no banco de dados. Isso se dá, pelo fato do banco de dados estar disponível apenas localmente, como já explicado na seção 3.1.

Após executar o sistema, o primeiro passo deve ser logar usuário no sistema de autenticação, como ilustra a Figura 19. A validação do sistema prevê três perfis de usuários, são eles: i) administrador; ii) avaliador e iii) participante.

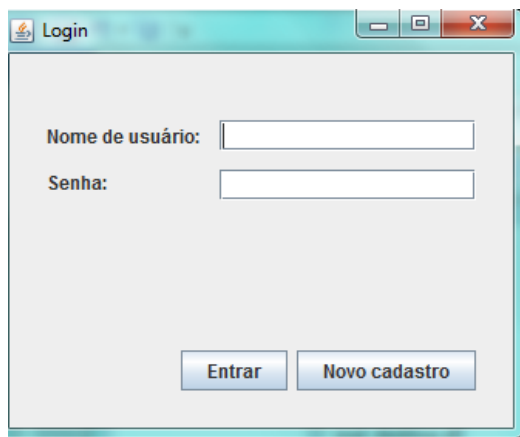


Figura 19. Sistema de Avaliação da Aprendizagem – tela de autenticação

Para iniciar a aplicação pela primeira vez, deve-se logar como administrador e em seguida será exibida a tela que é ilustrada na Figura 20.

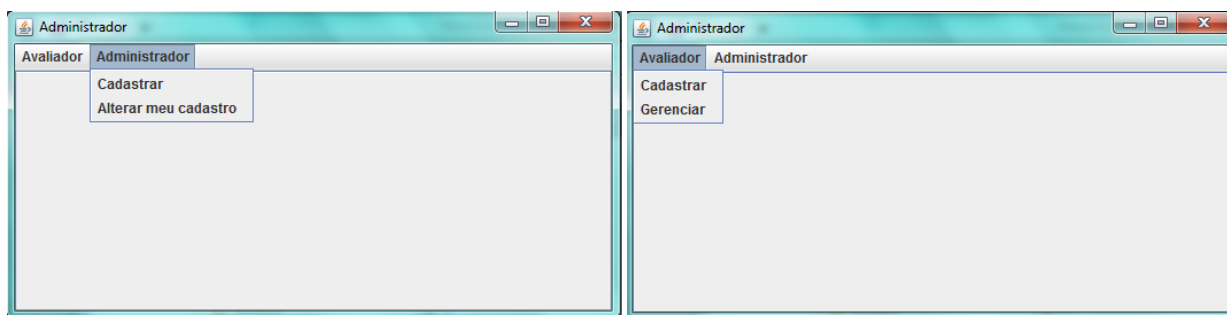


Figura 20. Sistema de Avaliação da Aprendizagem – tela de administrador

Observando a Figura 20, verifica-se duas funcionalidades na barra de menu (Avaliador e Administrador). Acessando no menu a opção avaliador, é possível executar as ações cadastrar e gerenciar avaliador. Salienta-se que o próprio administrador pode ser cadastrado como avaliador, desde que este indique um nome de usuário diferenciado. Acessando a barra de menu na opção administrador, é possível alterar os dados do próprio administrador ou cadastrar um novo administrador para a aplicação.

Em seguida, a aplicação deve ser executada com o login de avaliador, para dar sequência ao Sistema de Avaliação da Aprendizagem. A tela de avaliador é exibida na Figura 21(a) e 21(b).

Figura 21(a). Tela de autenticação no sistema (perfil avaliador)

Figura 21(b). Tela inicial (perfil avaliador)

Observa-se na Figura 21(b) a barra de menu do perfil avaliador, cujas opções são descritas a seguir:

1. Participante – permite buscar participante da turma por nome, visando consultar e/ou alterar os dados do participante, conforme mostra a Figura 22.

Figura 22. Tela menu participante (perfil avaliador)

2. Turma – como o acesso ao sistema é permitido apenas aos usuários previamente cadastrados, o avaliador deverá cadastrar a turma a ser avaliada e os participantes vinculados à turma, como ilustra as Figuras 23(a) e 23(b). Para cada turma cadastrada é possível renomear a turma (**Editar Turma**), também é possível inserir, excluir ou alterar alunos da turma (**Editar Alunos**), além de ser possível excluir a turma (**x**), lembrando que se a turma for excluída, todos os participantes da turma serão excluídos. Para adicionar uma nova turma, basta informar o nome da turma, clicar no botão adicionar que o sistema irá exibir a tela apresentada na Figura 23(b) para adicionar os e-mails dos alunos à turma.

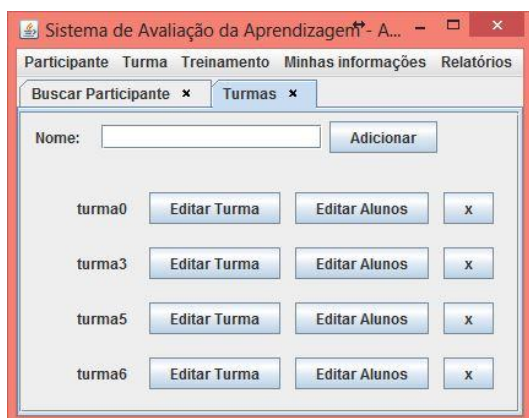


Figura 23(a). Tela adicionar ou editar turma (perfil avaliador)

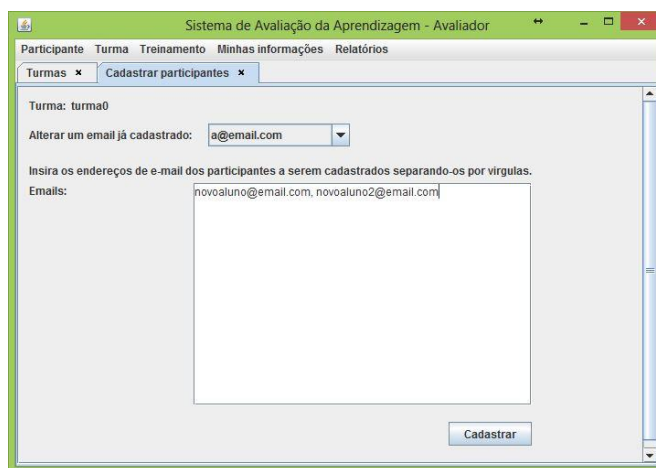


Figura 23(b). Tela adicionar participantes na turma (perfil avaliador)

3. Treinamento – apresenta um submenu com as opções: *gerenciamento de interações e gerenciamento de questionários* (Figura 24).

O gerenciamento de interações, já foi apresentado na seção 4.3, na qual foi descrito os testes realizados com o módulo de monitoramento das interações do usuário. Esta funcionalidade inclui o cadastro de uma nova interação no AVA 3D, como também possibilita gerenciar as interações já cadastradas (refazer interação, alterar algum dado da interação e excluir interação).

O gerenciamento de questionários inclui as opções gerenciar questionários e cadastro de questões das Avaliações Diagnósticas Pré-Teste e Pós-Teste. Em relação a gerenciar questionários, é possível cadastrar e editar questionários (perfil, *feedback*, capacitação, pré-teste e pós-teste). No caso dos questionários de Caracterização do Perfil, *Feedback* e Capacitação, também serão adicionadas as perguntas. Já no caso dos questionários de Avaliações Diagnósticas Pré-Teste e Pós-Teste. Nesta etapa será criado o

questionário e gerado com questões aleatórias, a partir das questões cadastradas previamente (Figura 25).

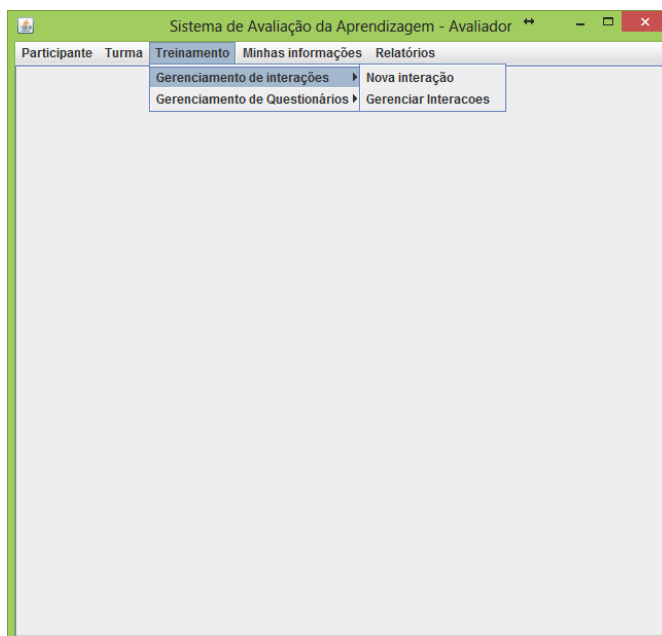


Figura 24. Tela menu treinamento (perfil avaliador)

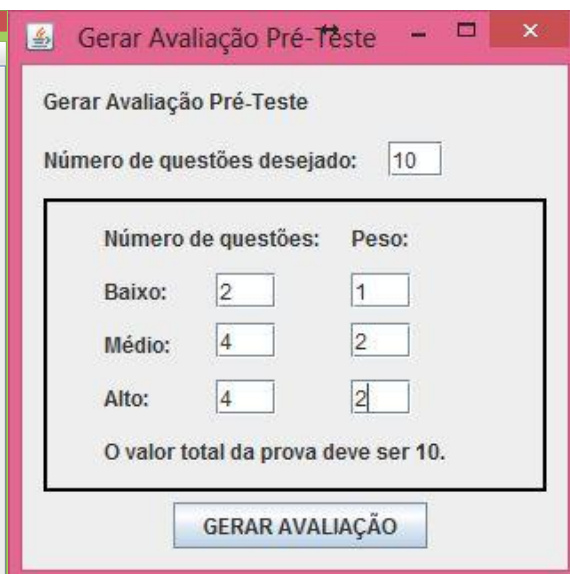


Figura 25. Tela gerar avaliação (perfil avaliador)

O cadastro das questões é realizado pelo submenu cadastro de questões de Avaliações Diagnósticas Pré-Teste e Pós-Teste, no submenu gerenciamento de questionários que se encontra no menu treinamento.

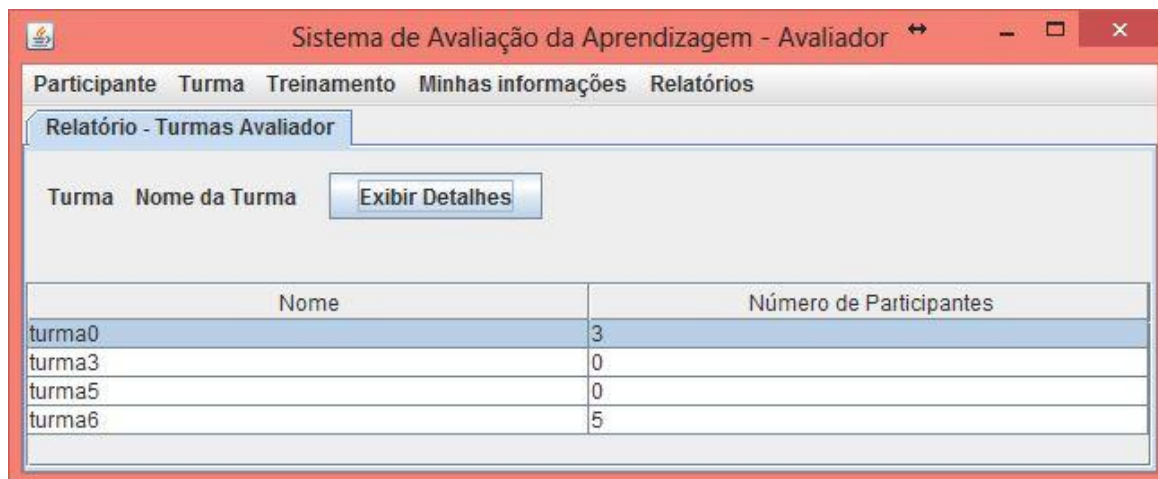
Para adicionar perguntas ao questionário, deve-se selecionar o tipo de pergunta no campo adicionar pergunta. O sistema irá disponibilizar cinco tipos de perguntas para todos os tipos de questionários, são elas: i) texto curto (recomendável para uma resposta curta de uma única linha); ii) texto longo (recomendável para respostas com mais de uma linha); iii) múltipla escolha (apenas uma das alternativas deve ser a correta); iv) afirmativas (pode-se ter várias afirmativas corretas para assinalar) e v) escala.

No caso das perguntas relacionadas aos questionários de Avaliações Diagnósticas Pré-Teste e Pós-Teste, também será informado no momento do cadastro da pergunta o nível de cada questão (fácil, médio ou difícil).

Todas as perguntas adicionadas ao questionário selecionado serão adicionadas a tela de Questionário, permitindo editar a pergunta posteriormente, caso seja necessário.

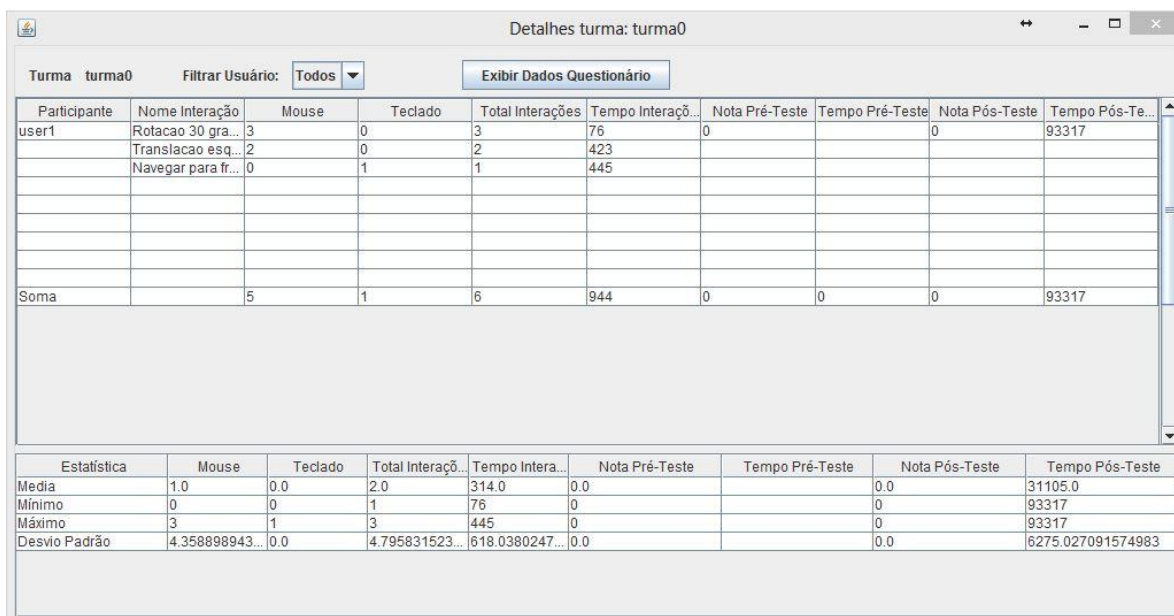
4. Minhas Informações - permite alterar os dados do cadastro do avaliador.

5. Relatórios – alguns relatórios foram implementados e estão disponíveis no Sistema de Avaliação da Aprendizagem. No entanto, a implementação deste módulo encontra-se em andamento. Alguns dos relatórios disponíveis no sistema são exibidos nas Figuras 26 e 27. Os relatórios são gerados, a partir dos dados armazenados no banco de dados.



Nome	Número de Participantes
turma0	3
turma3	0
turma5	0
turma6	5

Figura 26. Tela relatório das turmas cadastradas (perfil avaliador)



Participante	Nome Interação	Mouse	Teclado	Total Interações	Tempo Interaç...	Nota Pré-Teste	Tempo Pré-Teste	Nota Pós-Teste	Tempo Pós-Te...
user1	Rotacao 30 gra...	3	0	3	76	0		0	93317
	Translacao esq...	2	0	2	423				
	Navegar para fr...	0	1	1	445				
Soma		5	1	6	944	0	0	0	93317

Estatística	Mouse	Teclado	Total Interaç...	Tempo Intera...	Nota Pré-Teste	Tempo Pré-Teste	Nota Pós-Teste	Tempo Pós-Teste
Media	1.0	0.0	2.0	314.0	0.0		0.0	31105.0
Mínimo	0	0	1	76	0		0	93317
Máximo	3	1	3	445	0		0	93317
Desvio Padrão	4.358898943...	0.0	4.795831523...	618.0380247...	0.0		0.0	6275.027091574983

Figura 27. Tela detalhes da turma (perfil avaliador)

A aplicação também pode ser executada no perfil de participante. Para isso, o participante ao executar o sistema pela primeira vez, deverá iniciar pelo botão novo cadastro disponível na tela de autenticação do sistema. Ao clicar no botão novo cadastro será exibida a tela ilustrada na Figura 28, na qual ele deve informar o email que foi usado pelo avaliador na realização do pré-cadastro dos participantes da turma.

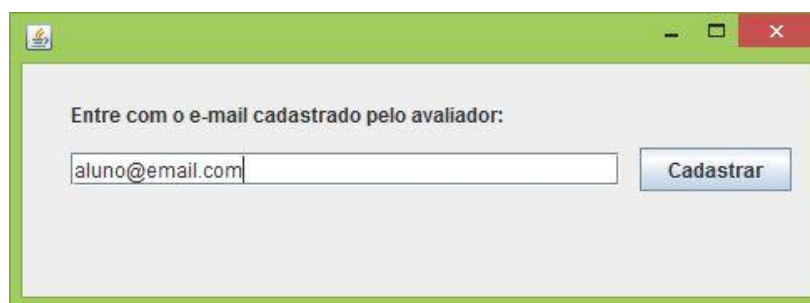


Figura 28. Cadastro do participante (perfil participante)

Na sequência o participante deve completar o seu cadastro e salvá-lo no sistema. Após a realização do cadastro, o participante pode iniciar o treinamento virtual, conforme as instruções do avaliador. A Figura 29 apresenta a interface do sistema, na qual estão disponíveis o menu Treinamento, que apresenta as opções Iniciar Capacitação e Iniciar Treinamento (AVA 3D) e o menu Avaliação, no qual estão disponíveis os questionários que devem ser respondidos pelo participante (Perfil, *Feedback*, Capacitação, Avaliação Diagnóstica Pré-Teste e Pós-Teste).

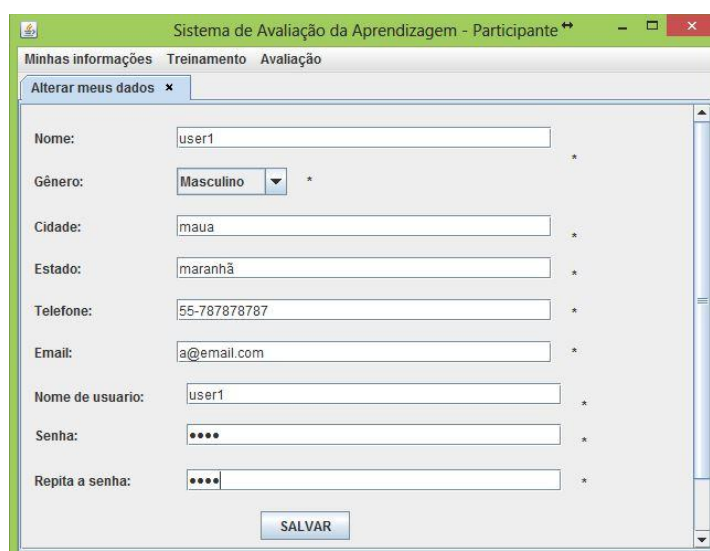


Figura 29. Interface do sistema (perfil participante)

As respostas dos questionários respondidos pelo participante são armazenadas no banco de dados do sistema. Quanto aos questionários de Avaliação Diagnóstica Pré-Teste e Pós-Teste, além das respostas, também é armazenado o tempo de realização da avaliação.

Nesta primeira versão do sistema, o módulo de **Capacitação para uso de sistemas de Realidade Virtual** está implementado na forma de questionário, ou seja, o participante deve responder algumas questões relacionadas ao uso de AVAs 3D dentro do contexto em estudo, para que o avaliador verifique por meio das suas respostas, se o mesmo está apto para iniciar o treinamento. No entanto, em uma próxima versão do sistema, pretende-se que o participante execute, de fato, uma tarefa em um ambiente virtual tridimensional com propósito de adquirir familiaridade com sistemas de Realidade Virtual e assim não comprometer o seu desempenho escolar. A partir do registro das interações no ambiente durante a fase de capacitação, o avaliador terá condições de verificar se o participante está apto para avançar para a próxima etapa, que seria iniciar o treinamento no AVA 3D em estudo.

Considerações Finais

Das técnicas encontradas na RS, aquelas que melhor se aplicam ao objetivo deste trabalho é o *Proxy* (usado entre o gerenciamento de eventos da linguagem Java e a aplicação) e a Programação Orientada a Aspectos. Apesar de serem limitados a sistemas computacionais implementados em Java, tais abordagens podem ser aplicadas a sistemas já existentes, de forma não invasiva ao código fonte da aplicação monitorada, e não dependem de um sistema operacional específico.

Na maioria dos trabalhos encontrados na RS, os recursos da linguagem de programação Java foram utilizados. Isto leva a induzir que seja pelo fato da linguagem Java oferecer técnicas de monitoramento das interações independente do sistema operacional, considerando que quanto maior a flexibilidade da ferramenta de monitoramento, maior será o número de aplicações possíveis de serem monitoradas. Este fato reforça a decisão pela linguagem AspectJ como principal recurso para implementação do módulo de monitoramento das interações dos usuários que compõe o Sistema de Avaliação da Aprendizagem.

Vale salientar que mais de 50% dos trabalhos incluídos na fase de seleção final e extração de resultados da RS apresentaram técnicas de monitoramento que não são invasivas ao código fonte da aplicação, o que é essencial em um cenário no qual o código da aplicação não está disponível.

Sendo assim, uma ferramenta de monitoramento interativa, não invasiva, adaptável a qualquer tipo de aplicação e independente de sistema operacional favorece a comunidade científica de RV, RA, AVAs e áreas correlatas. Porém, é um objetivo difícil de ser cumprido completamente, visto que as diferentes tecnologias que apoiam as linguagens de programação e os sistemas operacionais não oferecem suporte nativo para o problema em questão.

Os resultados das implementações confirmam que o uso da POA foi adequada ao desenvolvimento desta pesquisa, devido as suas características, e por ser possível executar o módulo de registro das interações dos usuários em um maior número possível de AVAs 3D.

Referências

- ASPECTJ. *The AspectJ Project*, Disponível em <http://www.eclipse.org/aspectj/>, Acessado em 28/01/2013.
- AZZOPARDI, L.; DOOLAN, M.; GLASSEY, R. **ALF: a client side logger and server for capturing user interactions in web applications**. Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. Anais: SIGIR'12. New York, USA: ACM, 2012. Disponível em: <<http://doi.acm.org/10.1145/2348283.2348428>>
- BATEMAN, S.; GUTWIN, C.; OSGOOD, N.; MCCALLA, G. **Interactive usability instrumentation**. Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems. Anais: EICS '09. New York, NY, USA: ACM, 2009. Disponível em: <<http://doi.acm.org/10.1145/1570433.1570443>>
- BURGER, M.; ZELLER, A. **Minimizing reproduction of software failures**. Proceedings of the 2011 International Symposium on Software Testing and Analysis. Anais: ISSTA'11. New York, USA: ACM, 2011. Disponível em: <<http://doi.acm.org/10.1145/2001420.2001447>>
- CAMARGO, V. V.; MASIERO, P. C. **Desenvolvimento de Software Orientado a Aspectos**. In: Ildeberto Rodello, Kalinka Castelo Branco, José Remo. (Org.). II Escola Regional de Informática do Oeste Paulista. 1ed. Marília - SP: Omnigráfica, 2006, p. 138-164.
- COULTER, R.; GUTIÉRREZ F.; PIERCE J.; VERGARA V.M.; SALAND L.; CAUDELL TP.; GOLDSMITH TE.; ALVERSON D.C. **The effect of degree of immersion upon learning performance in virtual reality simulations for medical education**. Medicine Meets Virtual Reality 15: In Vivo, in Vitro, in Silico: Designing the Next in Medicine, v. 125, p. 155, 2007.
- DOLUNAY, B.; AKGUNDUZ, A. **Automated end-user behaviour assessment tool for remote product and system testing**. Expert Syst. Appl., v. 34, n. 4, p. 2511–2523, maio. 2008.
- FENSTERMACHER, K. D.; GINSBURG, M. A Lightweight Framework for Cross-Application User Monitoring. **Computer**, v. 35, n. 3, p. 51–59, mar. 2002.
- HERBOLD, S.; GRABOWSKI, J.; WAACK, S.; & BÜNTING, U. **Improved Bug Reporting and Reproduction through Non-intrusive GUI Usage Monitoring and Automated Replaying**. Proceedings of the 2011 IEEE Fourth International Conference on Software Testing, Verification and Validation Workshops. Anais: ICSTW'11. Washington, DC, USA: IEEE Computer Society, 2011. Disponível em: <<http://dx.doi.org/10.1109/ICSTW.2011.66>>
- HERNANDEZ, J.; BALDIRIS, S.; SANTOS, O. C.; FABREGAT, R.; & BOTICARIO, J. G. **Conditional IMS Learning Design Generation Using User Modeling and Planning Techniques**. Proceedings of the 2009 Ninth IEEE International Conference on Advanced Learning Technologies. Anais: ICALT '09. Washington, DC, USA: IEEE Computer Society, 2009. Disponível em: <<http://dx.doi.org/10.1109/ICALT.2009.185>>

- HORNBY, P.; CLEGG, C. W.; ROBSON, J. I.; MACLAREN, C. R. R.; RICHARDSON, S. C. S.; & O'BRIEN, P. **Human and organizational issues in information systems development.** Behaviour & Information Technology, v. 11, n. 3, p. 160–174, 1992.
- IMSAND, E. S.; HAMILTON, JR., J. A. **Impact of daily computer usage on GUI usage analysis.** Proceedings of the 4th annual conference on Information security curriculum development. **Anais:** InfoSecCD '07. New York, USA: ACM, 2007. Disponível em: <<http://doi.acm.org/10.1145/1409908.1409941>>
- KITCHENHAM, B. **Procedures for performing systematic reviews.** Keele, UK, Keele University, v. 33, p. 2004, 2004.
- MASUHARA, H.; KAWAUCHI, K. **Dataflow pointcut in aspect-oriented programming.** Programming Languages and Systems, p. 105–121, 2003.
- MENDES, J. L. S.; CARVALHO, J. V.; CARVALHO, C. V. A. **"CONSTRUFIG3D: Uma Ferramenta Computacional para apoio ao ensino da Geometria Plana e Espacial".** RENOTE - Revista Novas Tecnologias na Educação 5.1, 2007.
- MOORE, R. J.; CHURCHILL, E. F.; KANTAMNENI, R. G. P. **Three sequential positions of query repair in interactions with internet search engines.** Proceedings of the ACM 2011 conference on Computer supported cooperative work. **Anais:** CSCW'11. New York, USA: ACM, 2011. Disponível em: <<http://doi.acm.org/10.1145/1958824.1958889>>
- NAKAMURA, T.; IGARASHI, T. **An application-independent system for visualizing user operation history.** Proceedings of the 21st annual ACM symposium on User interface software and technology. **Anais:** UIST'08. New York, USA: ACM, 2008. Disponível em: <<http://doi.acm.org/10.1145/1449715.1449721>>
- NUNES, E. P. S.; NUNES, F. L. S.; TORI, R.; KENSKI, Vani M. **A Model of assessment of knowledge acquisition in Three-Dimensional Learning Virtual Environments.** In: XIV Symposium on Virtual and Augmented Reality, 2012, 2012, Niteroi. Proceedings of XIV Symposium on Virtual and Augmented Reality. Porto Alegre: SBC, 2012. p. 21-30.
- NUNES, E. P. S.; NUNES, F. L. S.; TORI, R. **Avaliação da Aquisição de Conhecimento em Ambientes Virtuais de Aprendizagem 3D para Treinamento Médico.** In: XIII Workshop de Informática Médica - XXXI Congresso da Sociedade Brasileira de Computação 2011, 2011, Natal. **Anais do XIII Workshop de Informática Médica - XXXI Congresso da Sociedade Brasileira de Computação 2011.** Porto Alegre: SBC, 2011. v. 1. p. 1904-1913.
- RAMSAY, N.; MARSHALL, S.; POTANIN, A. **Annotating UI architecture with actual use** Proceedings of the ninth conference on Australasian user interface - Volume 76. **Anais:** AUIC '08. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2008. Disponível em: <<http://dl.acm.org/citation.cfm?id=1378337.1378351>>
- SOLER, L.; NICOLAU, S.; FASQUEL, J. B.; AGNUS, V.; CHARNOZ, A.; HOSTETTLER, A.; & MARESCAUX, J. **Virtual reality and augmented reality applied to laparoscopic and notes procedures Biomedical Imaging: From Nano to Macro,** 2008. ISBI 2008. 5th IEEE International Symposium on **Anais,** maio, 2008.
- STEVEN, J.; CHANDRA, P.; FLECK, B.; & PODGURSKI, A. **jRapture: A Capture/Replay tool for observation-based testing.** SIGSOFT Softw. Eng. Notes, v.

25, n. 5, p. 158–167, ago. 2000.

- TORI, R.; NUNES, F. L. S.; GOMES, V. H. P.; TOKUNAGA, Daniel Makoto. **VIDA: Atlas Anatômico 3D Interativo para Treinamento a Distância**. In: WIE 2009 - XV Workshop Sobre Informática na Escola, 2009, Bento Gonçalves. Anais do XXIX Congresso da Sociedade Brasileira de Computação. Porto Alegre: Sociedade Brasileira de Computação, 2009. p. 1-10.
- TORO-TROCONIS, M.; MEERAN, K.; HIGHAM, J.; MELLSTRÖM, U.; & PARTRIDGE, M. **Design and Delivery of Game-Based Learning for Virtual Patients in Second Life: Initial Findings**. In: PEACHEY, A. *et al.* (Eds.). *Researching Learning in Virtual Worlds*. Human-Computer Interaction Series. [s.l.] Springer London, 2010. p. 111–138.
- VASCONCELOS, L. G. DE; BALDOCHI, JR., L. A. **Towards an automatic evaluation of web applications**. Proceedings of the 27th Annual ACM Symposium on Applied Computing. Anais: SAC'12. New York, USA: ACM, 2012. Disponível em: <<http://doi.acm.org/10.1145/2245276.2245410>>