



UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE POS-GRADUAÇÃO

Abordagem Visual Para Aprendizagem Colaborativa de Programação

Nilmara da Silva Salgado

Manaus
Agosto 2014

Nilmara da Silva Salgado

Abordagem Visual Para Aprendizagem Colaborativa de Programação

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Informática (PPGI) da Universidade Federal do Amazonas, como requisito para obtenção do título de Mestre em Informática.

Orientadora:

Professora Dra. Thaís Helena Chaves de Castro

UNIVERSIDADE FEDERAL DO AMAZONAS – UFAM

Instituto de Computação - IComp

Manaus
Agosto 2014

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

S164a Salgado, Nilmara
Abordagem Visual Para Aprendizagem Colaborativa de
Programação / Nilmara Salgado. 2014
115 f.: il.; 29 cm.

Orientadora: Thaís Helena Chaves de Castro
Dissertação (Mestrado em Informática) - Universidade Federal do
Amazonas.

1. Aprendizagem de Programação. 2. Colaboração. 3. Linguagem
de Programação. 4. Mundos Virtuais. I. Castro, Thaís Helena
Chaves de II. Universidade Federal do Amazonas III. Título

Dissertação de Mestrado sob o título “Abordagem Visual Para Aprendizagem Colaborativa de Programação”, defendida por Nilmara da Silva Salgado e aprovada em 04 de 08 de 2014, em Manaus, Estado do Amazonas, pela banca examinadora constituída por:

Profa. Dra. Thaís Helena Chaves de Castro
Universidade Federal do Amazonas
Orientadora

Prof. Dr. Alberto Nogueira de Castro Junior
Universidade Federal do Amazonas

Prof. Dr. Bruno Freitas Gadelha
Universidade Federal do Amazonas

Prof. Dr. Crediné Silva de Menezes
Universidade Federal do Espírito Santo

Profa. Dra. Elaine Harada Teixeira de Oliveira
Universidade Federal do Amazonas

Para minha mana pelo apoio incondicional. Nós duas sabemos como o caminho foi trilhado até aqui. Sinta-se parte dessa conquista, afinal, sua torcida sempre foi positiva, principalmente nos momentos de dificuldade.

Agradecimentos

À Deus, pelo amor imerecido.

À minha orientadora pela confiança, pelo apoio e por todas as diretrizes que proporcionaram reflexões para aprimorar as ideias e o projeto.

Ao prof. Alberto Castro, pelo apoio e conselhos durante o Estudo Piloto e por todas as revisões e orientações nos artigos submetidos.

Aos colegas do PPGI pelo companheirismo. Não citarei nomes, mas todo aquele que é chamado “amiguinho” ou “amiguinha” está incluso nessa lista.

Aos professores e funcionários do PPGI que incentivaram meu trabalho na Instituição.

À minha grande família, onde encontro refúgio.

Resumo

As estruturas básicas de algoritmos são fundamentais para o desenvolvimento de sistemas e entendê-las é um passo essencial no processo de aprendizagem, pois para se tornar um bom programador é necessário um conjunto de conhecimentos que vai além de conhecer a sintaxe e a semântica de uma linguagem de programação. Há vários métodos de aprendizagem de programação, como por exemplo, experimentos realizados em laboratórios e programação em pares. Porém, é necessário considerar formas de integração desses métodos que potencializem o aprendizado. Este trabalho apresenta uma abordagem diferenciada para a aprendizagem colaborativa de programação visual e sua conexão a um ambiente imersivo 3D onde os objetos, atores e comportamentos são resultantes dos programas desenvolvidos pelos estudantes organizados em times. É relatado um estudo piloto desenvolvido em uma disciplina de introdução à programação ofertada em uma Instituição Federal de Ensino Superior, utilizando-se como estratégia central de colaboração as etapas iniciais de um esquema progressivo para aprendizagem de programação em grupo. Os resultados sugerem um impacto positivo especialmente no que diz respeito à motivação, visualização das produções e da interação entre os estudantes.

Palavras-chave

Aprendizagem de Programação, Cenários de colaboração, Linguagem de Programação, Mundos Virtuais

Abstract

The basic structures of algorithms are fundamental for the development of systems and understand them is an essential step in the learning process, because to become a good programmer you need a body of knowledge that goes beyond the syntax and semantics of a language programming. There are several methods of learning programming, such as the experiments performed in laboratories and pair programming. However, it is necessary to consider how to integrate these methods that enhance the learning. This paper presents an unconventional approach to collaborative learning of visual programming and its connection to a 3D immersive environment where objects, actors and behaviors are the result of programs developed by team-organized students. We report a pilot study in an introductory programming course offered in a Federal Institution of Higher Education, using as a central strategy for collaboration the initial stages of a progressive scheme for group programming learning. Results suggest a positive impact especially with regard to motivation, visualization of production and interaction among students.

Keywords

Learning Programming, Collaborative Scenarios, Programming Language, Virtual Worlds

Sumário

| | | |
|-----|---------------------------------------------------------------------|-----|
| 1 | Introdução..... | 13 |
| 1.1 | Contexto..... | 14 |
| 1.2 | Objetivo..... | 19 |
| 1.3 | Problematização | 19 |
| 1.4 | Justificativa | 21 |
| 1.5 | Metodologia | 21 |
| 1.6 | Organização do Trabalho..... | 22 |
| 2 | Análise sobre Aprendizagem de Programação | 24 |
| 2.1 | Teorias de Aprendizagem | 24 |
| 2.2 | Métodos para Aprendizagem de Programação..... | 30 |
| 2.3 | Método de Aprendizagem Colaborativa de Programação..... | 34 |
| 3 | Tecnologia para Aprendizagem de Programação em Grupo..... | 36 |
| 3.1 | Ferramentas para Apoio à Aprendizagem de Programação | 36 |
| 3.2 | Ambientes Virtuais de Aprendizagem de Programação..... | 39 |
| 3.3 | Ambientes Colaborativos de Aprendizagem | 43 |
| 4 | Abordagem Visual para Aprendizagem Colaborativa de Programação..... | 45 |
| 4.1 | Definição da Abordagem..... | 45 |
| 4.2 | Estudo Piloto..... | 47 |
| 4.3 | Análise dos Resultados..... | 54 |
| 4.4 | Follow UP..... | 68 |
| 5 | Conclusão..... | 103 |
| 5.1 | Contribuição..... | 104 |
| 5.2 | Trabalhos Futuros..... | 105 |
| 5.3 | Publicações de Resultados Parciais | 107 |
| | Referências Bibliográficas..... | 108 |
| | APENDICE A: Questionário de Perfil..... | 113 |
| | APENDICE B: Questionário sobre Ambiente 3D..... | 114 |
| | APENDICE C: Questionário sobre a utilização do Ambiente 3D | 115 |

Lista de figuras

| | |
|--------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Figura 1. Esquema progressivo para Aprendizagem de Programação em Grupo | 48 |
| Figura 2. Esquema do planejamento das atividades práticas | 49 |
| Figura 3. Versões dos grupos para o objeto ventilador..... | 54 |
| Figura 4. Questionário sobre conhecimento prévio 3D. | 55 |
| Figura 5. Avaliação final referente ao Projeto Colaborativo | 56 |
| Figura 6. Levantamento de Perfil | 74 |
| Figura 7. Questionário de Avaliação de Conhecimento | 75 |
| Figura 8. Instantâneo de Prototipagem Manaus Transito | 77 |
| Figura 9. Telas do Desenvolvimento no Scratch - Trânsito Manaus | 78 |
| Figura 10. Instantâneo do Trecho do relatório da dupla “Trânsito Manaus”..... | 79 |
| Figura 11. Instantâneo da Prototipagem da aplicação Lazy Foot | 79 |
| Figura 12. Telas da aplicação <i>Lazy Foot</i> no Scratch | 80 |
| Figura 13. Instantâneo do Trecho do relatório da dupla Lazy Foot..... | 80 |
| Figura 14. Tela de abertura do jogo Caça ao Tesouro..... | 82 |
| Figura 15. Tela de distribuição de peças do jogo..... | 82 |
| Figura 16. Tela com desafio final | 83 |
| Figura 17. Levantamento de Perfil | 87 |
| Figura 18. Avaliação do Conhecimento | 88 |
| Figura 19. Comparativo de Perspectivas | 89 |
| Figura 20. Gráfico de Avaliação do Grupo Motivação | 90 |
| Figura 21. Gráfico de Avaliação do Grupo Experiência do Usuário | 91 |
| Figura 22. Gráfico de Avaliação do Grupo Aprendizagem | 92 |
| Figura 23. Levantamento de Perfil dos Participantes | 97 |
| Figura 24. Avaliação de Conhecimento dos Participantes | 97 |
| Figura 25. Instantâneo de Respostas de participantes cuja preferência é trabalhar sozinho - Questionário Pós-Jogo Turma de Matemática (T1)..... | 98 |
| Figura 26. Instantâneo de Respostas de participantes cuja preferência é trabalho em grupo - Questionário Pós-Jogo Turma de Matemática (T1) | 99 |
| Figura 27. Grupo de Motivação..... | 100 |
| Figura 28. Grupo de Experiência do Usuário | 101 |
| Figura 29. Grupo de Aprendizagem | 102 |

Lista de Tabelas

| | |
|---------------------------------------------------------------------------------------|----|
| Tabela 1. Declarações sobre a não aprovação..... | 57 |
| Tabela 2. Resumo sobre pontos positivos e negativos mencionados pelos alunos. | 58 |
| Tabela 3. Diálogo grupo 01: “estimulando os colegas” | 59 |
| Tabela 4. Diálogo grupo 02: “assumindo a liderança” | 60 |
| Tabela 5. Diálogo grupo 04: “seleção tranquila” | 61 |
| Tabela 6. Diálogo grupo 05: “detalhamento cuidadoso” | 62 |
| Tabela 7. Diálogo grupo 06: “análise + crítica positiva” | 63 |
| Tabela 8. Diálogo grupo 07: “detalhamento e espontaneidade” | 64 |
| Tabela 9. Diálogo grupo 08: “justificativa” | 65 |
| Tabela 10. Tipos e exemplos de padrões de interação no Fórum | 66 |
| Tabela 11. Duplas X Proposta de Aplicação | 75 |
| Tabela 12. Objetivo do Estudo Exploratório segundo o paradigma GQM. | 84 |

Lista de Sigla

| Sigla ou Termo | Significado |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2D | Duas Dimensões |
| 3D | Três Dimensões |
| AVA | Ambiente Virtual de Aprendizagem |
| <i>brainstorming</i> | Tempestade de Ideias - Dinâmica de grupo |
| CC | Ciência da Computação |
| E/S | Entrada/saída (em inglês: <i>Input/output</i> , sigla I/O) é um termo utilizado quase que exclusivamente no ramo da computação (ou informática), indicando entrada de dados, bem como a sua saída. |
| EC | Engenharia da Computação |
| Flash | Ou <i>Adobe Flash</i> , é um <i>software</i> primariamente de gráfico vetorial - apesar de suportar imagens <i>bitmaps</i> vídeos - utilizado geralmente para a criação de animações interativas que funcionam embutidas num navegador <i>web</i> e também por meio de desktops, celulares, smartphones, tablets e televisores. |
| HTML | Abreviação para a expressão inglesa <i>HyperText Markup Language</i> , que significa Linguagem de Marcação de Hipertexto) |
| LMS | <i>Learning Management Systems</i> , conhecido como LMS, ou Sistema de Gestão da Aprendizagem (SGA), são softwares desenvolvidos sobre uma metodologia pedagógica para auxiliar a promoção de ensino e aprendizagem virtual ou semi-presencial. |
| LSL | <i>Linden Scripting Language</i> - Linguagem de programação orientada a eventos, utilizada no simulador Second Life. |
| MIT | Instituto de Tecnologia de Massachusetts |
| MOODLE | " <i>Modular Object-Oriented Dynamic Learning Environment</i> ", ambiente virtual, de apoio à aprendizagem. |
| OOHDM | Object Oriented Hypermedia Design Method, ou Metodologia de Projeto Hipermídia Orientado a Objetos |
| Opensource | Código aberto, criado pela <i>OSI (Open Source Initiative)</i> . |
| Viewer | Programa que permite ver um arquivo gravado num determinado formato. |
| VRML | <i>Virtual Reality Modeling Language</i> . Uma linguagem de descrição de cena para a criação de gráficos interativos tridimensionais da Web. |

1 Introdução

A compreensão dos elementos básicos da programação possui importância central para a formação do estudante de computação. Entretanto, dentre os muitos complicadores envolvidos no tema, a necessidade de operar com abstrações para conceber, desenvolver e testar artefatos que tornam explícitos os processos para a solução de problemas, torna-se um obstáculo para os iniciantes.

Além disso, o professor dos cursos introdutórios precisa, além de planejar seu curso e cuidar para que as atividades transcorram conforme planejado, ter a habilidade de modificar uma atividade ou intervir sempre que identificar alguma dificuldade que pareça intransponível para os grupos. Para identificar essas dificuldades durante a realização dos exercícios é desejável que haja monitoramento das atividades e um processo bem definido para a solução de exercícios.

Em vista disso, este trabalho apresenta a proposta de uma abordagem diferenciada para a aprendizagem colaborativa de programação visual e sua conexão a um ambiente imersivo 3D. Ambientes 3D têm sido usados em contextos educacionais, inclusive de aprendizagem de programação. Por isso, propomos a inclusão do método de resolução de problemas de Polya (Polya, 1995). Esse método pode ser adotado como um meio para organizar a resolução de desafios inerentemente colaborativos descritos nos próprios cenários propostos, utilizando-se como estratégia central de colaboração as etapas iniciais de um esquema progressivo para aprendizagem de programação em grupo. Essas atividades promovem, além da integração entre professor e alunos, habilidades de construção do raciocínio lógico, coordenando o processo de aprendizagem.

Essa tecnologia 3D amparada por estratégias pedagógicas bem planejadas podem permitir avanços no processo de ensino-aprendizagem, pois, conforme notamos em Esteves (2010), os ambientes virtuais podem disponibilizar uma importante estrutura de recursos educativos para suporte

aos professores e, conseqüentemente, influenciam o comportamento dos alunos e apoiam a construção de conhecimento.

1.1 Contexto

Com uma revisão de literatura observamos uma grande variedade de trabalhos que propõem melhorias para o ensino da lógica de programação. Alguns desses trabalhos já adotando ambientes virtuais imersivos como ferramenta de apoio. Essas pesquisas buscaram métodos para sanar os principais problemas nessa fase inicial da disciplina. Os problemas mais citados nos trabalhos a seguir são: dificuldade de abstração, dificuldade na interpretação de enunciados de problemas, raciocínio lógico fraco e prática pedagógica desmotivadora.

Em seus estudos Castro *et al.* (2002) observou que a aprendizagem de conceitos que envolvem a construção de programas de computador não é trivial, pois requer o uso de habilidades de alto nível e bom raciocínio abstrato. Mesmo com o foco do trabalho em educandos que estão na fase inicial da programação, os autores advertem que os alunos desconhecem técnicas para entender e solucionar problemas. Por isso, em sua pesquisa procuraram desenvolver uma arquitetura que utilizasse padrões de erros cometidos na elaboração de soluções de problemas que contribuísse para o aprendizado de conceitos básicos de programação.

Em outra oportunidade de pesquisa Castro *et al.* (2004) citou que um outro aspecto a considerar é a habilidade de raciocínio lógico, que ajuda o estudante a enxergar melhores caminhos para chegar às soluções dos problemas destacando o método de resolução de problemas apresentado por *Polya* (1995), no livro “A arte de resolver problemas”. Segundo os autores, nesse método, a ideia geral para se resolver um problema é primeiramente distanciar-se dele, e só então tentar dividi-lo em partes menores, relacionando-as a outros problemas já resolvidos. Os autores reforçam que

esse distanciamento requer certa prática e sua experiência tem mostrado que os estudantes que ingressam nas universidades não estão acostumados a pensar dessa maneira por não terem o hábito de sistematizar seus próprios métodos de resolução de problemas.

Em seu estudo, (Falckembach e Araújo 2005) abordam as dificuldades encontradas pelos alunos na resolução de problemas. A dificuldade em reconhecer os procedimentos necessários para se chegar à solução do problema muitas vezes geram resistência. Os autores citam que essa resistência é, muitas vezes, estimulada pela metodologia usada no ensino do conteúdo, que é ministrada de forma coletiva. Isso porque realizar um acompanhamento individual e personalizado em uma sala de aula é difícil para um professor. Por isso houve a indicação em se realizar um levantamento de estratégias cognitivas de resolução de problemas utilizadas pelos alunos do curso de Sistemas de Informação, sendo escolhida a Estratégia Ascendente de Resolução de Problemas.

O resultado dessa pesquisa foi a criação de um ambiente de aprendizagem hipermídia obedecendo a Estratégia Ascendente de Resolução de Problemas, ou seja, resolver os algoritmos partindo do todo, e passo a passo, determinar os procedimentos para chegar ao resultado. Por exemplo, os alunos podem visualizar o modelo da solução de um problema, na forma gráfica, através dos fluxogramas ou através do algoritmo pseudocódigo (Falckembach e Araújo 2005).

Em suas pesquisas, (Rapkiewicz *et al* 2006) observaram a prioridade no desenvolvimento do raciocínio lógico do aluno, uma vez que este não consegue abstrair a lógica dos problemas e transformá-la numa sequência lógica para posteriormente desenvolver o algoritmo. Para (Rapkiewicz *et al* 2006) a utilização de recursos de jogos podem suprir essas necessidades, pois, além da finalidade de entretenimento, observaram que os jogos podem ser elaborados de acordo com problemas envolvendo lógica, ou seja, por meio do jogo o aluno pode conseguir abstrair a lógica de forma mais simbólica e visualizar sequências necessárias para a sua resolução.

Em (Castro *et al* 2008), seguindo a premissa que antes de aprenderem a programar os estudantes devem ser capazes de resolver problemas, é

potencializada a necessidade para aquisição de habilidades e estratégias para resolução de problemas em grupo, ou seja, indicando a colaboração entre as pessoas para promover o desenvolvimento dessas habilidades de forma cooperativa.

Já (Valentim, 2009) cita que é fundamental para o professor compreender como ocorre o processo ensino-aprendizagem. Este entendimento dará a ele suporte para a criação de novos meios (exercícios, formas de apresentar o conteúdo, como avaliar) que trarão melhorias à sua prática diária. Por isso, utilizou em seus estudos a abordagem de Campos (1972) do livro *Psicologia da Aprendizagem – teoria conexionista*.

Ainda em (Valentim, 2009) é citado que na teoria conexionista a aprendizagem baseia-se na Lei do Exercício, onde a conexão com o conhecimento fortalece quando há a prática e diminui quando não é exercitada. E, como o desenvolvimento do raciocínio é crucial para que os objetivos de ensinar e aprender sejam atingidos, é necessário despertar o interesse dos alunos através de uma abordagem motivacional, despertando a curiosidade dos alunos. No primeiro momento são propostos aos alunos problemas de lógica, os quais devem ser feitos sem o auxílio do professor. Raciocinar a partir de seu próprio conhecimento é o que se propõe ao aluno. Somente após este passo é que são apresentados aos alunos conceitos formais de algoritmos.

Pesquisas encontradas em Paula *et al.* (2009), consideram que um dos maiores fatores que impedem um rápido aprendizado em disciplinas que focam o desenvolvimento de Algoritmos, reside justamente na dificuldade de abstração. Na mesma pesquisa os autores consideram também que o hábito de leitura é fator que estimula a capacidade de abstração do aluno. Com base nessa análise, o estudo apresentou resultados preliminares sobre a importância da leitura e da representação prévia dos problemas (abstração) como motivadora da formação do raciocínio lógico-abstrato em cursos de Computação e Informática.

Segundo (Ferreira *et al.* 2009), aproximar o usuário cada vez mais desse ambiente de programação, sem que seja necessário aprender uma linguagem específica é uma solução muito favorável. Por isso, a equipe

realizou um estudo sobre a aprendizagem de lógica de programação utilizando Programação por Demonstração, técnica que tende aproximar o usuário do ambiente de programação sem a obrigação de aprender uma linguagem específica. A principal motivação deste estudo está no fato de que os problemas no processo de ensino e aprendizagem de algoritmos e programação estão relacionados ao entendimento das abstrações de lógica pelos estudantes. Sua aplicação permite que os usuários, por meio de um ambiente baseado em recursos visuais, aprendam a programar por meio de exemplos, especificando “o que” deverá ser feito, sem ter a preocupação de “como” será feito.

Dentre os recursos visuais temos os ambientes virtuais. No que diz respeito aos ambientes virtuais observamos que disponibilizam uma importante estrutura de soluções, pois permitem gerir recursos educativos que podem apoiar o processo de aprendizagem. Além de possuir uma interface mais agradável, seu emprego permite uma renovação nos métodos de ensino. A maneira de expor as informações pode ser constantemente trabalhada.

Com a tecnologia 3D surgiu os ambientes virtuais 3D ou mundos virtuais. Em Freire *et al.* (2008), nos deparamos que mundos virtuais podem ser definidos como uma metáfora computacional do mundo em que vivemos, com pessoas, lugares e objetos. Esses ambientes virtuais são iterativos capazes de simular algumas características do mundo real, tais como: gravidade, topografia, locomoção; bem como situações irreais como a capacidade de voar.

As pesquisas localizadas em Esteves (2010) analisam que a melhor forma de compreendermos mundos virtuais é através de um conjunto mínimo de características. Dentre elas podemos citar:

- Representação física – um corpo virtual designado de *avatar*, que é uma representação personalizada do ser humano. Sua funcionalidade consiste em permitir ao usuário interagir com o mundo virtual.
- Interatividade – permite uma interação entre vários usuários de forma simultânea, bem como a interação desses usuários com os objetos existentes no mundo virtual.

- Persistência – o mundo virtual continua a existir mesmo quando o usuário não está conectado e conseqüentemente não está participando.

De acordo com Esteves (2010), com relação ao acesso, podemos encontrar mundos com código-fonte aberto, mundos com acesso livre e mundos privados disponíveis apenas por pagamento de uma licença de utilização. A pesquisadora realizou uma investigação-ação de março de 2007 até julho de 2008 da disciplina de programação, sendo utilizado o mundo virtual Second Life e sua respectiva linguagem de programação, a *Linden Scripting Language* (LSL) para introdução ao ensino da programação.

No estudo de Morgado *et al.* (2008) também foi utilizado o Second Life e sua respectiva linguagem de programação no ensaio realizado em 2006-2007 com alunos das disciplinas Laboratório de Informática III e Laboratório de Tecnologias de Informação e Comunicação III.

Já em Freire *et al.* (2008) observamos a escolha do mundo virtual *Opensimulator*, por ser uma opção de código aberto permitindo que os usuários adaptem o ambiente de acordo com as suas necessidades. Dessa forma, estimulando a criatividade e o raciocínio dos alunos.

Em suma, todas as estratégias acima citadas prezam por um objetivo comum que é a motivação da aprendizagem de programação para alunos que estão iniciando nesse universo. Haja vista que todas elas buscaram recursos que trabalham com o raciocínio lógico, solução de problemas e soluções visuais, ao invés de dar prioridade à adoção de alguma linguagem de programação. Além disso, outra característica observada nos trabalhos foi o incentivo do trabalho em times.

As buscas nessas pesquisas nos levam a acreditar que é válido persistir em criar estratégias para dar aos alunos domínio sobre sua própria aprendizagem, principalmente quando há um esforço coletivo estruturado. A adoção desses recursos de aprendizagem utilizando estratégias colaborativas pode inspirar confiança nos alunos, aumentando o conhecimento, as habilidades e a segurança em resolver problemas.

1.2 Objetivo

Desenvolver nos alunos habilidades de resolução de problemas utilizando um ambiente visual em conexão com um ambiente 3D, empregando como estratégia central de colaboração as etapas iniciais de um esquema progressivo para aprendizagem de programação em grupo.

Para atingirmos o objetivo geral desse projeto será necessário focarmos os seguintes objetivos específicos:

- Estruturar a coleta de dados realizada junto aos coordenadores dos cursos de Engenharia da Computação e Ciência da Computação e comparar esses dados com questionários realizados posteriormente com alunos e professores que já participaram das referidas disciplinas que permitam analisar experiências de ensino-aprendizagem;
- Desenvolver objetos de um mundo virtual em atividades de estratégias de resolução de algoritmos utilizando o método de resolução de problemas do matemático Polya para uniformizar a disseminação do conhecimento;
- Aplicar uma sistematização específica para aprendizagem de programação em grupo;
- Desenvolver cenários em um mundo virtual, que tenham similaridades com estratégias de jogos para manter os alunos motivados com desafios contínuos;
- Aplicar e avaliar testes realizados com um grupo de estudo controlado.

1.3 Problematização

Aprendizagem de programação para alunos iniciantes é difícil pela necessidade de desenvolver raciocínio lógico, desenvolver processo de resoluções de problema e dificuldades em compreender abstrações. Observamos que os mesmos enfrentam essas dificuldades no que diz

respeito à assimilação do conteúdo das disciplinas introdutórias nos cursos de Ciência da Computação, Engenharia da Computação e áreas afins, mais especificamente no contexto relacionado a algoritmos. Essa problemática pode levar o aluno para um sentimento de frustração, e podem levar à desistência da disciplina (Castro *et al* 2008).

Além disso, a utilização de modelos tradicionais de aulas práticas dificulta o acompanhamento e a avaliação de todos os exercícios de todos os alunos em curto prazo. Em geral, os sistemas proveem um ambiente que permite ao aluno fazer seus algoritmos, porém, para o professor torna-se difícil fornecer um rápido *feedback* e disponibilizar as correções para cada aluno (Santos e Costa, 2006).

Em entrevistas realizadas com os coordenadores dos cursos de Ciência da Computação (CC) e Engenharia da Computação (EC) da Universidade Federal do Amazonas foi realizado um levantamento de notas e situação acadêmica dos últimos seis anos (2007- 2012). Com uma análise inicial foi observado a situação de reprovação de 41% dos alunos de Ciência da Computação e 40% dos alunos de Engenharia da Computação. Sendo que a reprovação por frequência nesses casos foi de 23% para o curso de CC e 25% para o curso de EC.

Assim, torna-se necessário adotarmos estratégias que melhor se enquadre nessa situação de forma a prover os requisitos necessários para motivar os alunos no processo de aprendizado, no sentido de que eles possam ser estimulados na comunicação e na busca pelo conhecimento. Para isso é necessário que exista um ambiente dotado de recursos em que o professor possa exercer a integração, o controle de atividades, bem como servir de mediador nesse novo espaço.

Então, questionamos: como podemos motivar os alunos para desenvolver atividades em um assunto que não é simples? Como intervir no processo de assimilação de habilidades dos alunos para resolução de problemas? Como mobilizar alunos e professores para incluírem esses recursos em suas práticas? E, como tornar todo esse sistema colaborativo?

1.4 Justificativa

A compreensão dos elementos básicos da programação possui importância central para a formação do estudante de computação. Entretanto, dentre os muitos complicadores envolvidos no tema, como desenvolvimento do raciocínio lógico, a necessidade de operar com abstrações para conceber; e o desenvolvimento e testes artefatos que tornam explícitos os processos para a solução de problemas, torna-se um obstáculo para os iniciantes.

Por outro lado, além de planejar seu curso e cuidar para que as atividades transcorram conforme planejado, o professor dos cursos introdutórios precisa ter a habilidade de modificar uma atividade ou intervir sempre que identificar alguma dificuldade que pareça intransponível para os grupos. Para identificar essas dificuldades durante a realização dos exercícios é desejável que haja monitoramento das atividades e um processo de solução de exercícios.

Essas características nos levam a um contexto colaborativo onde o uso de ferramentas e ambientes virtuais pode apoiar o ensino-aprendizagem de linguagem de programação nos direcionando ao estado interativo entre alunos e professores que se comunicam por meio destes ambientes.

1.5 Metodologia

Para elaboração da pesquisa do presente trabalho foram organizadas três etapas. Quanto à natureza da pesquisa, intrinsecamente qualitativa, inicialmente empregamos a Pesquisa Aplicada, seguida da Pesquisa Descritiva como comprovação para sua finalidade, conforme definições em (Moresi, 2003).

A aplicabilidade dos tipos de pesquisa foi constatada utilizando-se de forma prática instrumentos de coleta de dados como entrevistas realizadas com coordenadores, pelos quais tivemos acesso a situações de notas e frequências da disciplina Introdução à Computação dos últimos 6 anos na Universidade Federal do Amazonas. Esses dados forneceram subsídios para organizarmos o estudo piloto.

Os fatores listados a seguir nos levaram a escolha do Estudo de Caso Piloto (Yin, 2004). Tínhamos os participantes constantes no local de estudo, ser geograficamente conveniente, termos uma quantidade de dados e documentos necessários para o estudo, bem como por representar um caso real complexo, pois tínhamos à nossa disposição uma turma na qual inicialmente estavam matriculados 59 alunos, sendo que na sua maioria eram alunos que não conseguiram aprovação na primeira vez que cursaram a disciplina.

No decorrer do Estudo Piloto e após a sua aplicação, utilizamos questionários para checar os resultados encontrados com a abordagem proposta. Esses questionários forneceram dados para análise das ferramentas Scratch, Scratch for Opensimulator e Opensimulator; e do método adotado. Arquivos de *log* das atividades registrados durante o estudo também forneceram subsídios para verificação de padrões de interação.

1.6 Organização do Trabalho

Para a apresentação deste trabalho, além do presente capítulo, esta dissertação encontra-se organizada em quatro outros capítulos. Iniciamos com uma análise sobre Aprendizagem de Programação e para isso buscamos compreender as teorias basilares de aprendizagem, na sequência como alguns pesquisadores aplicaram essas teorias utilizando os métodos para

aprendizagem de programação em seguida focamos sobre Aprendizagem Colaborativa de Programação.

No capítulo sobre Tecnologia para Aprendizagem de Programação em Grupo descrevemos a pesquisa sobre as referidas tecnologias. Aqui destacamos ferramentas e ambientes virtuais criados com objetivo de apoiar esse processo de aprendizagem.

Uma das principais contribuições do presente trabalho é apresentada no capítulo sobre a Abordagem Visual para Aprendizagem Colaborativa de Programação. Esse capítulo constitui o projeto piloto, onde procuramos visualizar indícios para a abordagem proposta, apresentando as coletas de dados e análise desses dados.

E, o capítulo de conclusão onde descrevemos nossa argumentação sobre as ideias principais do trabalho. Além disso, destacamos os pontos positivos, situações de trabalhos futuros, bem como citamos as publicações de resultados parciais durante o desenvolvimento desse projeto.

2 Análise sobre Aprendizagem de Programação

O ato de aprender pode ocorrer em diversas situações. No entanto, o processo de aprendizagem não é algo simples. Se um conhecimento está sendo transmitido, no ponto de recepção ele deve ser compreendido. Essa ação pode abranger vários aspectos, como cognitivos e emocionais.

Em Polya (1995), nos deparamos com uma tradução do livro “Aprender, ensinar e aprender a ensinar” em que o autor cita: “Ensinar não é, em minha opinião, apenas um ramo da psicologia aplicada. Não o é em nenhum aspecto, pelo menos no presente. Ensinar está em correlação com aprender”.

Veremos a seguir que, a busca pelo aprimoramento desse processo gerou ao longo dos tempos o surgimento de várias teorias, como por exemplo, comportamentalismo e cognitivismo. Esta pesquisa de teorias permitiu o incremento de várias metodologias que tentam elucidar como o ser humano aprende e, principalmente, como deve ensinar.

2.1 Teorias de Aprendizagem

A aprendizagem envolve a vida do ser humano em vários aspectos, como aprendizagem cultural, aprendizagem cognitiva e aprendizagem emocional. Sendo assim, observamos que as teorias buscam reconhecer os aspectos que envolvem nos atos de ensinar e aprender, o elo entre o conhecimento que o ser humano já possui e o novo conhecimento.

Conforme descrito no dicionário¹, teoria pode ser um conjunto de conhecimentos que explicam certa ordem de fatos ou de princípios fundamentais de uma arte ou ciência. E, aprendizagem² diz respeito ao ato ou ao efeito de aprender, ou seja, ir adquirindo o conhecimento de algo. Com isso, podemos descrever que teoria de aprendizagem é um conjunto de

¹ <http://www.priberam.pt/dlpo/teoria>

² <http://www.priberam.pt/dlpo/aprendizagem>

conhecimentos que esclarecem a busca para explicar ordenadamente um determinado assunto com objetivo de aumentar o conhecimento nessa área.

Agora, vamos procurar trazer esse conceito para a aprendizagem de programação tratando de algumas teorias. É importante ressaltar que as teorias de aprendizagem não foram tratadas aqui em um contexto amplo, destacamos aqui somente algumas teorias basilares.

2.1.1 Teoria Comportamental

Esta teoria teve início em 1913 quando John Watson, conhecido como pai do Behaviorismo Metodológico, utilizou o termo para enfatizar sua preocupação com os aspectos observáveis do comportamento. O trabalho de Watson foi influenciado pelo fisiologista russo chamado Ivan Pavlov, que realizou estudos com animais em laboratório analisando estímulos que influenciavam no comportamento de cães (Ostermann e Cavalcanti, 2010). Ele escreveu um artigo que abriu o caminho para o Ensino Programado e as máquinas de ensinar: *The science of learning and the art of teaching*. John B. Watson, em 1913, publica um artigo intitulado *Psychology as the behaviorist views it*, que funda a corrente comportamentalista da psicologia (Miranda e Bahia, 2007). Em seu artigo ele cita: “O que nós precisamos fazer é começar a trabalhar sobre a psicologia, fazendo o comportamento, não a consciência, o ponto objetivo de nosso ataque” (Watson, 2008).

No entanto, Skinner (1974) citou que não havia nada de novo em tentar prever o comportamento por meio de observação e da manipulação de acontecimentos externos antecedentes. Porém poucos estudos sistemáticos foram feitos acerca do papel desempenhado pelo ambiente físico.

Com isso surgiu outra ramificação dessa teoria conhecida como Behaviorismo Radical. Segundo Skinner (1969) “nenhuma descrição do intercâmbio entre o organismo e meio ambiente estará completa enquanto não incluir a ação do ambiente sobre o organismo depois da emissão da resposta”. Ou seja, Behaviorismo Radical tem sua metodologia baseada na teoria do reforço, onde o aluno deve alcançar um objetivo em troca de

recompensas. Seu autor, Burrhus Frederic Skinner, cita que é possível programar o ensino de disciplina e comportamentos. Proposta essa baseada na estruturação de um plano que irá guiar os alunos para alcançar os objetivos. Nessa abordagem, se incluem tanto a aplicação da tecnologia educacional e estratégias de ensino quanto, formas de reforço no relacionamento professor-aluno (Miranda e Bahia, 2007).

Aos alunos caberia o controle do processo de aprendizagem, um controle científico da educação, o professor teria a responsabilidade de planejar e desenvolver o sistema de ensino-aprendizagem, de tal forma que o desempenho do aluno seja maximizado, considerando-se igualmente fatores tais como economia de tempo, esforços e custos. O estudo era individualizado, cada aluno trabalhava sozinho e procurava torná-los iguais, ou seja, com um mesmo nível de conhecimento (Ostermann e Cavalcanti, 2010).

No que diz respeito ao processo de avaliação, estava diretamente ligada aos objetivos estabelecidos pelo professor, sendo avaliado no decorrer do procedimento, um elemento que completava o processo. Decorrente da hipótese de que o aluno avança em seu ritmo próprio, em pequenos passos, sem cometer erros, a avaliação consiste, em se constatar se o aluno aprendeu e atingiu os objetivos propostos quando o programa foi conduzido até o final de forma adequada (Miranda e Bahia, 2007).

Para aprendizagem de programação poderíamos discutir aqui a necessidade de seguir um padrão de etapas seguindo essa abordagem comportamentalista, isto é, fazer com que o aluno siga passos previamente estabelecidos pelo professor. Entretanto, a aprendizagem de programação visa a construção do conhecimento através da interação do aluno com o ambiente, que irá oferecer um conjunto de elementos que sofrerão alteração durante seu ciclo. Portanto, nesse caso não podemos reforçar o modelo behaviorista de aprendizagem em detrimento de teorias construtivistas.

2.1.2 Teoria Cognitiva

A corrente cognitivista enfatiza o processo de cognição, ou seja, da função da inteligência ao adquirir um conhecimento³, através do qual a pessoa atribui significados à realidade em que se encontra. Preocupa-se com o processo de compreensão, transformação, armazenamento e uso da informação envolvido na cognição e procura regularidades nesse processo mental. Os cognitivistas acreditam que o indivíduo já nasce com condições para se apropriar de conhecimentos, ou seja, o processo de aprendizagem é tarefa exclusiva do sujeito, o ambiente não participa do processo. Nesta corrente, situam-se autores como Brunner, Piaget, Ausubel, Novak e Kelly (Ostermann e Cavalcanti, 2010).

Tratando-se de Piaget, ele impulsionou a Teoria Cognitiva. O mesmo relata em seus estudos sobre o desenvolvimento cognitivo que o aprendizado é um processo gradual no qual a criança vai se capacitando em níveis cada vez mais complexos do conhecimento. No livro *A Epistemologia Genética*, Piaget assinala esses níveis de conhecimentos como os quatro períodos gerais de desenvolvimento cognitivo. São eles: sensório-motor, pré-operacional, operacional-concreto e operacional-formal. Segundo o autor, o crescimento cognitivo da criança se dá por meio de *assimilação* e *acomodação*. O sujeito constrói esquemas de assimilação mentais para abordar a realidade. Muitas vezes, os esquemas de ação da pessoa não conseguem assimilar determinada situação. Neste caso, a mente abandona essa situação ou se modifica. Quando a mente se modifica, ocorre o que o autor chama de *acomodação*. As acomodações levam à construção de novos esquemas de assimilação, originando, com isso, o desenvolvimento cognitivo. Somente há aprendizagem quando o esquema de assimilação sofre acomodação (Piaget, 1971).

Agora, quando falamos de aprendizagem no ensino superior observamos que está relacionada com a atividade de pesquisa tanto do aluno quanto do professor, ou seja, é necessário que o professor promova situações em que o aluno aprenda a buscar e analisar informações comparando com conhecimentos anteriores para chegar a suas próprias conclusões. Podemos resumir esses passos no ato de aprender a aprender (Libaneo, 2003).

³ <http://www.priberam.pt/dlpo/cognicao>

Todavia, não existe uma fórmula pronta para assistir o aluno a aprender. No que diz respeito à construção de programas, Castro (2011) afirma que não é um processo trivial, pois requer o uso de habilidades cognitivas de alto nível e um processo de raciocínio abstrato. Levando-se em conta que essas habilidades diferem de indivíduo para indivíduo, é indispensável o desenvolvimento de métodos pedagógicos para ensinar a pensar, isto porque a aprendizagem não é algo que acontece individualmente, existe um contexto social que precisa ser levado em consideração.

2.1.3 Teoria do Construtivismo

Na linha da teoria Cognitivista temos o Construtivismo. O termo construtivismo surge na Psicologia também com a obra de Jean Piaget, no contexto de sua investigação sobre os processos da inteligência, para indicar o papel ativo do sujeito na construção de suas estruturas cognitivas. Outros pesquisadores que também se destacam são Henri Wallon, L.S. Vigotsky, A. N. Leontiev, A. R. Luria e Emília Ferreiro (Castañon, 2005).

Notamos em Souza (2006) que o enfoque construtivista enfatiza a construção de novo conhecimento e maneiras de pensar mediante a exploração e a manipulação ativa de objetos e ideias, tanto abstratas como concretas, e explicam a aprendizagem através das trocas que o indivíduo realiza com o meio. Essa característica procura contextualizar e aproveitar os recursos tecnológicos para os processos de ensino e aprendizagem.

O construtivismo implica que o aluno participe ativamente do próprio aprendizado, mediante a experimentação, a pesquisa em grupo e o desenvolvimento do raciocínio, entre outras formas. Com isso, segundo Machado e Miranda (2006), um dos grandes desafios da pedagogia construtivista é fazer com que o aluno entenda o processo de aprendizagem e o papel do erro na construção do conhecimento. Em geral, o erro ocasiona um efeito negativo no aspecto afetivo do sujeito. O estudante quando não consegue alcançar um resultado esperado sente-se frustrado.

Nesse caso, na relação professor-aluno, o diálogo é fundamental, pois o professor assume o papel de transmissor do conhecimento e pode construir cenários que oportunizam certos tipos de problemas, pensar situações instigantes e organizar novas propostas e investigações. Esse processo faz com que o aluno seja acompanhado e a avaliação é percebida como um procedimento contínuo. Nesse sentido, o rendimento do aluno irá refletir o trabalho desenvolvido em classe pelo professor (Machado e Miranda, 2006).

2.1.4 Teoria do Construcionismo

Seymour Papert, pesquisador sul-africano, professor de matemática e de educação do Instituto de Tecnologia de Massachusetts (MIT), pioneiro no uso do computador na educação, fundamentando-se na teoria construtivista de Piaget, define o termo *construcionismo* para a construção do conhecimento pelo aluno por meio do computador. Em sua teoria, propõe-se a explicar as relações aprendiz-computador para produzir o máximo de aprendizagem com o mínimo de ensino. Isto, porém, não significa deixar que a criança aprenda sem qualquer auxílio (Altoé e Penati, 2005).

O construcionismo busca meios de aprendizagem que valorizem a construção das estruturas cognitivas do sujeito a partir de suas ações, apoiada em suas próprias construções de mundo. Nesse caso, o professor age como facilitador, mediador da aprendizagem do aluno, respeitando o ritmo e o estilo de cada um. Nesta abordagem, o aluno constrói o seu conhecimento sobre determinado assunto por meio da resolução de um problema ou desenvolvimento de um projeto significativo (do interesse do aluno) e contextualizado (vinculado à realidade do aluno), em um trabalho compartilhado e colaborativo (Altoé e Penati, 2005).

Apesar de esta teoria incluir tudo que está associada ao Construtivismo, eles se diferem, pois afirmar que aprendizagem construcionista acontece especialmente bem quando as pessoas estão

engajadas na construção de um produto, algo externo a elas. Por isso, Papert tenta direcionar o aluno para o computador, a fim de que ele construa seus conhecimentos utilizando-se do computador e de programas próprios. No Construcionismo a aprendizagem tem que ser construída e entendida pelo próprio aluno, também deve ser construída com outros alunos e deve haver a produção de artefatos, pois são estes artefatos que irão comprovar a aprendizagem.

2.2 Métodos para Aprendizagem de Programação

Quando falamos que tal Instituição adota um método ou que um professor utilizar uma teoria, é preciso distinguir as diferenças entre ambos. Existem teorias de aprendizagem que são hipóteses e modelos de como o ser humano aprende, e métodos são as maneiras de proceder em atividades de ensino, normalmente com base em alguma teoria. No nosso caso, teorias que foram citadas na seção 2.1.

Em (Escórcio, 2010) é descrito que, métodos do Grego **METHODOS**, **METO** = Para e **ODOS** = Caminho, é o caminho utilizado pelo formador para atingir o seu objetivo. Portanto, o método pedagógico implica a ordenação de meios em direção a um fim. Nesse caso, o que fazer para proporcionar a aprendizagem.

Existem diferentes formas de observar o indivíduo, o que resultou em várias teorias. Isso não é diferente para os métodos. Sabemos que não existe somente um método, pois há diferentes maneiras de agir quando trabalhamos com a aprendizagem. Vejamos a seguir alguns tipos, suas principais características, bem como vantagens e desvantagens.

a) MÉTODO AFIRMATIVO EXPOSITIVO

Em (Escórcio, 2010) é abordado que neste método as informações são estruturadas e transmitidas de acordo com o conteúdo programático e com os alunos. Há algumas vantagens na sua utilização, como a participação de um elevado número de alunos, tempo produtivo, permite a utilização de meios audiovisuais, redução de custos, adequação a conhecimentos teóricos, permitindo o ensino de grande variedade de raciocínios, conceitos e técnicas.

No entanto, a quantidade e o grau de dificuldade dos conhecimentos são impostos pelo professor, a probabilidade de se perder conteúdo é grande, desrespeita o ritmo individual do aluno, ausência de *feedback* e, a criatividade e a iniciativa do aluno pode ser comprometida pelo formalismo na relação professor-aluno.

Na aprendizagem de programação o uso desse método pode comprometer o andamento da turma, pois, por si só, a programação é uma disciplina que requer prática e somente aulas expositivas seriam insuficientes, levando assim a um baixo desenvolvimento do raciocínio lógico.

b) MÉTODO AFIRMATIVO DEMONSTRATIVO

Este método tem seu foco no ensino prático de uma tarefa. (Escórcio, 2010) declara que é ensinar alguém a executar uma tarefa prática sendo a demonstração o meio de transmissão, como por exemplo, experimentos em laboratórios.

Sua adoção permite a transmissão de conhecimentos teóricos e práticos, possibilita a participação dos alunos, há presença de feedback, exige maior atenção dos alunos, o ritmo individual de aprendizagem é respeitado, favorece a comunicação professor-aluno, presença de relação informal, favorece a capacidade de conceber o trabalho e permite a realização do trabalho em grupo. Em contra partida possui desvantagens como: número reduzido de alunos, grande exigência com relação à disponibilidade de tempo,

necessidade de material pedagógico peculiar, demanda elaboração pedagógica e obriga a uma preparação exaustiva por parte do professor.

Já temos uma melhoria na contribuição desse método, pois como estamos tratando de aprendizagem, a prática com exercícios é fundamental. Neste caso aplicam-se bem as sessões de laboratórios para desenvolvimentos de programa. Entretanto, como nossos times de trabalho são do primeiro período acadêmico temos uma desvantagem com relação ao tamanho das turmas. Por isso, o docente, além de acompanhar toda a turma em sala de aula, deve preparar um ambiente pedagógico que atenda toda demanda e que lhe dê suporte, e isso requer uma disponibilidade de tempo dobrada. Devemos levar em consideração também o fato de que um professor de disciplinas introdutórias normalmente não tem apenas uma única turma.

c) MÉTODO INTERROGATIVO

Conforme localizado em (Escórcio, 2010), é um sistema de ensino de pergunta-resposta cuja sequência de pergunta e resposta parte da mais simples para a mais complexa. Através das perguntas o professor leva o aluno à descoberta, estimulando um feedback constante.

A utilização desse método possui vantagens como: motivação pela descoberta, a sessão de perguntas pode ocorrer a qualquer momento, a programação é controlada, os alunos participam bem, contínuo feedback e, em casos de alunos distraídos ou inibidos podem ser utilizadas perguntas diretas. Entretanto, a iniciativa deve vir do aluno, contribui para acomodação do grupo, não permite a dinâmica do grupo, é necessário disponibilizar mais tempo, se utilizado de forma exclusiva pode ser cansativo, pode inibir e é de difícil aplicação.

Tratando de turmas de cursos introdutórios, onde os alunos na sua maioria são calouros, utilizar essa estratégia requer flexibilidade. Conforme já foi dito esse método não pode ser utilizado de forma exclusiva, mas os alunos

podem ser incentivados a experimentar os conceitos de algoritmos de uma maneira prática e agradável por meio de desafios para desenvolver habilidades de resolução de programas. Além disso, esses desafios podem ser colocados em forma de estratégias de jogos e podem ser aplicados em grupos, o que pode contribuir para integração e interação da turma.

d) MÉTODO ATIVO

Permite trazer para a formação a experiência pessoal dos alunos, este aprende melhor se sentir implicado na ação. O centro de interesse é o aluno e o professor tem o papel de facilitador das relações. Há várias técnicas de método ativo, dentre elas podemos citar o *brainstorming*, estudo de casos e trabalho em grupo.

Há diversas vantagens na utilização desse método, como o desenvolvimento harmonioso entre professor-aluno, oportunidade de participação de todos da turma, debates de ideias gerando maior participação. Porém, é difícil de aplicar a situações, toda iniciativa é do professor, requer mais tempo que o método expositivo, necessário número limitado de alunos, dificuldade de coordenação e condução da sessão por parte do formador e necessidade de mais tempo para apresentação de conteúdos (Escórcio, 2010).

Conforme comentado no método demonstrativo, como nossos times de trabalho são do primeiro período acadêmico temos uma desvantagem com o tamanho das turmas, é realmente necessário o professor ter um sistema de acompanhamento que lhe dê suporte. Além disso, as estratégias pedagógicas adotadas precisam estar relacionada ao desenvolvimento de algoritmos e programas, pois o foco do curso é prático.

Agora, em um contexto real, a solicitação de adoção desses métodos pode ser encontrada nas Diretrizes Curriculares dos cursos de Bacharelado em Ciência da Computação, Engenharia da Computação, Engenharia de Software e Sistemas de Informação e dos cursos de Licenciatura em

Computação (2003), a metodologia de ensino deve ser centrada no aluno como sujeito da aprendizagem e apoiada no professor como facilitador do processo de ensino-aprendizagem (Nunes, 2003).

Para isso, é ressaltado nas Diretrizes Curriculares que o professor deve encorajar a execução de trabalhos extraclasse como forma do aluno aprender a resolver problemas, aprender a aprender, tornar-se autônomo e criativo estimulando seu raciocínio. O professor deve mostrar, ainda, como decompor conteúdos teóricos em aplicações práticas, motivando trabalhos em equipe para estimular a colaboração e o desenvolvimento das capacidades de comunicação e de negociação, ou seja, os métodos de ensino adotados devem ampliar a visão do aluno para aprendizagem, e não somente para o ensino (Nunes, 2003).

Ainda em (Nunes, 2003), observamos como atividades complementares podem apoiar essa transmissão de conhecimento, servindo de estímulo na formação prática do aluno e, conseqüentemente, na sua formação profissional. Estas atividades podem ser oferecidas sob vários aspectos, dentre os quais podemos citar: cursos de capacitação profissional, cursos de extensão universitária e pesquisa de iniciação científica.

2.3 Método de Aprendizagem Colaborativa de Programação

Durante a explanação dos métodos observamos a necessidade por parte do docente em se dispor de práticas e técnicas que contribuam para o acompanhamento do conteúdo disciplinar e distribuição de atividades, bem como a integração da turma. Esse acompanhamento pode ser de caráter individual ou em grupo. Nesse caso, agregar à aprendizagem de programação estratégias para ao trabalho em grupo, pode favorecer a relação do estudante com o conhecimento, com o professor e com os colegas, pois a colaboração possibilita ao indivíduo alcançar patamares superiores no que diz

respeito à construção do conhecimento sobre o tema, dado as várias possibilidades de trocas de significados essenciais à aprendizagem.

Colaboração é primordialmente conceitualizada como o processo de construção de significado compartilhado. A construção de significado não é assumida como uma expressão de representações mentais dos participantes individuais, mas sim como uma realização da interação (Stahl *et al*, 2006).

Nesse ponto podemos destacar a importância de aproveitar o convívio entre os alunos para conseguirmos atingir propósitos educacionais, por meio da colaboração. Com colaboração podemos instituir atividades de grupo que ambicionam um objetivo em comum que pode se concretizar no consenso de troca de conhecimentos, no trabalho em conjunto e na presença da coordenação.

Colaborar já é uma necessidade tanto no mercado de trabalho quanto na educação. Mais especificamente em empresas desenvolvedoras de software, a demanda crescente por produtos e serviços leva a uma competição acentuada, fazendo com que essas empresas busquem concentrar suas atividades. Isto se reflete na formação de times de desenvolvimento e na necessidade de colaboração no time e entre times. Na educação não é diferente, quando os estudantes estão colaborando, eles podem ver o ponto de vista de seus pares para construir uma solução unificada para o problema em questão (Castro *et al*, 2008).

Ainda em (Castro *et al*, 2008) observamos que, em aprendizagem de programação é necessário se adotar um modelo ou esquema para sugerir o desenvolvimento de estratégias e o acompanhamento do desenvolvimento de programas em grupo. Tal modelo deve servir a ambos os domínios: resolução de problemas e colaboração.

3 Tecnologia para Aprendizagem de Programação em Grupo

A tecnologia deve ser usada, preferencialmente, para proporcionar aos alunos a oportunidade de interagir e trabalhar juntos em problemas e projetos significativos. Seu uso deve servir de auxílio ao estudo e facilitar a aprendizagem trazendo o conhecimento de forma estruturada. A adaptação e absorção da escolha tecnológica e do projeto didático além de facilitar a aquisição de conhecimento contribuem para aumentar a autoestima transformando tarefas complexas em algo dinâmico e positivo.

Como mediador, o professor tem papel expressivo. Ele tem a incumbência de buscar alternativas viáveis para fazer desaparecer o desinteresse dos alunos que estão com dificuldades e não querem se envolver nos projetos adotados durante a disciplina. Mas não podemos confundir a dificuldade de aprendizagem com a falta de vontade de realizar tarefas, uma vez que os problemas de aprendizagem podem ser causados por uma simples preferência por determinadas disciplinas ou assuntos.

Vale destacar que, nas diversas ferramentas e ambientes que serão apresentados na sessão 3.1, as dificuldades identificadas não se encontram na questão tecnológica, ou ainda de infraestrutura, mas sim, na adequação de professores e alunos que necessitam ser capacitados, pois não basta saber operar um computador e acessar Internet, é preciso manter a motivação dos alunos e principalmente adaptar-se a um sistema novo que funciona com outro paradigma. Por isso, essa mudança exige ainda, uma formação docente que ultrapasse a prática linear e passe a ter características mais interativas e colaborativas.

3.1 Ferramentas para Apoio à Aprendizagem de Programação

Existem diversos tipos de ferramentas para auxiliar na aprendizagem de programação. De modo geral, elas diferem quanto à aplicação, tipo de estratégia que utilizam e atividades suportadas. A seguir serão descritos alguns trabalhos que fizeram uso dessas ferramentas durante suas pesquisas. Para facilitar a compreensão desse conjunto classificamos essas ferramentas de acordo com o paradigma de programação adotado. Isso porque, conforme encontramos em (Sampaio e Maranhão, 2008), algumas linguagens possuem particularidades em comum e são ditas pertencerem a um mesmo paradigma cujo conceito faz referência às técnicas ou padrões amplamente utilizados pelos programadores no que diz respeito a como programar. A classificação de linguagens em paradigmas é uma consequência de decisões de projeto que impactam a forma na qual uma aplicação real é modelada do ponto de vista computacional (Sampaio e Maranhão, 2008). Dentre os paradigmas estabelecidos iremos comentar sobre: imperativo, lógico, funcional, orientado a objetos e orientado a eventos.

Na Programação Imperativa os programas estão no conceito de um estado, modelado por variáveis, e ações ou comandos que manipulam o estado. Esse paradigma também é denominado de procedural, por incluir subrotinas ou procedimentos como mecanismo de estruturação (Sampaio e Maranhão, 2008).

Em (Nobre e Menezes, 2002) observamos, que dentro do conjunto de ferramentas definidas e modeladas para auxílio dos estudantes, eles utilizaram a AIDE, Uma Ferramenta Automatizada para o Ensino de um Curso de Introdução a Programação e o PORTUGOL/PLUS, Uma Ferramenta de Apoio ao Ensino de Lógica de programação. Ambas trabalham com a linguagem Pascal que é baseada no paradigma imperativo. É interessante destacar que um dos tópicos apontados como futuras melhorias para o ambiente foi a animação de rastreios, o que nos leva a um argumento visual.

A Programação Orientada a Objetos pode ser entendida como, uma aplicação que é estruturada em módulos (classes) que agrupam um estado e operações (métodos) sobre este. Classes podem ser estendidas e/ou usadas como tipos (cujos elementos são objetos) (Sampaio e Maranhão, 2008). Como exemplo do uso desse paradigma podemos citar o trabalho de (Tonini,

2013). Esse autor utilizou o modelo OOHDM (*Object Oriented Hypermedia Design Model*), que fornece mecanismos para construir um ambiente hipermídia complexo, dividindo o processo em etapas, construindo e desenvolvendo o modelo em cada uma delas, aplicado no ensino da programação orientada a objetos, na Universidade Federal de Lavras cuja finalização foi à criação de um sistema *web* hipermídia educacional, voltado a auxiliar o processo de ensino-aprendizagem utilizando-se como referência a área de Programação Orientada a Objetos.

Já a Programação Funcional tratam de funções que descrevem uma relação explícita e precisa entre E/S. Possui um estilo declarativo trabalhando com avaliação de funções matemáticas. Em (Castro *et al*, 2004) encontramos o uso desse paradigma aplicado em um ambiente intitulado Aprende - Ambiente Cooperativo de Apoio à Aprendizagem de Programação. Dentre os motivos de sua escolha, os autores explanaram que paradigma funcional é baseado em conhecimento já familiar ao estudante desde o ensino médio (funções e mapeamentos entre domínios); permite concentrar a atenção na elaboração de soluções e nas descrições formais delas e o foco da atenção está em “o que fazer”, ao invés de “como fazer”.

No Paradigma de Programação em Lógica encontramos características como as relações entre E/S; um estilo declarativo, como no paradigma funcional Na prática, inclui características imperativas, por questão de eficiência (Sampaio e Maranhão, 2008). Podemos encontrar sistemas especialistas e banco de dados, como em (Neto, 2007) que foi desenvolvido um AVA voltado para o ensino-aprendizagem de uma linguagem de programação específica. Esse ambiente fornece funcionalidades novas experimentais, como por exemplo, a possibilidade de utilização de um algoritmo inteligente para correção automática de respostas discursivas. Através dele, alunos podem submeter exercícios de programação através de uma interface *web* conectada ao interpretador Swi-Prolog, previamente instalado no servidor, entre outras atividades.

E o paradigma orientado a eventos, cuja facilidade de criação e manipulação de interfaces gráficas para programas contribui para interação do usuário com o programa. Atrelado a este paradigma temos uma

ferramenta visual criada pelo MIT chamada Scratch. Essa ferramenta utiliza a orientação a eventos com uma interface visual que torna a programação agradável e desafiadora para os alunos, conforme observamos em (Bini e Koscianski, 2009) onde a ferramenta Scratch foi intitulada de um ambiente criativo e motivador, procurando com isso trabalhar o processo de resolução de problemas. Um dos pontos positivos citados foi a facilidade na programação por reduzir a exigência de sintaxe.

Por fim, (Neto, 2013) apresenta o *Scratch* como um ambiente interativo e descontraído, proporcionando uma programação atraente, fácil e mais intuitiva para ser utilizado na disciplina de Lógica de Programação. Ele obteve essa confirmação a partir da aplicação por meio de um estudo de caso em turmas do curso Técnico de Informática em uma Instituição de Ensino. O autor cita que apesar da ferramenta apresentar baixo desempenho com muitos recursos, como por exemplo, colocar muitos recursos (imagens, sons, músicas) deixa o programa pesado e dependendo da capacidade da máquina utilizada, os comandos ficam lentos e acabam travando e utilizar muitos blocos de comando para realizar algo. Também, em alguns casos, dependendo do que se deseja fazer, necessita da junção de muitos blocos de comando, tornando a programação trabalhosa. Entretanto, a ferramenta auxiliou no processo de aprendizagem dos alunos nos conceitos iniciais da disciplina de Lógica de Programação trazendo resultados bastante favoráveis.

3.2 Ambientes Virtuais de Aprendizagem de Programação

Para melhor compreensão do contexto sobre um ambiente virtual, vamos iniciar com o conceito de realidade virtual. Segundo (Zhou e Deng, 2009), o termo realidade virtual foi proposto em 1989 por um cientista americano chamado Jaron Lanier Zepel, criador da empresa VPL Research, Inc, quando descreveu sobre a tecnologia de simulação de computador. Atualmente, essa

tecnologia está sendo empregada em diversas áreas, tais como: simulação de treinamentos, arquitetura e urbanismo, medicina, defesa, comunicação à distância, educação, treinamento e mundos virtuais.

Esses autores conceituam realidade virtual como sendo uma tecnologia de interação natural. Com esta definição o usuário pode entrar em ambiente em tempo real e pode aumentar a percepção de presença, ou seja, o usuário consegue criar uma realidade e o cérebro aciona a presença dos personagens ou objetos que o computador processa.

A partir desse contexto surgiram os ambientes virtuais, que são ambientes criados em computador que parecem reais, ou seja, são mostrados para o usuário formas, desenhos, objetos e situações reais e irreais em tempo real com o cérebro analisando a presença dos personagens ou objetos que o computador processa. (Zhou e Deng, 2009) e (Garcia *et al.* 2001) citam que esse processo considera algumas características: imersão, navegação e interação.

Ainda em (Zhou e Deng, 2009), é mencionado que os psicólogos sempre usam a palavra “imersão” para descrever a experiência única em que as pessoas estão completamente atraídos pela atividade e nela envolvidos, ou seja, os usuários tem a impressão de estar realmente interagindo com aquela realidade artificial (realidade virtual).

Para navegação, (Garcia *et al.* 2001) observam que é utilizado um painel para o usuário se locomover dentro do ambiente virtual. Porém, a pessoa somente “peregrina” por aquele programa sem modificá-lo de nenhuma maneira.

Já na interação, os autores ressaltam que há recursos que podem ser utilizados pelo usuário dentro do ambiente virtual para executar tarefas ou atividades que permitam o conhecimento do conteúdo disponibilizado dentro do ambiente. Isso quer dizer que na interação os usuários podem interagir com o objeto, podendo movimentá-lo ou alterá-los.

Cabe ressaltar que ambientes virtuais podem envolver características tridimensionais e bidimensionais. Uma considerável parte de ambientes virtuais de ensino a distância tem seu conteúdo disponibilizado através de ambientes bidimensionais, como as aplicações desenvolvidas para web. Já

ambientes tridimensionais permitem a visualização, manuseio e exploração de mundos virtuais (Alves et al, 2003).

Tratando-se de ambientes virtuais de ensino e aprendizagem, em (Haguenauer et al. 2009) encontramos que Ambiente Virtual de Aprendizagem (AVA) é um ambiente baseado em Internet, sendo possível também o uso em Intranet cujo objetivo fundamental é o Ensino Assistido por Computador ou Ensino Apoiado pelas Tecnologias da Informação e da Comunicação (NTIC).

Em (Haguenauer et al. 2009) também citam que os Ambientes Virtuais de Aprendizagem podem ser desenvolvidos por meio de Sistemas de Gerenciamento da Aprendizagem. No entanto, os AVA (online ou offline) podem também ser desenvolvidos a partir de diferentes softwares e ferramentas, como o Flash, o HTML, o VRML (*Virtual Reality Modelling Language*) dentre outros.

Existem diversas características e qualidades que um AVA deve apresentar. Algumas relativas à linguagem (tanto gráfica quanto textual), outras relativas à arquitetura da informação e à navegação e outras ainda, relativas ao grau de interação com o conteúdo e com os participantes. Algumas dessas características e qualidades são: navegação intuitiva, clareza e consistência de signos utilizados no projeto gráfico, projeto gráfico harmonioso e agradável, conteúdo adequado ao público alvo, linguagem direta e simples, possibilidade de autoria, dialogicidade, interatividade (com o conteúdo, com a equipe de ensino e com outros internautas) (Bento, 2011).

Com todas essas características embutidas proporcionando uma variedade de formas de aprendizagem fizeram surgir no meio acadêmico e comercial vários sistemas, dentre eles destacamos a plataforma *Modular Object-Oriented Dynamic Learning Environment* (Moodle), um ambiente open source, disponibilizado em 2001. Essa plataforma possui um grande número de ferramentas que permitem criar atividades de aprendizagem, tais como: lições interativas, portfólios, textos colaborativos, glossários, perguntas frequentes, canais de conversa, fórum de discussão, diários, questionários de avaliação, tarefas e exercícios (Bento, 2011).

Os ambientes virtuais disponibilizam uma importante estrutura de recursos, pois permitem gerir recursos educativos que podem apoiar o processo de aprendizagem. Além de possuir uma interface mais agradável, seu emprego permite uma renovação nos métodos de ensino. A maneira de expor as informações pode ser constantemente trabalhada.

Com a tecnologia 3D surgiu também os ambientes virtuais 3D ou mundos virtuais. Segundo (Freire *et al.* 2008), mundos virtuais podem ser definidos como uma metáfora computacional do mundo em que vivemos, com pessoas, lugares e objetos. Esses ambientes virtuais são iterativos capazes de simular algumas características do mundo real, tais como: gravidade, topografia, locomoção; bem como situações irreais como a capacidade de voar.

Com essas características os ambientes virtuais de aprendizagem têm sido utilizados nos mais diversos campos de desenvolvimento tecnológico, pois envolve um controle interativo com capacidade de manusear e conceber uma grande variedade de conteúdo. Essas características são ideais para aplicações colaborativas em disciplinas de iniciação a Ciência da Computação e áreas afins.

Observamos uma de suas aplicações de com o desenvolvimento de uma ferramenta computacional chamada MetaBot, desenvolvida com o propósito de auxiliar no processo de ensino-aprendizagem de lógica computacional. (Morgado *et al.* 2008) também utilizou o Second Life e sua respectiva linguagem de programação no ensaio realizado em 2006-2007 com alunos das disciplinas Laboratório de Informática III e Laboratório de Tecnologias de Informação e Comunicação III.

Já em (Freire *et al.* 2008) observamos a escolha do mundo virtual *Opensimulator*, por ser uma opção de código aberto permitindo que os usuários adaptem o ambiente de acordo com as suas necessidades. Dessa forma, estimulando a criatividade e o raciocínio dos alunos.

3.3 Ambientes Colaborativos de Aprendizagem

Os ambientes virtuais, sejam eles 2D ou 3D, possuem recursos onde suas tecnologias de aprendizagem não são apenas utilizadas em situações de ensino a distância, mas também para reforçar a aprendizagem colaborativa, onde a comunicação imita a interação presencial. Essa afirmação pode ser comprovada em vista das várias pesquisas que já foram realizadas com esse foco e continuam em andamento.

Ambientes virtuais de aprendizagem colaborativos procuram refletir as estruturas práticas do docente, desde o planejamento passando por processos de mediação da aprendizagem, até as atividades de avaliação e monitoramento. Isso pode ser visto no REDU – Rede Social Educativa é um ambiente virtual de ensino. Seus autores buscaram a junção da educação presencial e tradicional com a educação virtual com o propósito de que o ensino possa garantir novas formas de aprendizagem. A ideia é que em seu próprio ambiente, o aluno através do Redu poderá se tornar um sujeito ativo na construção do seu conhecimento, permitindo que o processo de aprendizagem se desenvolva não somente em sala de aula, mas também no mesmo ambiente em que ele trabalha e vive, pois ele poderá interagir com seus colegas ou professor dentro e fora da sala (Gomes *et al.* 2012).

Em (Nobre e Menezes, 2002) foi proposto um ambiente de suporte à cooperação e colaboração entre aluno-aluno e professor-aluno. Alguns aspectos quanto a metodologia adotada em sala de aula contribuíram para fundamentar o desenvolvimento de algumas ferramentas para o professor, bem como nos aspectos levantados junto aos alunos, no que diz respeito à prática da disciplina em laboratórios e a dificuldade encontrada para sanar dúvidas no caso de atividades extraclasse.

Quando se trata de estimular a cooperação e colaboração aluno-aluno, foi proposta por (Neves e Coelho, 2006) uma ferramenta com base em ontologias que auxiliasse a revisão entre os alunos de programas escritos na linguagem Java, chamada OntoRevPro. A ontologia proporcionará aos alunos

meios para entender o que deve ser revisado em um programa Java e ainda fornecerá explicações sobre cada termo utilizado.

No aspecto de motivação encontramos em (Silva *et al.* 2012) a proposta de uma plataforma de criação e execução de Mundos Virtuais 2D - o Clic&Ação baseada nas tecnologias e conceitos vigentes no momento, como é o caso das redes sociais. Sendo assim, as criações poderão ser compartilhadas e testadas conforme a disponibilidade dos jogadores. Nele é possível criar aventuras e compartilhá-las com os colegas, que podem então fazer alterações e jogá-las da maneira que quiserem, estimulando a criação e o aprendizado por meio do jogo.

Já (Esteves, 2010) realizou uma investigação-ação de março de 2007 até julho de 2008 da disciplina de programação, sendo utilizado o mundo virtual Second Life e sua respectiva linguagem de programação, a *Linden Scripting Language* (LSL) para introdução ao ensino colaborativo da programação.

E, em (Freire et al, 2008) encontramos uma análise do potencial dos mundos virtuais na educação a distância através do Opensimulator, incluindo a criação e o desenvolvimento de um ambiente virtual de ensino-aprendizagem. Isso porque esse mundo virtual, assim como muitas outras plataformas de mundos virtuais existentes, não possui ferramentas específicas para a educação, até mesmo por não ter sido criado com esse propósito. Assim, faz-se necessária a criação de espaços e ferramentas educativas que auxiliassem no processo de ensino-aprendizagem.

4 Abordagem Visual para Aprendizagem Colaborativa de Programação

Este capítulo apresenta uma abordagem diferenciada para a aprendizagem colaborativa de programação visual e sua conexão a um ambiente imersivo 3D onde os objetos, atores e comportamentos são resultantes dos programas desenvolvidos pelos estudantes organizados em times.

Aqui será relatado um estudo piloto desenvolvido em uma disciplina de introdução a programação oferecida na Universidade Federal do Amazonas, utilizando-se como estratégia central de colaboração as etapas iniciais de um esquema progressivo para aprendizagem de programação em grupo (Castro, 2011). Os resultados sugerem um impacto positivo especialmente no que diz respeito à motivação, visualização das produções e da interação entre os estudantes.

4.1 Definição da Abordagem

A compreensão dos elementos básicos da programação possui importância central para a formação do estudante de computação. Entretanto, dentre os muitos complicadores envolvidos no tema, a necessidade de operar com abstração para conceber, desenvolver e testar artefatos que tornam explícitos os processos para a solução de problemas, torna-se um sólido obstáculo para os iniciantes.

Nesse contexto, associar à aprendizagem de programação, estratégias para ao trabalho em grupo pode favorecer a relação do estudante com o conhecimento, com o professor e com os colegas. A colaboração possibilita ao indivíduo atingir patamares superiores no que diz respeito à construção do conhecimento sobre o tema, devido as várias possibilidades de trocas de significados essenciais à aprendizagem (Pimentel e Fuks, 2011). Em Castro (2011) é relatada uma investigação no tema que envolveu uma

sistematização específica (um esquema progressivo) para a aprendizagem de programação em grupo.

Tem sido cada vez mais frequente que a aprendizagem de programação, especialmente quando apoiada em abordagens colaborativas, faça uso de ambientes virtuais e outras ferramentas específicas que auxiliam a organização e registro das interações, metas, recursos e produtos desenvolvidos pelos indivíduos e grupos envolvidos. Um levantamento na literatura relacionada apresentado em (Aureliano e Tedesco, 2012) confirma a utilização desse tipo de sistema de apoio, onde se destacam os LMS (*Learning Management Systems*) dentre os mais usados, sendo o Moodle e suas extensões, um dos mais populares.

Várias abordagens para a aprendizagem de programação compartilham a perspectiva que programar equivale a desenvolver uma solução a um problema e construir um artefato que incorpore tal solução, possibilitando sua verificação. Em tais casos, as peculiaridades da linguagem de programação utilizada (e a complexidade associada) são tratadas em segundo plano. O uso dos ambientes para programação visual (Ribeiro *et al.*, 2012) estão associados a tais abordagens e dentre eles, o Scratch⁴ tem sido utilizado por iniciantes a partir dos 8 anos de idade em mais de 150 países, estando disponível em mais de 40 idiomas.

Há ainda relatos de pesquisa onde se buscou explorar os ambientes imersivos de realidade virtual como fator motivacional para iniciantes na aprendizagem de programação (Esteves, 2010), (Freire *et al.* 2008), (Morgado *et al.* 2008).

Portanto, relatamos uma investigação na aprendizagem de programação em grupo que adotou o esquema progressivo desenvolvido em (Castro, 2011) e utilizou o ambiente *Scratch* e uma extensão do *Moodle* como elementos para o desenvolvimento de projetos de programação numa abordagem progressiva na complexidade (i) dos problemas, (ii) da plataforma (2D para 3D), e (iii) das articulações (atuação individual para grupos).

⁴ Scratch (2007). About Scratch. Disponível em: <<http://scratch.mit.edu/>>. Acesso em: 01 dez. 2012

4.2 Estudo Piloto

Um estudo piloto é geralmente considerado para ser um ensaio de pequena escala dos processos propostos, materiais e métodos, que auxilia os pesquisadores na hora de aprimorar os planos para a coleta de dados tanto em relação ao conteúdo desses dados quanto às ações que devem ser seguidas (Yin, 2004). No projeto aqui relatado, esse estudo foi desenvolvido na disciplina Introdução a Computação, ofertada em 2012/2 para alunos que em sua grande maioria não conseguiram aprovação na oferta regular no período anterior.

A proposta do estudo piloto foi trabalhar segundo algumas das premissas centrais da disciplina – foco na resolução de problemas, menor ênfase dos detalhes da linguagem utilizada, abordagem orientada a projetos e colaboração baseada no trabalho em grupos.

O tema dos projetos e a plataforma para desenvolvimento dos projetos foram alterados para o desenvolvimento de jogos e a transposição de experimentos para ambiente de realidade virtual.

4.2.1 Design do Experimento

Para o estudo em questão utilizamos o esquema progressivo para aprendizagem de programação em grupo apresentado em (Castro, 2011). A Figura 1 ilustra o avanço da atividade de programação do individual para o grupo, em um contexto que inicia na Fase 1, com uma preparação que envolve sessões no laboratório tratando com problemas introdutórios e esclarecimentos sobre a metodologia. A Fase 2 consiste da solução individual com o respectivo registro.

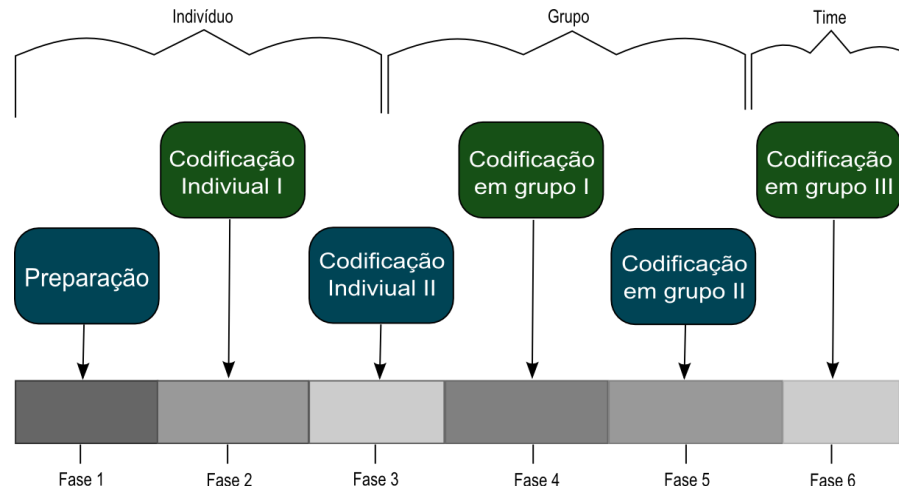


Figura 1. Esquema progressivo para Aprendizagem de Programação em Grupo
Fonte: (Castro, 2011)

Na Fase 3 o trabalho em grupo começa com a decisão sobre qual seria a melhor solução individual desenvolvida. Por conseguinte a solução do problema torna-se colaborativa, ou seja, é realizada a Codificação em Grupo I na Fase 4. Nessa fase o professor define as tarefas e o grupo define os atores para cada atividade. Já na Fase 5 o grupo também é responsável pela definição das tarefas. Por fim, na Fase 6 acontece uma atividade de desenvolvimento onde os grupos competem num cenário que reproduz situações reais de trabalho

Uma vez que adicionamos ferramentas de trabalho diferentes das descritas em (Castro, 2011), optamos por utilizar somente até a Fase 4 do esquema, pois nessa fase seria incluído o ambiente 3D e a necessária orientação de uso para a turma.

Castro, 2011 descreve o planejamento das atividades de acordo com as fases durante o período letivo.

1. Preparação – No início do período os alunos responderam um questionário para identificarmos o perfil da turma. Em paralelo foi apresentado no Plano de Ensino como seria o seguimento das aulas durante o período. Ou seja, da utilização da ferramenta *Scratch* para programação visual e durante o Projeto Cooperativo as ferramentas *Scratch for Opensim* para geração dos scripts e o *Opensimulator*.

2. Codificação Individual I – Sessões de laboratórios utilizando a ferramenta *Scratch*. Nessa fase os alunos trabalharam com os seguintes conceitos disciplina: variáveis, estrutura de repetição, definições condicionais, expressões lógicas e tipo lógico. As atividades eram apresentadas em forma de desafios.
3. Codificação Individual II – Sessões de laboratórios utilizando a ferramenta *Scratch*. Nessa fase os alunos trabalharam os conceitos listas, objetos, eventos e sincronismo. Os desafios nessa fase foram mais complexos, pois os alunos precisavam dos conceitos iniciais em paralelo aos conceitos dessa fase.
4. Codificação em Grupo I – Desenvolvimento do Projeto Colaborativo utilizando a ferramenta *Scratch for Opensim* em conjunto com o *Opensimulator*. Análise e seleção de soluções individuais com anotações colaborativas sobre o processo decisório foram realizadas nessa fase.

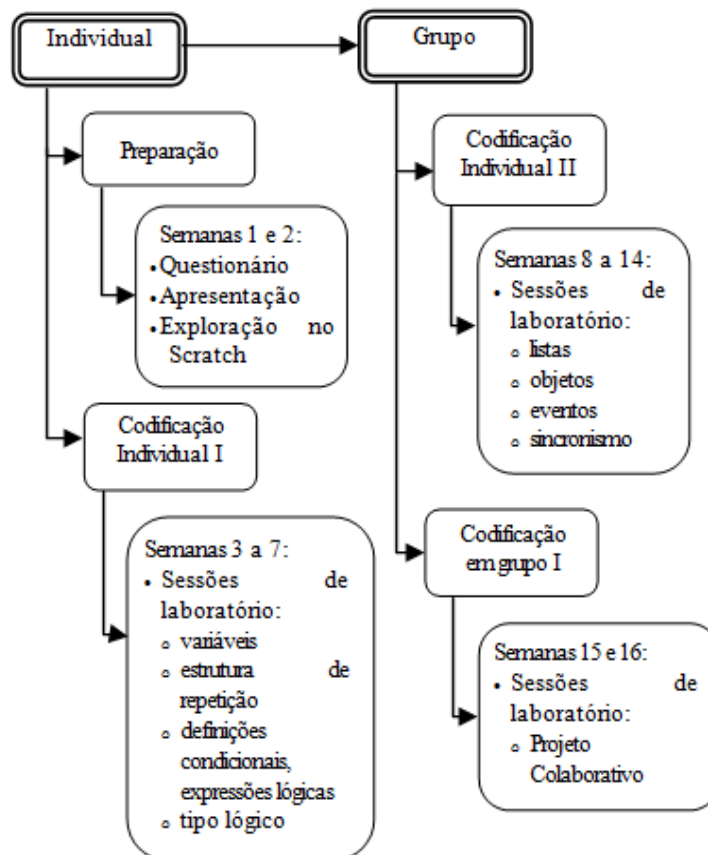


Figura 2. Esquema do planejamento das atividades práticas
 Fonte: Elaborado pelos autores, 2012.

4.2.2 Plataforma Utilizada no Estudo Piloto

Para acompanhar as atividades propostas para os alunos foi utilizado o ambiente *Moodle*⁵, além do *OpenSimulator* com uma 'viewer' chamada *Imprudence*, bem como as ferramentas de programação visual *Scratch* e *Scratch4Opensim*⁶.

A busca por facilitar o desenvolvimento do raciocínio lógico em cenário agradável ao trabalho conduziu à escolha do Scratch, um produto do grupo Lifelong Kindergarten no Media Lab do MIT, fornecida gratuitamente no site. Com o Scratch, você pode programar suas próprias histórias interativas, jogos e animações, bem como compartilhar suas criações com outros membros da comunidade online de forma criativa e sistemática.

Outra característica importante para o projeto era que a ferramenta gráfica utilizada fornecesse uma conexão de uso para o ambiente de realidade virtual selecionado – o *Opensimulator*⁷ ou simplesmente *OpenSim*. Trata-se de um servidor de aplicativo 3D multi-plataforma *open-source*. Ele pode ser usado para criar um ambiente virtual (ou mundo), que pode ser acessado através de uma variedade de clientes, em vários protocolos. Ele também tem uma instalação opcional (o Hypergrid) para permitir aos usuários visitar outras instalações *OpenSimulator* em toda a web a partir de uma conta em uma instalação 'home'.

O ambiente de integração entre o Scratch e o Opensim é o *Scratch4Opensim*, foi desenvolvido originalmente para o ambiente *Second Life*TM – a versão proprietária do *OpenSim*. A estrutura das ferramentas é bem semelhante, o que torna a passagem de uso de uma para outra relativamente fácil.

O ColabWeb, um ambiente virtual de apoio à aprendizagem desenvolvido na instituição a partir do Moodle, foi utilizado para estruturarmos a comunicação dos grupos e a organização dos projetos. Dessa forma,

⁵ Moodle.org

⁶ <http://www.greenbushlabs.com/LabsBlog/?p=797>

⁷ Opensimulator (2011). What is Opensimulator? Disponível em:<http://opensimulator.org/wiki/Main_Page>. Acesso em: 01 nov. 2012

poderíamos registrar os resultados de suas atividades, bem como suas dúvidas e opiniões.

E, para visualização do mundo virtual, dentre algumas *viewers* possíveis, selecionamos a *Imprudence*. Também é um projeto de código aberto baseado na *viewer* do *Second Life*. Após a instalação e configuração do *OpenSim* o *Imprudence* deve ser ativado.

4.2.3 Procedimentos

A descrição do procedimento segue o planejamento das atividades descritas na seção a. Conforme apresentado no esquema de atividades, durante a Fase de Preparação os alunos responderam a um questionário com 10 interrogativas sobre o perfil da turma. Nessa fase foram expostos aos alunos como seria o período letivo, apresentação do *Scratch* e a devida explicação de que a mesma seria a ferramenta adotada para as atividades, pois iríamos trabalhar o conteúdo didático da disciplina com programação visual, dando ênfase ao raciocínio lógico. Foi esclarecido também que a próxima ferramenta a ser utilizada era uma sequência do *Scratch*, o *Scratch for Opensim*, pois a mesma tem os recursos necessários para converter os scripts para o ambiente 3D.

Na Fase 2, onde houve a codificação individual I, atividades foram realizadas em forma de desafios abordando os conceitos de variáveis, estruturas de repetição, definições condicionais, expressões lógicas e tipo lógico. Ao todo, 11 desafios foram realizados nesse tempo. Os primeiros desafios de menor complexidade para ambientação da turma com relação a ferramenta. Em todos os desafios foi observada uma integração natural da turma para resolução dos mesmos. Eles se reuniam para checar eventuais erros, scripts com resoluções parecidas e formas de uso da ferramenta. Porém, o último desafio dessa fase, intitulado desafio Zumbi, de maior complexidade exigiu maior tempo dos alunos. Com o acompanhamento

realizado em laboratório observamos uma integração voluntária maior dos alunos para resolução.

A Fase 3 demandou dos alunos desafios com o uso de listas, objetos, eventos e sincronismo. As atividades nessa fase exigiam que os mesmos utilizassem os conceitos iniciais, fazendo com os mesmos buscassem os scripts do início do período. Outra característica dessa fase foi o método utilizado pelo professor da disciplina. Ele elaborou desafios de forma evolutiva, ou seja, a cada script trabalhado, o próximo era a sequência do anterior. Durante as Fases 2 e 3 os enunciados dos desafios eram colocados no ColabWeb bem como as resoluções feitas pelos alunos.

Por fim, a Fase 4 aconteceu durante as últimas 5 aulas do período. Essa fase foi conhecida pelos alunos como Projeto Colaborativo. As seguintes regras foram estabelecidas para o andamento do estudo:

- Equipes de 4 integrantes (A,B,C,D);
- Cada integrante da equipe teria 30 minutos no laboratório para receber as instruções do projeto e executar as atividades;
- As instruções deveriam ser disponibilizadas no ColabWeb, bem como as orientações dos desafios no ambiente virtual.
- A comunicação entre os membros da equipe deveriam ser registradas no fórum criado no ColabWeb;
- A resolução do desafio deveria seguir os passos do método de Polya e também registradas no fórum.
- Cada membro da equipe deveria elaborar a atividade proposta e será avaliado de forma individual.
- A equipe deveria escolher apenas uma das soluções para representar a equipe.

Os desafios devem ser desenvolvidos nos espaços criados para este no *OpenSimulator*.

Na primeira sessão a turma recebeu toda a orientação referente ao projeto sendo expostas todas as regras para sua execução. Após isso, eles responderam a uma avaliação para testar o conhecimento com relação aos conceitos de mundos virtuais e 3D. A aplicação desse exame foi para testar o conhecimento da turma com relação aos conceitos de mundos virtuais e 3D.

Em seguida houve a explanação dos conceitos abordados no teste. Na sequência foi solicitado dos alunos a formação dos grupos. Deu-se início às atividades com 36 alunos formando um total de 9 grupos com 4 integrantes cada. Ressaltando que, para cada grupo, foi criado um fórum de comunicação no ColabWeb.

Para segunda sessão foi preparada a introdução referente a ferramenta *Opesim*. Foram vistos procedimentos de uso incluindo a personalização dos avatares. O ponto chave foi a janela de construção de objetos. Nessa janela os alunos trabalharam comandos de manipulação dos objetos como rotação e posição. Em paralelo foi apresentada a forma de uso do *Scratch for Opesim* usando-se de um exercício onde os alunos realizaram uma atividade que exigia a rotação em 90° de um objeto a cada vez que pressionavam a letra 'g'. Esse script foi copiado do *Scratch for Opesim* para o *OpenSim*.

Na terceira sessão foi ministrado o processo de agrupamento de componente. Esse passo é crucial para a construção de objetos na ilha, pois o funcionamento deles é influenciado caso o agrupamento não seja feito de forma correta. Como diferencial para essa aula adotamos a comunicação com os alunos via *chat*. Essa decisão foi adotada para observar o comportamento colaborativo quanto ao trabalho no ambiente.

Na quarta sessão aconteceu a apresentação das instruções para o trabalho final, com dicas de como proceder com o desafio. Mas uma vez procuramos passar as instruções via *chat* estimulando os alunos no uso e, também, utilizando o recurso de apresentação da própria ferramenta.

Enfim, na quinta sessão, a turma utilizou todas as orientações que foram transmitidas nos dias anteriores e todos do grupo fizeram um ventilador o qual devia começar a girar a cada vez que o “*dono*” se aproximasse a uma distância menor do que 5 metros. Um instantâneo dessa atividade é mostrado na Figura 3.

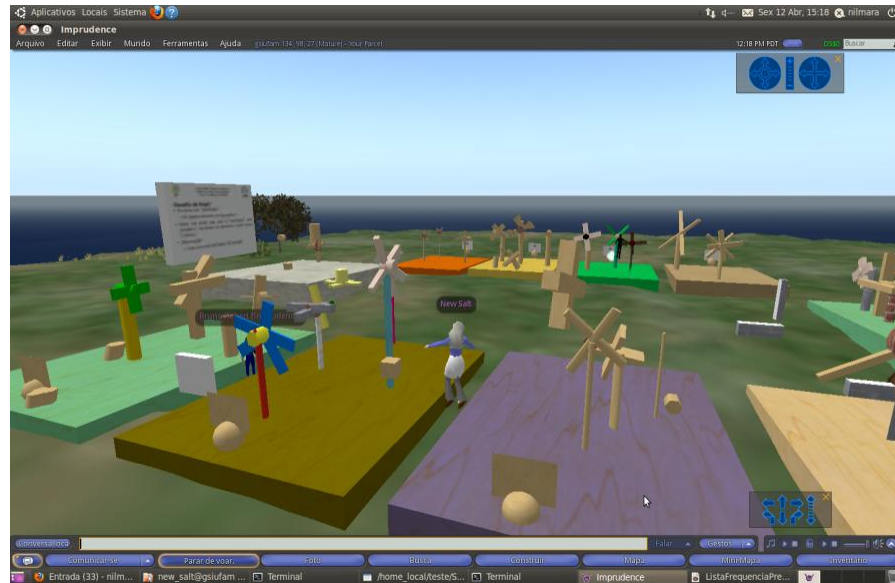


Figura 3. Versões dos grupos para o objeto ventilador.
Fonte: Elaborado pelos autores, 2012.

4.3 Análise dos Resultados

Este projeto proporcionou dados quantitativos e qualitativos. Do ponto de vista quantitativo foram realizados 3 questionários, um foi realizado no segundo dia de aula do período para identificar o perfil dos alunos, e os outros dois foram realizados durante a semana do projeto para levantar o conhecimento sobre os assuntos explorados.

No questionário sobre o conhecimento prévio da turma em temas relacionados 3D incluímos a verificação sobre a familiaridade da turma com ambientes imersivos, como jogos 3D, e constatamos que de um total de 26 alunos que responderam à questão, 16 se definiram como bastante familiarizados. Também observamos que os conceitos diretamente associados a jogos (eg. 'Avatar') são familiares à maioria, embora elementos com interesse mais "acadêmico" (eg. Eixo de Referência) são menos familiares – vide o gráfico de acertos e erros nas questões, apresentado na Figura 4.

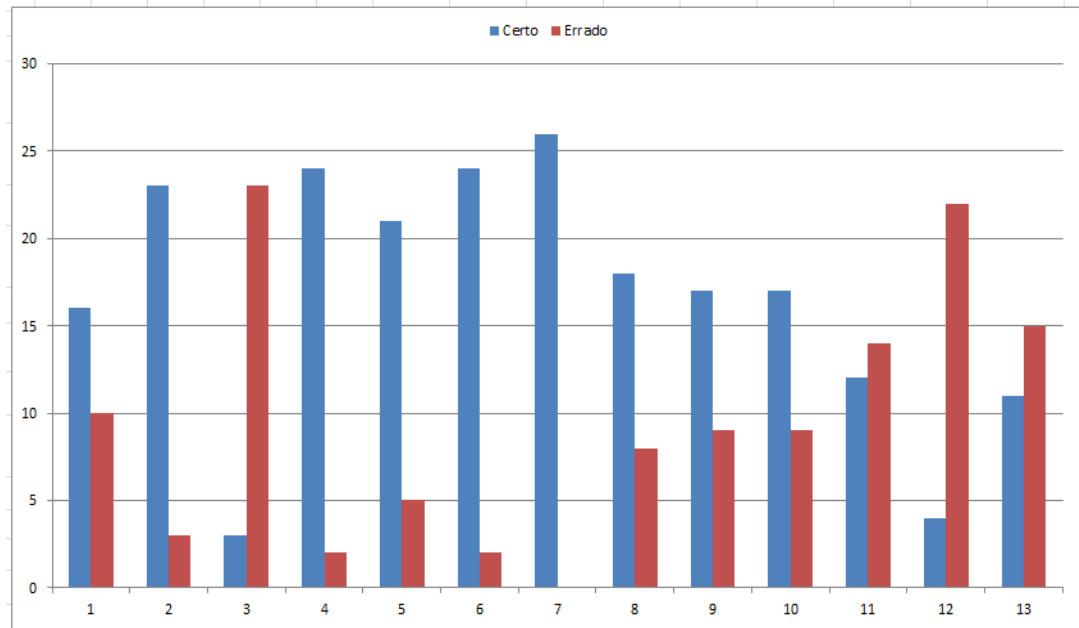


Figura 4. Questionário sobre conhecimento prévio 3D.
Fonte: Elaborado pelos autores, 2012

Na avaliação final aplicada após a execução do Projeto Colaborativo levantamos mais 11 questões referentes ao ambiente 3D, conforme podemos observar na Figura 5. Cabe destacar que dos 32 estudantes, todos demonstraram interesse em desenvolver projetos mais complexos no ambiente 3D (questão 2), 31 acharam que o uso do ambiente contribuiu à disciplina (questão 11), nenhum sentiu dificuldade na transposição do *Scratch* para o *Opensim* (Questão 4).

| Avaliação Final | | | | | | | | | | | |
|-----------------|----------------------------------------------------------------------------------------------------------|-----------------------------|----|----------------------|----|---------------|----|-----------|----|--------------------|----|
| N. | Questões | Escala/Quantidade de Alunos | | | | | | | | | |
| 1 | De modo geral, você considera que o uso do ambiente 3D facilita a resolução de problemas em programação? | sim | 9 | na maioria das vezes | 7 | Algumas vezes | 16 | não | 0 | | |
| 2 | Você teria interesse em desenvolver projetos mais complexos no ambiente 3D? | sim | 32 | não | 0 | | | | | | |
| 3 | O uso da abordagem de Polya contribuiu na resolução de problemas? | sim | 17 | na maioria das vezes | 10 | Algumas vezes | 5 | não | 0 | | |
| 4 | A transferência dos scripts em <i>Scratch</i> para o ambiente 3D foi: | muito fácil | 11 | fácil | 12 | ok | 9 | difícil | 0 | muito difícil | 0 |
| 5 | A experiência dos projetos no ambiente 3D, comparada com sua impressão no <i>Scratch</i> , ficou: | pior | 1 | igual | 9 | melhor | 22 | | | | |
| 6 | Ao longo de toda a unidade, no uso do ambiente 3D, cite: | escrita | | | | | | | | | |
| 7 | A navegação no <i>Imprudence</i> para a consequente definição de comandos, foi: | muito fácil | 5 | fácil | 8 | ok | 15 | difícil | 3 | muito difícil | 1 |
| 8 | A personalização do Avatar, foi: | muito fácil | 5 | fácil | 17 | ok | 6 | difícil | 2 | muito difícil | 2 |
| 9 | A dificuldade em registrar as atividades no seu grupo foi: | baixa | 17 | média | 14 | alta | 1 | | | | |
| 10 | A interação no seu grupo foi: | baixa | 4 | média | 21 | alta | 7 | | | | |
| 11 | De modo geral, você acha que para a disciplina, o uso do 3D: | prejudica a muito | 0 | prejudica | 0 | não influi | 1 | contribui | 19 | contribui bastante | 12 |

Figura 5. Avaliação final referente ao Projeto Colaborativo
Fonte: Elaborado pelos autores, 2012

No aspecto qualitativo, foi importante observar os fatores mencionados pelos alunos para reprovação anterior na disciplina, uma vez que poderia corroborar as escolhas adotadas na proposta. Observando a Tabela 1 podemos notar que “desmotivação” ou elemento equivalente, foi citado como o principal fator para a reprovação.

Tabela 1. Declarações sobre a não aprovação

| Aluno | Q.9 - Se já cursou essa disciplina anteriormente, qual considera o principal motivo para a não aprovação? |
|-------|------------------------------------------------------------------------------------------------------------------|
| A | Falta de vontade |
| B | Falta de interesse |
| C | Ficar 4 meses de greve e não pegar o assunto para estudar |
| D | Não tinha conhecimento algum em programação e isso afetou meu rendimento e também o trabalho atrapalhou bastante |
| E | Falta nas aulas, devido estagio ser no mesmo horário que a aula. |
| F | Sim. me senti desmotivado por não ter começado o curso do inicio |
| G | Focou outras disciplinas |
| H | Falta de interesse e motivação |
| I | Falta de interesse, não levei a serio |
| J | Porque no 1o período parou a faculdade para trabalhar |
| K | Dedicação e atividades pouco comuns (programação massiva, por exemplo) |
| L | Não identificação com a linguagem utilizada, pouca dedicação |
| M | Sim. Não pratiquei as atividades |
| N | Desinteresse, não conseguir estabelecer sentido nas aulas |
| O | Falta de pratica |
| P | Sim. A complexidade da linguagem dificultou meu entendimento |
| Q | Não absorção da ideia principal, dedicação fraca. |

Na Tabela 2 alguns pontos positivos e negativos apontados pelos alunos referentes a experiência no Projeto Colaborativo na disciplina, sendo que declarações semelhantes foram retiradas para evitar acúmulo de informações repetidas. É importante citar que “visualização” e “interação” foram os termos mais citados dentre os aspectos positivos, embora outros como “facilidade” e “solução” também tenham sido frequentes. Com respeito aos aspectos negativos, problemas técnicos com a configuração das placas

gráficas usadas nos computadores do laboratório utilizado estiveram presentes em todos os registros.

Tabela 2. Resumo sobre pontos positivos e negativos mencionados pelos alunos.

| Q.6 Ao longo de toda a unidade, no uso do ambiente 3D, cite: | |
|-------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| + | - |
| O uso do ambiente 3D facilitava a visualização das soluções ajudando a resolver problemas | Durante o uso do Opensim, os travamentos dificultaram a utilização do programa, causando atrasos na resolução dos desafios |
| Melhor visualização dos resultados alcançados através dos comandos e a sensação de estar inserido no ambiente | A lentidão que era desenvolver atividade por conta do ambiente |
| O ambiente 3D proporciona uma melhor visualização, além de dar mais impressão para quem usa | Exige uma capacidade de resolução visual onde vai rodar e ocorre um acréscimo de um terceiro eixo que as vezes confunde o usuário. |
| Caracterização avatar, interagir com outras pessoas podendo ver e discutir em tempo real as diferentes soluções para um mesmo problema | a necessidade de uma boa conexão, placa gráfica e um recurso de comunicação por voz, pois minimizaria você parar a ação para digitar |
| Nova maneira de olhar os objeto, os personagens, pois surge uma nova dimensão e isso influencia bastante na lógica da pessoa | parte gráfica foi o que mais nos prejudicou, pois, por mais fáceis que os desafios fossem, com a lentidão da placa de vídeo tudo se tornava complicado |
| A interação com o professor ficou algo divertido e interativo, as atividades foram simples porem no ambiente tornaram-se bastante interessantes | Os gráficos não são dos melhores, a velocidade dos processos são lentos e travosos. |
| Visualização e melhor facilidade para resolver os problemas | lentidão do sistema e a restrição de acesso (internet) |
| A possibilidade de experimentar uma nova interface e poder usar o conhecimento anteriores aplicando o método de polya | a lentidão pois o pc não suportou os efeitos gráficos do ambiente |

As Tabelas seguintes contêm partes dos diálogos entre os grupos que possibilitaram identificarmos padrões de interação ou atos de fala que auxiliam na observação dos grupos quanto ao seu funcionamento, participação bem como na assimilação de dificuldades. Devido o espaço e a relevância da informação, foram colocados aqui somente diálogos referente as ações dos membros do grupo na escolha da solução do desafio do ventilador, os quais foram identificados padrões de interação.

Na Tabela 3 observamos uma tentativa do membro D em estimular os demais membros na escolha do desafio. O integrante procura influenciar o grupo colocando sugestão para escolha do desafio e um questionamento.

Tabela 3. Diálogo grupo 01: “estimulando os colegas”

| Grupo 01 | |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula 5 | Desafio Ventilador |
| Aluno | Comentários |
| D | Caso não tenhamos as outras opiniões, eu voto para que escolha a sua resolução “B” já que ta pintadinho e está pronto, infelizmente o horário não me permitiu fazer um designer mais elaborado do ventilador que eu fiz. O que você(s) acha (m) ? |
| B | Pode ser o meu se todos concordam. |
| D | Tranquilo então, minha opinião é o ventilador do “B”. Aguardando a opinião de “A” e “C”... |
| A | Também não consegui terminar a atividade, porém vi o ventilador do “B”, então voto no script dele! |

Na Tabela 4 vemos que no grupo 02 o integrante B descreveu como desenvolveu o exercício, mas não se manifestou com relação a escolha. No entanto, observamos como o integrante A assumiu o papel natural de

liderança propondo a votação, comentando construtivamente sobre o trabalho do seu colega e descrevendo sobre a solução.

Tabela 4. Diálogo grupo 02: “assumindo a liderança”

| Grupo 02 | |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula 5 | Desafio Ventilador |
| Aluno | Comentários |
| B | <p>Desafio - Ventilador</p> <p>O objetivo do desafio era criar um ventilador que só girasse quando o avatar chegasse a menos de 5 metros dele. Para fazer o ventilador eu criei dois cilindros, um que serviu como haste do ventilador e outro como base para as palhetas. As palhetas eu fiz usando um cubo e depois copiando ele com (Ctrl + D) para criar as duas, depois eu agrupei as palhetas e a base com (Ctrl + L)</p> <p>No Scratch eu criei um script que fatia o ventilador girar 20 graus sempre que o avatar chegasse a menos de 5 metros e coloquei na base das palhetas do ventilador.</p> |
| C | <p>Escolha para representar a resposta do grupo:</p> <p>Já que dois dos integrantes do nosso grupo não tiveram como terminar o desafio, escolhemos a resposta entre a minha e a do "B". Os dois projetos funcionam corretamente de acordo com o enunciado e os scripts são parecidos, mas vamos escolher a minha resposta para representar o grupo, mas por conta da aparência no ambiente 3D.</p> <p>Script:</p> <pre>Sempre {se (distancia até o proprietário < 5) {gire 15'}}</pre> |

A ênfase para o grupo 04 vai para sua interação no fórum. A comunicação entre eles ocorreu de forma atenuada e coerente. Para eles não houve a necessidade de estarem reunidos fisicamente, mas conseguiram realizar a escolha da solução sem contratempos, conforme observamos na Tabela 5

Tabela 5. Diálogo grupo 04: “seleção tranquila”

| Grupo 04 | |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula 5 | Desafio Ventilador |
| Aluno | Comentários |
| A | Eu vi o do “C”. O script é igual o meu. O “B” não teve tempo suficiente pra terminar o objeto mas o script também funciona. A “D” teve o mesmo problema de tempo. Eu voto no meu. Tive mais tempo pra fazer o script e criar os objetos. |
| B | Então tem que se escolher entre o do “C” e do “A”. Pelo que olhei no fórum os dois estão funcionando, voto no do “A” então. Qual a posição do grupo? |
| C | Pow ngm me curte kkkk, tbm voto no “A” pq o objeto dele ta mais completo... |
| A | Valeu! Também voto no meu! |
| D | Então voto no “A” tbm. Fechou. O desafio escolhido é o do “A”. |

Com relação ao grupo 05, mesmo ficando com 3 integrantes, destacamos o cuidado da equipe em informar sobre a atividade proposta e descrever como foi executado o desafio, bem como o cuidado com a escolha da solução. Veja Tabela 6

Tabela 6. Diálogo grupo 05: “detalhamento cuidadoso”

| Grupo 05 | |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula 5 | Desafio Ventilador |
| Aluno | Comentários |
| B | A atividade da sexta (12/4) foi a conclusão do desafio do ventilador. Como foi passado na quarta, o objetivo era fazê-lo girar sempre que o avatar chegasse a menos de 5 metros do objeto. Infelizmente perdi muito tempo tentando por a base da hélice em cima da haste, e acabei não conseguindo terminar a atividade. O script era bem simples, bastava usar o comando de repetição SEMPRE, e dentro dele usar uma condição de sempre que o dono ficar a menos de 5 metros ele gire 45° |
| A | Bom, como o “D” se manifestou a favor de que escolhamos o melhor resultado, creio que a minha solução foi a melhor. Então, se o “B” concordar, a solução do desafio escolhida para a equipe será a minha. |
| B | Por mim OK!! |
| A | Logo, por 3 votos podemos considerar o resultado da equipe para o desafio do ventilado como a solução feita por mim. |

Uma situação singular aconteceu no grupo 6. A equipe procurou solucionar uma dúvida na escolha do desafio pelo fórum. Um dos integrantes detectou uma falha no desafio de um colega do grupo e procurou chamar atenção para esse item. Isso foi observado como um ponto positivo, pois os integrantes procuraram utilizar o recurso que foi disponibilizado. Observe Tabela 7.

Tabela 7. Diálogo grupo 06: “análise + crítica positiva”

| Grupo 06 | |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula 5 | Desafio Ventilador |
| Aluno | Comentários |
| C | Ei “B”, mas eu olhei e ainda continua com o mesmo problema cara :/, a correção ainda não foi feita, o teu moinho continua girando constantemente sem a presença do teu avatar. O objetivo é quando o avatar do “DONO” estiver ao menos “5 metros” de distancia do “moinho que o pertence” o mesmo giraria. E o teu fica girando sempre... e por esse motivo acharia melhor escolher o meu, é uma opção! |
| B | Se vc entrou no meu ambiente e afastou meu avatar do ventilador e ele continua girando então coloque o seu |
| C | Como assim? :/ |
| B | Pelo o que eu me lembro, o meu avatar estava próximo do ventilador, logo, girava, se afastar ele do ventilador 5 metros ele para, eu tinha testado e funcionava corretamente, vou ver terça n lab |
| C | Hmmm, a então vamos esperar né, pq a “T” que a resposta, mas se estiver certo coloque o seu, mas olhe lá! Pois está sobre a sua responsabilidade mano :) |
| C | Discutido e resolvido, então o projeto escolhido será do “B” |

Conforme observamos na Tabela 8, uma peculiaridade encontrada no grupo 07 foi a espontaneidade e o relato do desafio desenvolvido. A equipe descreveu como resolveu o exercício, falou das dificuldades e ainda usou de bom humor.

Tabela 8. Diálogo grupo 07: “detalhamento e espontaneidade”

| Grupo 07 | |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula 5 | Desafio Ventilador |
| Aluno | Comentários |
| D | No exercício do ventilador, a única dificuldade que senti foi a lentidão do simulador que acaba desmotivando e irritando muito. Acabei demorando muito na construção do objeto, o tempo acabou e não consegui fazer o script. |
| C | Gostei muito desse exercício do ventilador. Consegui fazer ele funcionar. Mas agora agente precisa decidir qual o script será enviado |
| A | <p>Frustrante! :/</p> <p>No exercício do ventidor, eu construí o comando no scratch do opensim que seguia esses passos:</p> <p>Quando se aproximar o dono do objeto</p> <p style="padding-left: 40px;">Se < ((distancia) < (5)) ></p> <p style="padding-left: 80px;">Gire 30 graus</p> <p>Depois eu o copiei em uma nova aplicação do objeto em que eu havia fito o “ventilador”. Que era composto por dois retângulos e um cilindro, sendo que eu agrupei primeiro os 3 retângulos e depois os agrupei com o cilindro!... Ficou horrível, mas atendeu às especificações do exercício</p> |
| C | <p>Já que eu sou o mais bonito do grupo pode ser o meu mesmo. Tô brincando a “D” é a mais bonita.</p> <p>Pode ser o meu mesmo “T”</p> |
| A | Sim, “T”! Vai ser o do “C” |
| B | Com relação ao exercício do ventilador, não pude termina-lo, na verdade não consegui fazer o bicho girar. |

No grupo 08, apesar dos 4 integrantes realizarem as atividades propostas somente 2 integrantes da equipe utilizaram o fórum. Nos demais momentos, observamos contato da equipe no laboratório. O destaque aqui vai para o integrante que procurou explicar o motivo de não ter finalizado o desafio. Note na Tabela 9.

Tabela 9. Diálogo grupo 08: “justificativa”

| Grupo 08 | |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aula | Desafio Ventilador |
| Aluno | Comentários |
| A | <p>Na aula do dia 12/4, fomos capazes de trabalhar na construção de um ventilador e fazê-lo funcionar. Para tal acontecimento tivemos que criar o ventilador com cilindros e cubos, modelando e agrupando. Na parte de fazer ele funcionar a ideia é usar os seguintes comandos:</p> <pre>Forever IF com o sensor Distance to owner dentro do operador <5 Roll 15 degrees</pre> |
| D | <p>Bastante legal criar o objeto e por nele movimentos, faz a computação ficar mais “concreta” podia passar horas ali brincando de programar nele. Errei um pouco os comandos, também ainda estava me habituando com os termos.</p> |
| A | <p>No meu grupo creio que o meu irá ser o escolhido, se o pessoal não tiver alguma objeção.</p> |
| D | <p>Bom, eu perdi muito tempo arrumando os objetos na dimensão devido ao lag, quando comecei o script o tempo já estava esgotado, não desenvolvi muito bem os comandos que eram simples.</p> <p>Escolho o da “A”</p> |

A Tabela 10 apresenta um resumo com os tipos e exemplos de padrões de interação encontrados nos diálogos dos grupos, nas Tabelas 03 a 09.

Tabela 10. Tipos e exemplos de padrões de interação no Fórum

| Categoria | Grupo | Exemplo |
|------------------------------|--------------|----------------------------------------------------------------------|
| Disponibilização de artefato | 07 | “No exercício do ventilador, eu construí o comando no scratch..” |
| Informe | 05 | “...O script era bem simples, bastava usa o comando...” |
| Clarificação | 08 | “...não desenvolvi muito bem os comandos que eram simples...” |
| Confirmação | 04 | “..Fechou. O desafio escolhido é o ...” |
| Pergunta | 01 | “...O que vocês acham?...” |
| Sugestão | 01 | “...caso não tenhamos as outras opiniões, eu voto para escolha...” |
| Chamada de atenção | 06 | “...mas olhe lá! Pois está sobre a sua responsabilidade mano :) ...” |
| Identificação de erro | 06 | “Ei B, mas eu olhei e ainda continua com o mesmo problema...” |
| Explicação | 02 | “No scratch eu criei um script...” |

Analizamos também fragmentos de diálogo no chat do ambiente imersivo. Das cinco sessões de laboratório, uma foi utilizada somente para comunicação via chat. A ideia foi observar o comportamento dos alunos em uma situação de ensino não presencial, sua interação com os colegas das outras equipes e com o professor. Ressaltamos essa necessidade, pois em cada sessão de laboratório os membros das equipes estavam separados, por exemplo, na primeira sessão somente os integrantes letra A estavam no laboratório. Com isso, além de conduzimos alguns padrões de interação outros foram identificamos espontaneamente.

No começo da aula destacamos o item de “Pergunta” para saber se todos os alunos estavam atentos ao início das instruções colocando o questionamento se todos estavam visualizando as mensagens no chat. Observe o dialogo a seguir:

| | |
|--------------------|-----------------------------------------------------|
| [2013/04/10 07:07] | NS: *bom... hoje vamos criar um " ventilador" :) |
| [2013/04/10 07:08] | MM: o.0 |
| [2013/04/10 07:08] | LA: :O |
| [2013/04/10 07:08] | NS: Todos estao visualizando minhas msg? |
| [2013/04/10 07:08] | LA: sim |
| [2013/04/10 07:08] | MM: sim |
| [2013/04/10 07:08] | NS: ok |
| [2013/04/10 07:08] | CF: x- |
| [2013/04/10 07:08] | VL: sim |
| [2013/04/10 07:08] | LT: nao pode ser um ar condicionado? |
| [2013/04/10 07:08] | NS: rrsrrss... |
| [2013/04/10 07:08] | IA: Yes |
| [2013/04/10 07:09] | MM: ainda não temos energia, ou podemos criar tabém |
| [2013/04/10 07:09] | NS: é porque eu preciso fazer ele girar.... |

Na sequência das atividades o padrão de interação “Chamada de atenção” foi utilizado para reforçar a atenção dos alunos na transmissão de instrução. Vejam que utilizamos a seguinte frase “..Vou iniciar... observem...”:

| | |
|--------------------|----------------------------------------------------------------------------------------------------------------------|
| 2013/04/10 07:09] | NS: Vou iniciar.. observem |
| [2013/04/10 07:10] | AJ está Online |
| [2013/04/10 07:11] | NS: Primeiro passo: criar os objetos (somente observem depois vou dar um tempo para vcs criarem os seus...) |
| [2013/04/10 07:15] | NS: Vou precisar de um outro objeto cubo-entao vou duplicar - Ctrl + D |
| [2013/04/10 07:19] | NS: Segundo Passo : vou agrupar os objetos (observacao: o objeto que vai girar deve ser o ultimo a ser selecionado!) |
| [2013/04/10 07:20] | NS: alguma pergunta ate aqui? |
| [2013/04/10 07:20] | LA: n.. ta ok :) |
| [2013/04/10 07:21] | IA: tudo maraaa |
| [2013/04/10 07:21] | NS: Certo |
| [2013/04/10 07:22] | NS: :) |

Padrões de interação foram identificados durante um momento de apoio entre os integrantes. Observem abaixo que o integrante “MM” faz uma

pergunta e na sequência o integrante “LA” explica o procedimento que deve ser feito:

| |
|---------------------------------------------------------------------------------|
| [2013/04/10 07:26]MM: onde eu mudo a textura? |
| [2013/04/10 07:26]LA: lá onde cria o objeto tem uma aba que tem a opção textura |
| [2013/04/10 07:26]LA: textura* |

O padrão de “Clarificação” identificado em um momento de dúvida do integrante “LT”. Ele deixa claro como não conseguiu realizar determinada ação no ambiente. Na sequência sua dúvida é eliminada com o padrão de “Explicação”:

| |
|-----------------------------------------------------------------------|
| LT: nao consigo colocar os dois blocos com uma curz em cima da roda |
| [2013/04/10 07:31]LT: comufas? |
| [2013/04/10 07:31]NS: ´E necessario deixar a opcao "posicao" marcada! |

4.4 Follow UP

A disciplina ministrada após a disciplina de IC, Estrutura de Dados I (EDI), também foi objeto de observação. Essa disciplina compõe parte da grade do curso que abrange o contexto de programação, e ao seu término foi realizada dois tipos de entrevistas. Uma com o professor da disciplina e outra com os alunos. O objetivo dessas entrevistas era verificar se em algum ponto a experiência vivenciada em IC contribuiu para EDI.

A entrevista com o professor da disciplina procurou conhecer como a mesma foi planejada e executada. Essas informações irão contribuir para as novas tomadas de decisão referente ao projeto.

Para entrevista com o professor da disciplina utilizamos a ferramenta 5W2H, basicamente, é uma metodologia que reúne as informações julgadas como mínimas necessárias para a execução de um determinado plano de ação. Ela engloba as seguintes questões: *What, Why, Where, When, Who, How e How much*. E, para nosso caso formulamos o seguinte questionário:

- 1 - O que é feito na disciplina Estrutura de Dados I? (Conteúdo programático adotado, métodos ou técnicas).
- 2 - Por que foi escolhido? (Ou seja, por que o conteúdo foi delimitado dessa forma? Por que foi adotado determinado método ou técnicas?)
- 3 - Onde é aplicado o conteúdo escolhido? (somente em sala de aula, laboratório, algum outro ambiente?)
- 4 - Quando foi feito? (tempo, carga horária, quando é feita atividade teórica e prática?)
- 5 - Por quem foi feito? (somente professor, ou há monitores?)
- 6 - Como foi feita a aplicação desse conteúdo, do método ou técnica?
- 7 - Quanto custa? (precisou adquirir ferramentas ou equipamentos?)

O professor explicou que o conteúdo programático abrange desde revisão de algoritmos com Portugol até a estrutura de dados Pilhas. Ele utiliza a seguinte estratégia: nas duas primeiras semanas de aula, aplica uma lista de exercícios para verificar o nível de conhecimento da turma. Com o resultado da lista em mãos ele conseguiu observar que alguns alunos precisavam de uma prática maior na elaboração de algoritmos. Por isso, ele dividiu o período letivo em três partes. Sendo que, um terço das aulas foi para Portugol e os outros dois terços com práticas de laboratório utilizando linguagem C. Durante as aulas de laboratórios o professor dispunha da colaboração de dois alunos monitores que auxiliavam a turma durante as atividades.

Já para os alunos a pergunta chave foi: - Ter participado do Estudo Piloto com a ferramenta Scratch e ambiente 3D contribuiu para disciplina EDI? Localizamos 10 alunos que haviam participado do Projeto Piloto na disciplina de IC. Desses 10 alunos 2 não conseguiram aprovação na referida disciplina. Esse contato foi feito individualmente para a resposta de um aluno não influenciar nas respostas dos demais. Porém, mesmo realizando a entrevista dessa forma a resposta foi praticamente unanime, isto é, todos atribuíram o fato de que a prática do raciocínio e da ênfase na resolução de problemas durante a disciplina de IC contribuiu bastante para EDI, principalmente nos exercícios que envolviam conceitos de jogos. O interessante é que todos comentaram que o tempo utilizado para Portugol poderia até ter sido menor, durante EDI. No caso dos dois alunos que não

obtiveram aprovação os mesmos atribuíram a desmotivação, mas não apresentaram justificativas maiores. Conceituamos retorno dos 8 alunos aprovados como positivo, pois visualizamos que a prática exercida durante a disciplina IC foi produtiva no que diz respeito à resolução de problemas com estruturas básicas de algoritmos. Entretanto, apesar de ser colocada como ponto positivo a visualização e produção no ambiente 3D, este foi considerado, nesse primeiro momento, como não adequado para uso na disciplina devido sua alta necessidade de conexão com a Internet e placa gráfica.

Por isso, nesse ponto, procuramos dividir nossa análise em três ramificações. A primeira diz respeito ao uso de um ambiente que seja mais acessível e que não demande tantos recursos como o ambiente 3D. Nesse aspecto estamos considerando o fator motivacional do ambiente e a interação colaborativa que ele proporciona. A segunda tratando do uso da ferramenta Scratch. E a terceira, a melhoria na aplicação de estratégias de jogos. Essa análise nos rendeu três Estudos de Caso Exploratório.

O primeiro estudo abordou o uso da ferramenta Scratch e a necessidade de outras ferramentas ou recursos para o desenvolvimento de programas. O segundo estudo abordou em pequena escala o uso de aplicação que trabalha com o raciocínio lógico e a resolução de problemas com foco em estratégias de jogos em um ambiente móvel, que nesse caso seria uma substituição do ambiente 3D. A seguir detalharemos cada um deles.

4.4.1 Primeiro Estudo de Caso Exploratório

Tratando-se do uso da ferramenta Scratch percebeu-se que seu uso associado há estratégias de jogos despertaram um interesse maior no aluno. Além do fato de se trabalhar em uma disciplina que o desenvolvimento de programas é algo abstrato e a ferramenta Scratch torna esse

desenvolvimento mais concreto. Entretanto, devido à ausência de algumas estruturas importantes na programação, como por exemplo, vetor multidimensional e registro, ela não pode ser utilizada em todos os assuntos, ou seja, trabalhar somente com ela durante as disciplinas relacionadas à programação. Nesse caso, é necessário que haja uma transição no uso de ferramentas.

Todavia, na fase de uso com o Scratch o uso das estruturas é mais concreto devido a sua estrutura de blocos e recursos visuais. E, nessa fase de transição o aluno passaria para algo mais abstrato, nesse caso o uso das estruturas de dados. Então, nos perguntamos qual estratégia utilizar para essa mudança, ou seja, como fazer com que o aluno consiga visualizar a necessidade de outros recursos ou ferramentas para dar continuidade nos assuntos da disciplina.

Por isso, decidimos realizar um experimento seguindo uma estratégia onde o próprio aluno teria que visualizar essa necessidade. Para isso chegamos na seguinte proposta: O aluno deveria propor a criação de alguma aplicação utilizando o recurso de prototipagem. Em seguida ele teria que desenvolver essa aplicação no Scratch. Após o planejamento, o estudo foi realizado em uma turma de Introdução a Computação do curso de Matemática, que nos forneceu indícios para essa análise.

a. Participantes

Tal estudo foi realizado com 32 alunos do Instituto, com idades de 16 e 50 anos. Os alunos deviam estar devidamente matriculados na disciplina e participar de todos os passos do estudo. Para identificação dos integrantes durante a análise será atribuído uma sequência numérica.

b. Material

Para coleta de dados durante o experimento utilizamos os seguintes instrumentos: Questionário para Levantamento de Perfil, Questionário de Avaliação de Conhecimento. Além disso, foi necessário aplicar um treinamento sobre prototipação em papel, um treinamento da ferramenta Scratch, bem como explanação sobre as estruturas básicas de Algoritmos.

Para conhecermos o perfil dos participantes o questionário procurou abordar tópicos como: já cursou a disciplina Introdução a Computação, cursou algum curso técnico durante o ensino médio, como utilizam o computador nas horas vagas, se utiliza jogos, se já programou em alguma linguagem dentre outros.

O questionário para avaliar o conhecimento buscou identificar se os participantes tinham conhecimento com relação ao conteúdo da disciplina Introdução a Computação, mais especificamente nos conceitos e estruturas iniciais. Por isso, os itens abordados foram: Você sabe o que é um algoritmo? Você já escreveu algum programa? Você sabe o que são constantes e variáveis? Você sabe o que são estruturas condicionais? Você sabe o que são estruturas de repetição? Você sabe o que são vetores e matrizes? Você sabe o que é recursão? Como Você avalia o seu conhecimento em “Algoritmos e programação de computadores”?

No treinamento sobre prototipação em papel os alunos aprenderam a como desenvolver sua ideias de forma que estas ficassem inteligíveis para eles e para o professor, sendo escolhido justamente pelo seu modo rápido para projetar e tornar visualmente fácil as ideias das aplicações. Além do treinamento da ferramenta Scratch.

c. Procedimentos

Esse trabalho foi distribuído em 8 aulas. Uma aula para treinamento sobre prototipação e preenchimento dos dois questionários, quatro aulas para treinamento com Scratch e explanação sobre as estruturas iniciais de

algoritmos; e, as demais aulas para desenvolvimento da aplicação. Para o relatório o tempo utilizado foi extraclasse. O trabalho foi desenvolvido em dupla.

Antes de iniciarmos o treinamento sobre Prototipação os alunos preencheram os questionários e foram orientados sobre os passos que seriam tomados durante essa atividade, ou seja, que o trabalho seria feito em dupla e em dividido em três tarefas: prototipagem em papel, desenvolvimento da aplicação no Scratch e Relatório.

Após o treinamento sobre prototipação os alunos receberam a instrução para organizarem suas duplas e propor o desenvolvimento de alguma aplicação realizando a atividade de prototipagem. Todas as duplas finalizaram essa atividade.

A partir da segunda aula já iniciamos a explanação dos conceitos iniciais de Algoritmos e o treinamento com Scratch. Os tópicos vistos foram: conceitos de variável e constante, estruturas condicionais, estruturas de repetição, utilização de operadores e introdução à lista. Das quatro aulas, duas foram utilizadas somente para o desenvolvimento da aplicação.

Na última fase os alunos desenvolveram o relatório. Nessa tarefa as duplas deveriam descrever sobre sua aplicação, o objetivo que almejavam alcançar, os comandos utilizados do Scratch, bem como os pontos que não conseguiram desenvolver.

d. Análise dos resultados

Iniciamos a análise dos dados com a verificação do questionário de Levantamento de Perfil. Conforme podemos visualizar na Figura 6, dos 32 alunos cinco fizeram curso técnico na área de computação, 17 fizeram outros cursos na área de informática e praticamente todos utilizam recursos da área para comunicação ou informação, como e-mail e redes de relacionamento.

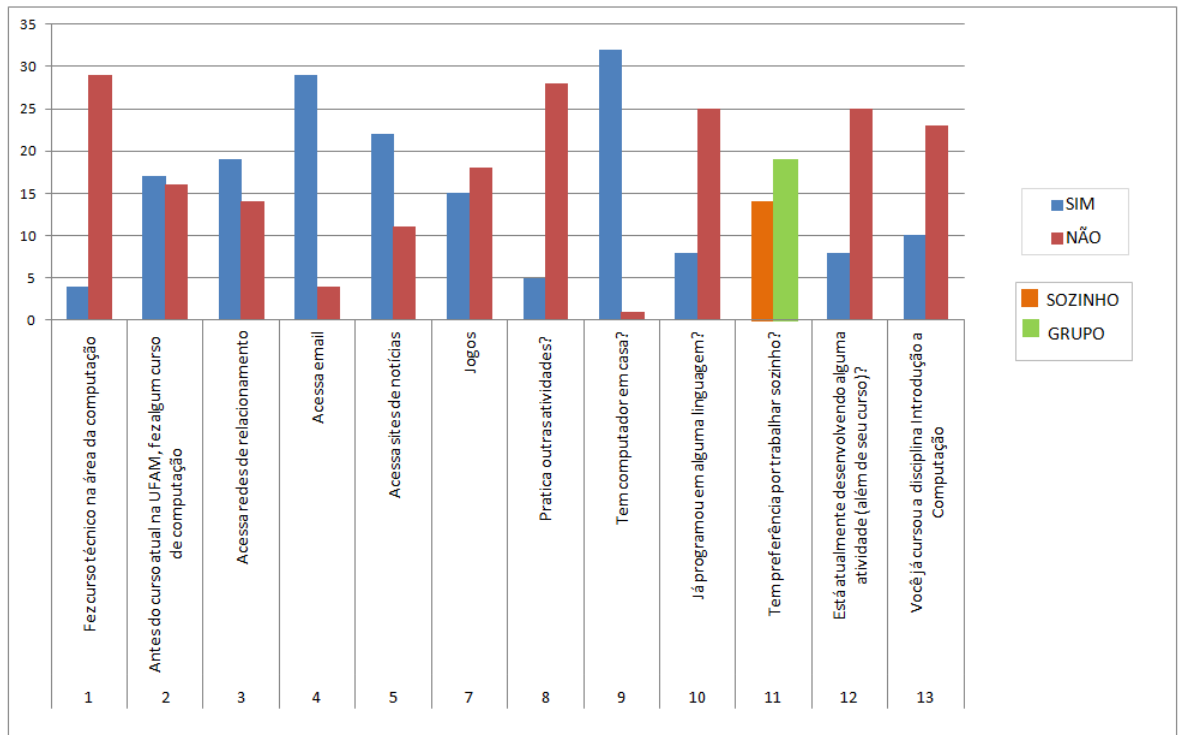


Figura 6. Levantamento de Perfil
Fonte: Elaborado pelos autores, 2014.

Destacamos a questão 11 do Levantamento de Perfil, a fim de chamar atenção para a quantidade de alunos que marcaram como preferência realizar trabalho prático sozinho, ou seja, 14 alunos. No início ficamos atentos, pois isso poderia gerar alguma resistência, mas no decorrer das atividades não destacamos nenhuma dificuldade.

A seguir iniciamos a análise do questionário de Avaliação do Conhecimento com questões cujo conteúdo aborda conceitos de algoritmos. Observamos na Figura 7 que os conceitos de variável, constante e matriz são mais familiares por ser também utilizados em conceitos matemáticos. Já os conceitos de algoritmos propriamente ditos, como as estruturas condicionais e repetição prevaleceu a opção de número 1, “Não tenho ideia”.

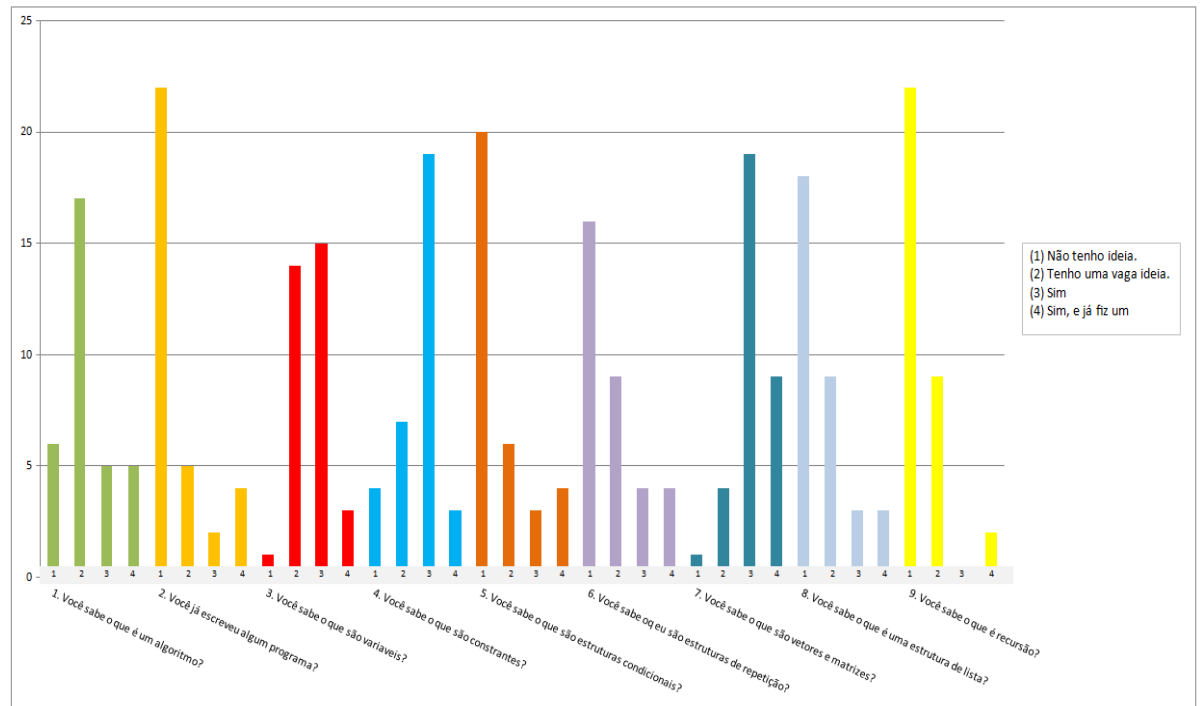


Figura 7. Questionário de Avaliação de Conhecimento
Fonte: Elaborado pelos autores, 2013.

Conforme comentado no tópico C procedimentos, após o treinamento de prototipagem as duplas iniciaram o desenvolvimento de suas ideias. No total, 16 duplas participaram dessa atividade com as propostas descritas na Tabela 11.

Tabela 11. Duplas X Proposta de Aplicação
Fonte: Elaborado pelos autores, 2014.

| No. | Aluno | Aplicação |
|-----|-----------------------------------|--------------------------------|
| 1 | Amanda Cardoso Silva | Photo Facil |
| 2 | Jonas Cavalcante Santos | Photo Facil |
| 3 | Caio Cesar de C. Cantuário | ROMA (A Criação de um Império) |
| 4 | Rodrigo Antonio T. da Silva | ROMA (A Criação de um Império) |
| 5 | Caio Julio Cesar Lima Barroso | Tenis de Mesa |
| 6 | Oéssimo da S. Padilha | Tenis de Mesa |
| 7 | Carina Vargas | Jogo de Obstáculos |
| 8 | Olga Michelle Queiroz Damas | Jogo de Obstáculos |
| 9 | Edson S. Menezes | Space Racing |
| 10 | Jose Augusto Guedes | Space Racing |
| 11 | Fabrizio dos S. C. Penna | Pettrek |
| 12 | Raione da Silva Souza | Pettrek |
| 13 | Francisco Carlos M. Cabral Junior | Lazy Foot |
| 14 | Thiago Ferreira Cacau | Lazy Foot |
| 15 | Geise Souza | Tetris |
| 16 | Leonardo Cardoso | Tetris |
| 17 | Igor Castro Chucre | MatePratica |
| 18 | Janderson N. de Araujo | MatePratica |
| 19 | Italo Nascimento da Costa | Jogo da Prenha |
| 20 | Jean Macedo | Jogo da Prenha |
| 21 | Jessica Heloisa de O. Teixeira | Sky Free |
| 22 | Elivando B. Dias | Sky Free |
| 23 | Jessica Martins Valim | Manaus Transito |
| 24 | Rafhael de A. Monteiro | Manaus Transito |
| 25 | João Ricardo Oliveira Ferreira | Pontos em forma de quadrado |
| 26 | Ulisses Gaston Melho Junior | Pontos em forma de quadrado |
| 27 | José Eduardo M. da C. Filho | Ghost Rider |
| 28 | Rafaela Marques Pinheiro | Ghost Rider |
| 29 | Leonardo Pessoa Matos | O Labirinto |
| 30 | Mauro Daisuke | O Labirinto |
| 31 | Luiz Alves Ferreira Neto | Math Game |
| 32 | Taywan PhoebeRaabe M. Canté | Math Game |

Podemos notar nos exemplos abaixo que alguns projetos foram um tanto ousados, pois a proposta feita inicialmente é que eles deveriam desenvolver suas ideias no Scratch. A Figura 8 mostrar uma proposta chamada Transito Manaus. Observem que a dupla em questão projetou um aplicativo cujo objetivo era fornecer informações via satélite, em conjunto com um gráfico que mostrasse a situação atual do transito.

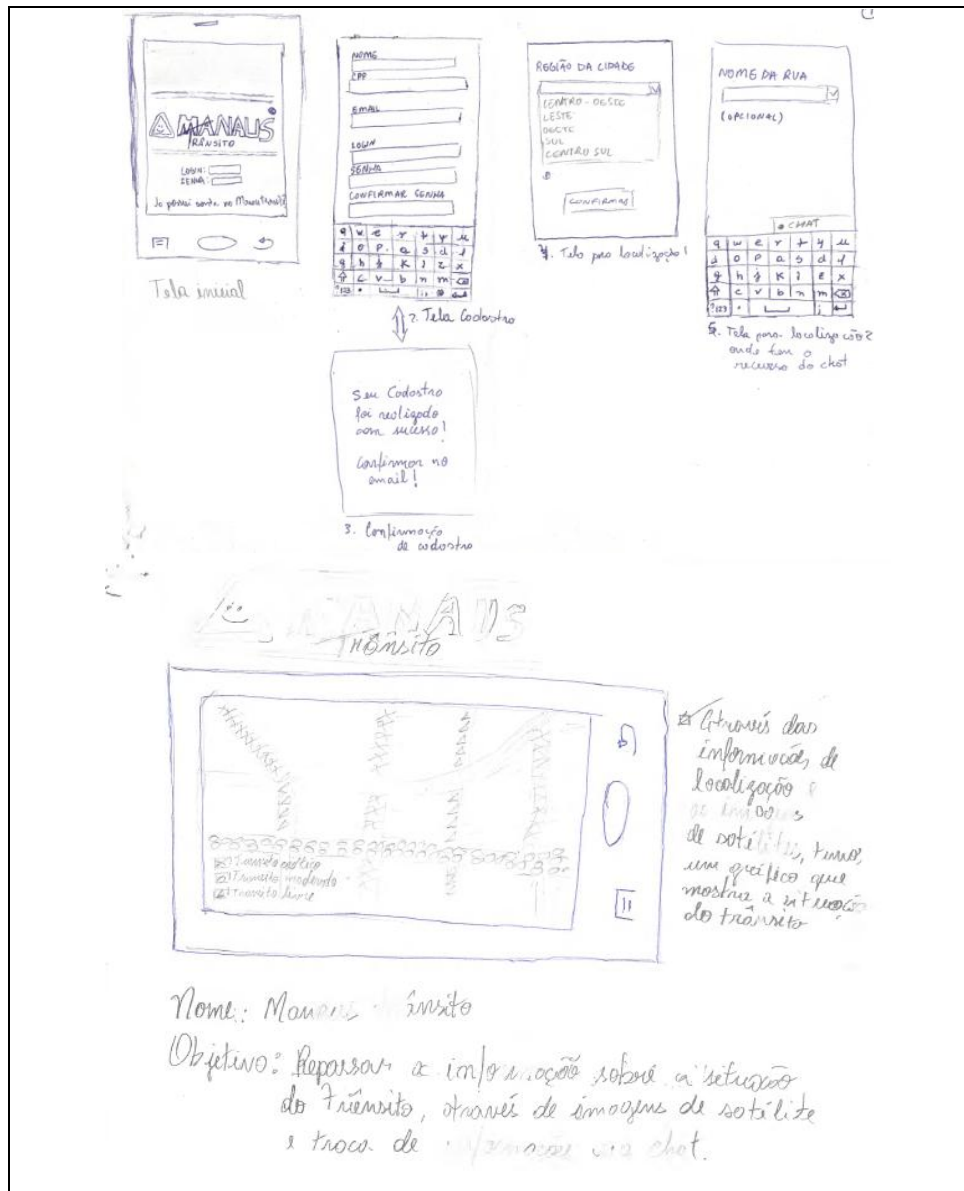


Figura 8. Instantâneo de Prototipagem Manaus Trânsito
Fonte: Elaborado pelos autores, 2014.

Durante as aulas dedicadas ao desenvolvimento a dupla chegou aos resultados apresentados na Figura 9. A figura apresenta um instantâneo de duas telas da aplicação que foi desenvolvida. Nesse período, acompanhamos os alunos e conforme partiam dúvidas, elas eram esclarecidas informando até que ponto eles iriam conseguir atingir no desenvolvimento. Dessa forma, ficou compreensível para dupla que eles iriam precisar de recursos que o Scratch não possui, como por exemplo, comandos que tratam de localização por satélite e recursos de dispositivos moveis.



Figura 9. Telas do Desenvolvimento no Scratch - Trânsito Manaus
 Fonte: Elaborado pelos autores, 2014.

Após a fase de desenvolvimento as duplas realizaram a escrita do relatório. Na Figura 10 segue o instantâneo de um trecho desse trabalho da dupla em questão.

Foram notadas dificuldades ao criar esse aplicativo, por mais que o programa Scratch permite a criação de histórias interativas, animações, jogos, música e diversos outros tipos de arte onde o processo de construção de um design seja necessário, a criação do Trânsito Manaus necessitou de mais recursos na área de controle especificamente de comandos para que o aplicativo funcionasse conforme o pedido.

Figura 10. Instantâneo do Trecho do relatório da dupla “Trânsito Manaus”.
Fonte: Elaborado pelos autores, 2014.

Como uma segunda demonstração temos a dupla que propôs a aplicação Lazy Foot. A proposta dessa dupla seria ter uma aplicação que iria centralizar todos os serviços de lanche e restaurante da Universidade para que alunos e funcionários realizassem seus respectivos pedidos. Observe na Figura 11 a prototipagem da aplicação.

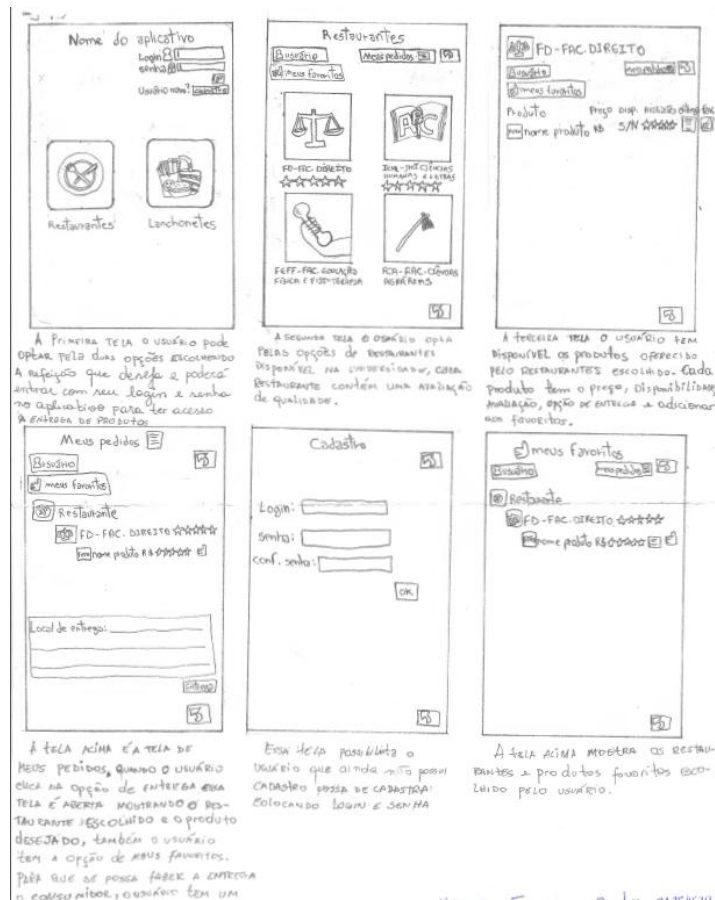


Figura 11. Instantâneo da Prototipagem da aplicação Lazy Foot
Fonte: Elaborado pelos autores, 2014.

Agora, vamos comparar com as telas da aplicação desenvolvida no Scratch. Apesar de ser uma aplicação que envolvia apenas cadastro de clientes e menus, essas informações deveriam ser colocadas em forma de registro. Estrutura esta que o Scratch não possui. Além de recursos móveis que não podem ser implementados na ferramenta.



Figura 12. Telas da aplicação *Lazy Foot* no Scratch
Fonte: Elaborado pelos autores, 2014.

O resultado do Relatório dessa dupla pode ser observado na Figura 13.

Certos objetivos do aplicativo Lazy foot não puderam ser feitos no Scratch por questões de tempo e recursos. Queríamos propor opções de entrega entre setores do campus, opções de avaliações e de telas e de opções de favoritos. O menu dos restaurantes e lanchonetes poderiam ser “logados” por alunos e servidores da UFAM. Onde os funcionários dos estabelecimentos poderiam atualizar seus produtos e etc.

Figura 13. Instantâneo do Trecho do relatório da dupla *Lazy Foot*
Fonte: Elaborado pelos autores, 2014.

Durante o desenvolvimento das propostas fizemos um acompanhamento no laboratório e, conforme a aplicação era desenvolvida as dúvidas eram esclarecidas fazendo com que os alunos entendessem o motivo de não conseguirem finalizar seus programas. Em suma, colocamos como positivo o fato deles mesmos conseguirem enxergar que é necessário mudar de ferramenta conforme as aplicações tornam-se mais robustas.

4.4.2 Segundo Estudo de Caso Exploratório

Esta seção apresenta um estudo de caso exploratório teve como objetivo compreender como a colaboração com ambientes visuais e dispositivos móveis podem ser aplicadas neste cenário. Sendo uma maneira de incentivar a interação entre os alunos através de jogos, usamos um jogo de caça ao tesouro que consiste em uma trilha onde os alunos encontrar pistas, como questões que devem ser respondidas ou atividades que devem ser feitas, a fim de obter o tesouro. Este tipo de jogo usa uma técnica de *peddy–paper* que é uma atividade geralmente ligada à aquisição de conhecimento sobre determinado tema ou local.

O objetivo central dessa abordagem é a de aumentar o interesse e o conhecimento dos alunos sobre questões de computação introdutórias. Os alunos são incentivados a experimentar a lógica de uma maneira prática e agradável e desenvolver habilidades de resolução de problemas.

A fim de motivar os alunos e testar a proposta foi desenvolvido um protótipo de jogo utilizando a ferramenta Mockup⁸. Vejamos nos próximos tópicos uma explicação sobre a aplicação e os passos do estudo.

a. Caça ao Tesouro

O jogo Caça ao Tesouro tem por objetivo explorar os conceitos e estruturas iniciais de algoritmos com pistas mapeadas com foco em colaboração. Este tipo de jogo usa uma técnica de “peddy–paper” que é uma atividade geralmente ligada à aquisição de conhecimento sobre determinado tema ou local. Na Figura 14 Tela de abertura do jogo Caça ao Tesouro. Figura 14 podemos conhecer a tela de apresentação do jogo.

⁸ <https://moqups.com>



Figura 14 Tela de abertura do jogo Caça ao Tesouro.
Fonte: Elaborado pela autora, 2013.

Utilizamos um jogo que consiste em uma trilha onde os alunos encontram pistas, como enigmas que devem ser respondidos ou atividades que devem ser feitas, a fim de obter um tesouro. Pistas deverão ser anexadas em locais geo-referenciados e somente ficarão visíveis quando o participante ficar em torno de um local específico. Sua distribuição acontece no início do jogo quando cada participante recebe uma peça de quebra-cabeça com cor específica. Observe Figura 15.

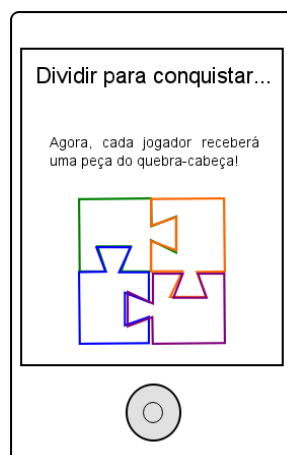


Figura 15. Tela de distribuição de peças do jogo
Fonte: Elaborado pela autora, 2013

As equipes serão formadas por quatro participantes. Serão quatro pistas para cada aluno e a descoberta da primeira pista levará a segunda, e assim por diante. Quando todos os membros da equipe desvendarem seus respectivos enigmas, eles devem realizar um último desafio juntos, conforme visualizamos na Figura 16, a fim de obter o tesouro. As pistas e o desafio final trabalham com questões que estimulam o raciocínio lógico e baseiam-se em conceitos de algoritmos.

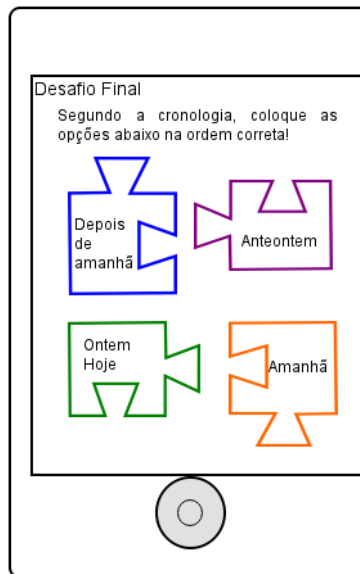


Figura 16. Tela com desafio final
Fonte: Elaborado pela autora, 2013.

Para avaliar a efetividade da proposta em questão, também definimos o estudo de caso exploratório. O objetivo do estudo foi estruturalmente definido segundo o paradigma GQM (Goal-Questio-Metrics) (Basili, 1998), conforme Tabela 12. Objetivo do Estudo Exploratório segundo o paradigma GQM. Tabela 12.

Tabela 12. Objetivo do Estudo Exploratório segundo o paradigma GQM.
Fonte: Elaborado pela autora, com base em Basili (1998), 2014

| | |
|---------------------------|------------------------------------------------------------|
| Analisar | O jogo Caça ao Tesouro |
| Com o propósito de | Caracterizar |
| Em relação a | Aprendizagem, Motivação e Experiência do Aluno. |
| Do ponto de vista | Dos Pesquisadores |
| No contexto de | Aprendizagem de Programação com um grupo de alunos da UFAM |

A análise inicial do Caça ao Tesouro foi feita com base no modelo específico para jogos educacionais proposto por Savi, (2014). Segundo os autores, este modelo é baseado no modelo de avaliação de programas de treinamento de Kirkpatrick (1994), nas estratégias motivacionais do modelo ARCS [Keller, 1987], na área de experiência do usuário e na taxonomia de objetivos educacionais de Bloom [Anderson e Krathwohl, 2001].

O modelo possui o foco em jogos que possam ser utilizados como material educacional para apoiar o processo de ensino e aprendizagem da disciplina, e, ao estruturarem os itens da escala de avaliação, os autores procuraram se o jogo consegue motivar os alunos a utilizarem o recurso como material de aprendizagem, se proporciona uma boa experiência nos participantes, e se gera uma percepção de utilidade educacional entre seus usuários, ou seja, se os alunos acham que estão aprendendo com o jogo.

b. Participantes

Tal estudo foi realizado com 8 alunos do Instituto, com idades de 18 e 32 anos. Os critérios para a seleção da amostra da população estudada dependeram de alguns fatores como: voluntários que já cursaram as disciplinas iniciais do curso de Ciência da Computação e áreas afins, ou seja, que já tivessem experiência com a disciplina para poderem opinar quanto ao uso do jogo e interesse por parte destes em participar da pesquisa. Para

identificação dos integrantes durante a análise será atribuído uma sequência alfabética (A, B, C, D, E, F, G, H).

c. Material

Para coleta de dados durante o experimento utilizamos os seguintes instrumentos: Questionário para Levantamento de Perfil, Questionário de Avaliação de Conhecimento, Questionário para Perspectiva Pessoal Pós-Jogo e o Questionário para Avaliação de Jogos Educacionais. O experimento contou também com a criação de um roteiro do jogo.

Para conhecermos o perfil dos participantes o questionário procurou abordar tópicos como: já cursou a disciplina Introdução a Computação, cursou algum curso técnico durante o ensino médio, como utilizam o computador nas horas vagas, se utiliza jogos, se já programou em alguma linguagem dentre outros.

O questionário para avaliar o conhecimento buscou identificar se os participantes tinham conhecimento com relação ao conteúdo da disciplina Introdução a Computação, mais especificamente nos conceitos e estruturas iniciais. Por isso, os itens abordados foram: Você sabe o que é um algoritmo? Você já escreveu algum programa? Você sabe o que são constantes e variáveis? Você sabe o que são estruturas condicionais? Você sabe o que são estruturas de repetição? Você sabe o que são vetores e matrizes? Você sabe o que é recursão? Como Você avalia o seu conhecimento em “Algoritmos e programação de computadores”?

No questionário para Perspectiva Pessoal Pós-Jogo foi solicitado que os alunos descrevessem sua visão após a participação no jogo. Nesse questionário os alunos deveriam informar se identificaram durante o uso os conceitos e estruturas iniciais de algoritmos. Essas questões foram previamente respondidas pelo avaliador do jogo para fins de comparação, ou seja, se os alunos conseguiram perceber esses itens conforme a visão do avaliador.

Já o questionário para Avaliação de Jogos Educacionais são apresentados na forma de afirmações para os alunos indicarem o quanto concordam com elas de acordo com uma escala Likert de 7 pontos variando de “discordo fortemente” até “concordo fortemente”, em uma escala que vai de -2 até +2.

d. Procedimento

O trabalho teve início com uma reunião para orientação e esclarecimentos do experimento, todos os participantes assinaram um Termo de Consentimento. Após autorizações, foi solicitado aos participantes o preenchimento do questionário de perfil e na sequência o questionário de avaliação do conhecimento. A partir desses passos os alunos foram divididos em duas equipes de 4 integrantes, para atender a estrutura do jogo. A divisão das equipes foi feita pelo avaliador que fez o papel de Professor, já que a proposta era simular uma atividade em sala de aula. A divisão ficou da seguinte forma: equipe 1 (B,D,E,G) e equipe 2 (A,C,F, H).

Para os testes com o jogo foi utilizado o método de prototipagem. A ferramenta *Moqups* foi utilizada para implementação de um protótipo funcional, ou seja, possui algumas funcionalidades e permite uma interação simulada destas funcionalidades. Com isso foi possível aos participantes realizarem os testes em dispositivos móveis. Um segundo avaliador fez o papel das funcionalidades que não estavam implementadas, ou seja, ele era o Sistema.

No início do jogo, o Avaliador que fez o papel do Sistema distribuiu uma cor para cada integrante. A partir disso cada integrante começou a trabalhar com seu primeiro enigma. Quando eles resolviam a questão, a resposta devia ser dita para o Avaliador Sistema, pois ele estava com o gabarito das questões. Caso o raciocínio estivesse correto o participante poderia passar para o próximo enigma, senão ele continuava na mesma tela. Cabe ressaltar que os enigmas escolhidos para o teste estavam entre nível fácil e médio. Somente o último enigma era considerado nível difícil.

Sendo a proposta do jogo colaborativa, os integrantes das equipes podiam trocar ideias e apoio entre si, bem como o avaliador no papel do professor podia dar orientações no sentido de auxiliar na estrutura do raciocínio do aluno, mas não poderia fornecer respostas. As duas equipes atenderam ao procedimento do jogo, e, ao chegarem à resolução do quarto enigma todos compreenderam que somente poderiam passar para o último enigma se os 4 membros tivessem resolvidos suas respectivas questões.

Durante a execução do jogo, o avaliador no papel de professor buscou tomar nota de algumas ações dos alunos para análise complementar posterior. A primeira equipe finalizou o teste no tempo de 23min. e a segunda equipe fechou o trabalho em 10min.

e. Análise dos Resultados

Iniciamos a análise dos dados com a verificação do questionário de Levantamento de Perfil. Conforme Figura 17.

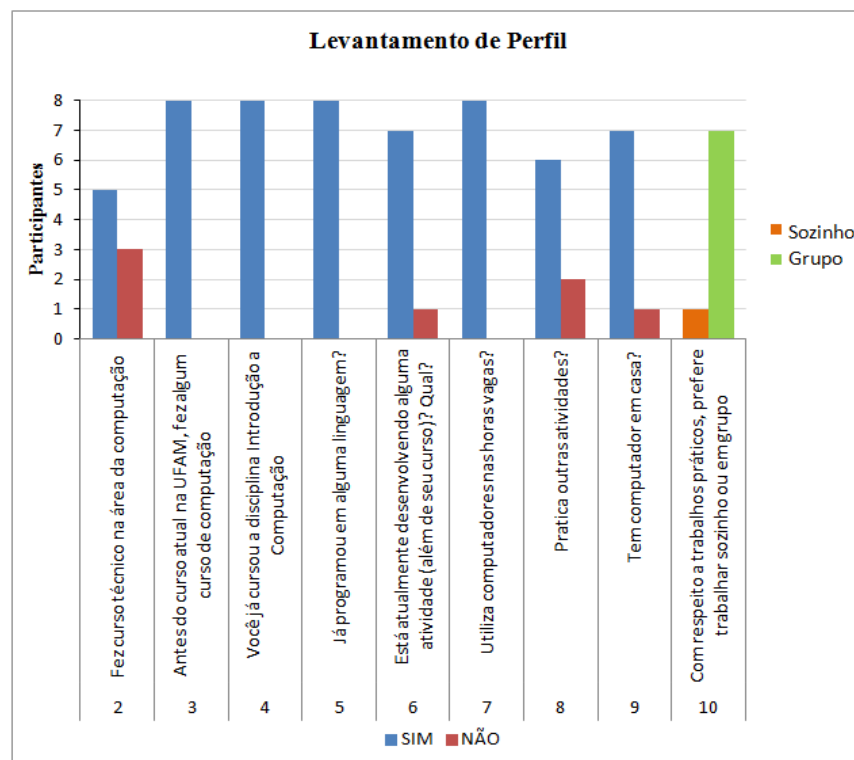


Figura 17. Levantamento de Perfil
Fonte: Elaborado pela autora, 2014.

Destacamos a questão 10 do Levantamento de Perfil, a fim de chamar atenção para o integrante G atribuído como preferência realizar trabalho prático sozinho. Isso se confirmou durante a execução do experimento. Nesse caso, foi uma oportunidade para o avaliador, no papel de professor, intervir no processo. Por ter a característica por preferir trabalhar sozinho, o integrante passou boa parte do tempo no primeiro enigma, enquanto os outros participantes estavam iniciando o terceiro enigma. Então, foi solicitado que o integrante B contribuísse para que seu parceiro de equipe desse andamento no jogo.

A seguir iniciamos a análise do questionário de Avaliação do Conhecimento com questões cujo conteúdo aborda conceitos de algoritmos. Conforme observamos na Figura 18.

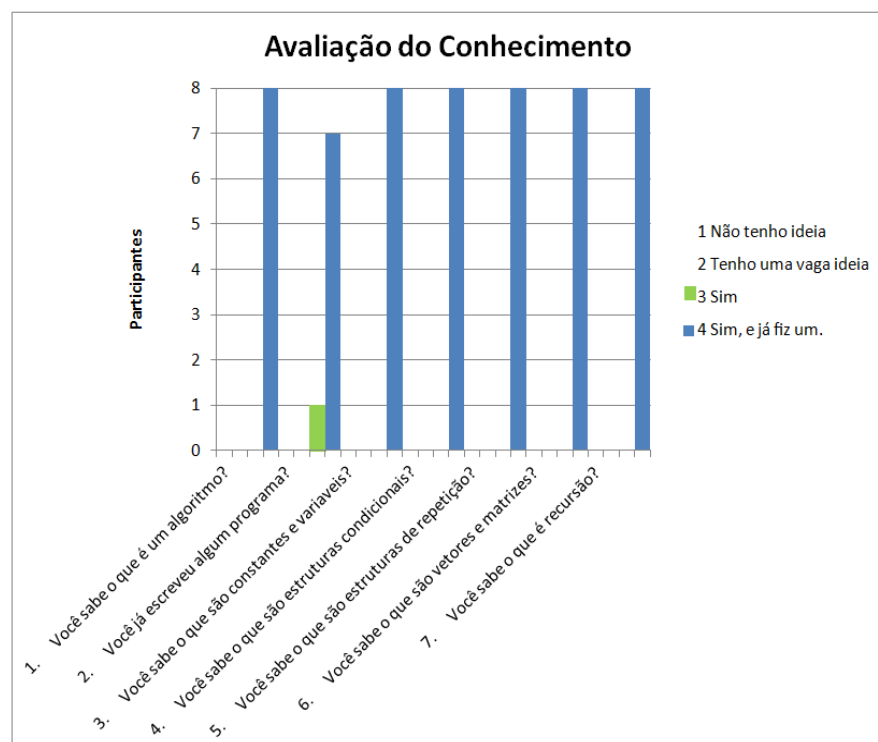


Figura 18. Avaliação do Conhecimento
 Fonte: Elaborado pela autora, 2014.

Na sequência iniciamos o exame dos questionários com Perspectiva Pessoal Pós-Jogo. Este questionário foi respondido pelo avaliador Professor

antes da execução do experimento, pois o objetivo é comparar a visão do Professor com a visão dos participantes, com relação aos conceitos de algoritmos abordados durante o jogo. Para exemplificar observe na Figura 19 as questões 4 e 5 na visão do Professor em comparação com a visão dos integrantes A e B.

| Questões | Perspectiva do Avaliador | A | B |
|----------------------------------------------------------------------|--------------------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4) Em alguma etapa do jogo você identificou o uso de matriz? Qual? | não | não | Sim, na questão de composição de palavras, como: ontem, hoje, amanhã, depois de amanhã |
| 5) Em alguma etapa do jogo você identificou o uso de recursão? Qual? | não | não | sim. Na questão da sequência numérica 2, 10, 12, 16, 17, 18 e 19 onde foi necessário fragmentar o problema maior em partes menores, iterar e retornar respondendo, o próximo número, no caso 200, para compor a resposta |

Figura 19. Comparativo de Perspectivas
Fonte: Elaborado pela autora, 2014.

Por fim, temos a utilização de uma planilha fornecida pelos autores do modelo Questionário para Avaliação de Jogos Educacionais viabilizou a análise qualitativa e quantitativa dos resultados.

Ao analisar a Figura 20 observamos que o experimento resultou em um efeito positivo nos alunos, pois de acordo com as frequências, vários itens dentro do grupo de Motivação receberam nota entre +1 e +2. O requisito Atenção, desse mesmo grupo também teve um bom resultado. Dentre os dois tópicos abordados houve somente uma atribuição a -1 e duas atribuições ao item Indiferente, cujo valor é zero.

Para o requisito Relevância também vale ressaltar o efeito positivo, pois diz respeito ao conteúdo da disciplina, e, como vimos no resultado do questionário de Avaliação do Conhecimento, todos os participantes já possuem familiaridade e prática com os conceitos.

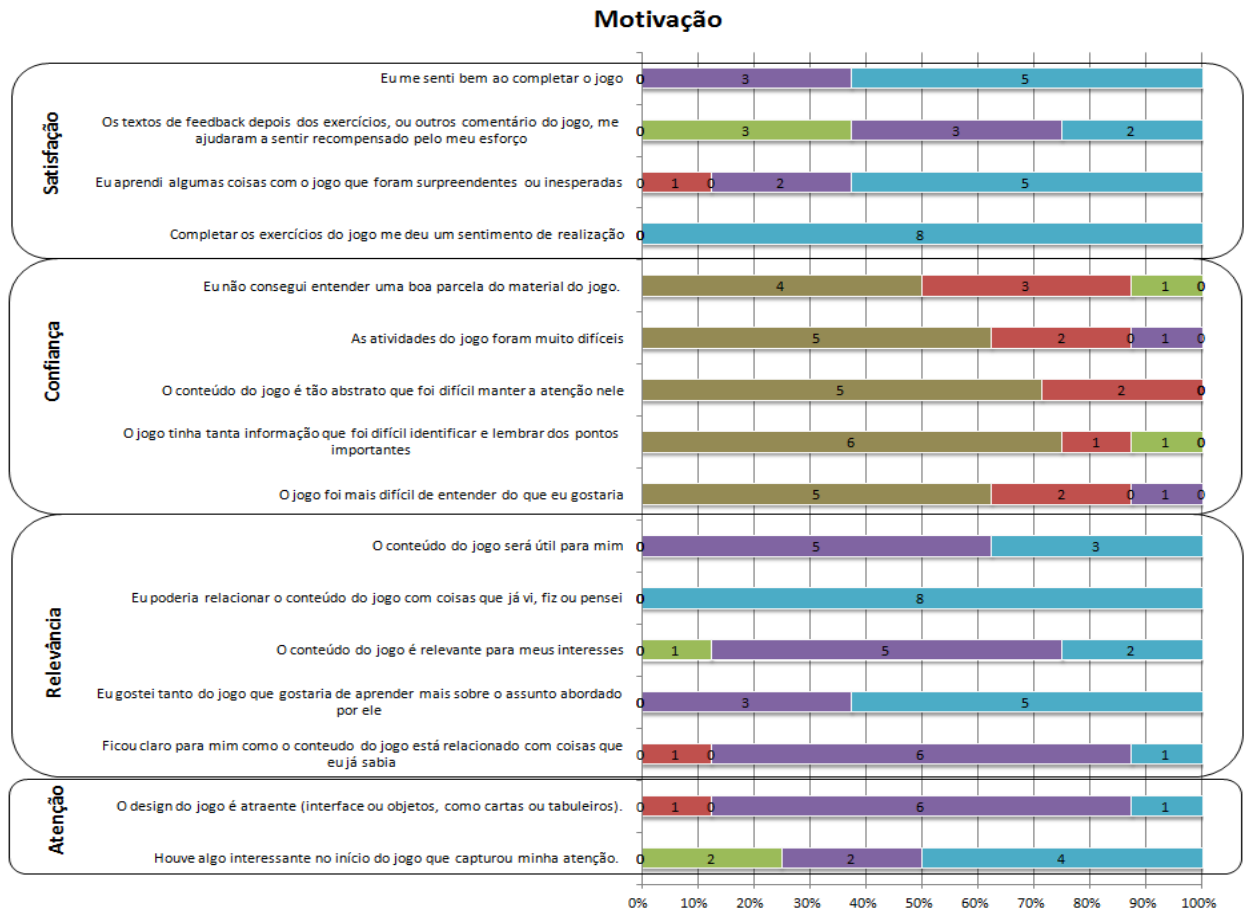
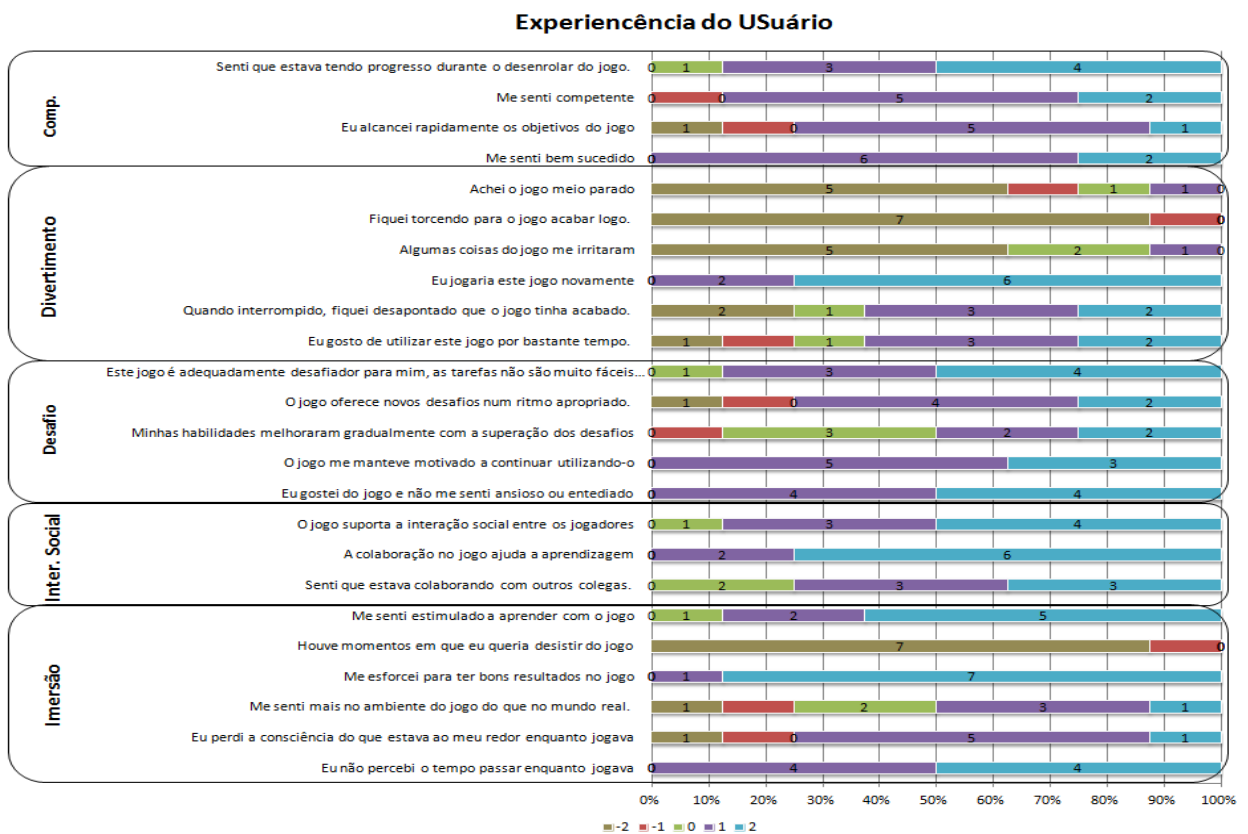


Figura 20. Gráfico de Avaliação do Grupo Motivação
 Fonte: Elaborado pela autora, baseado em Savi, (2014).

Na

Figura 21 temos o resultado do grupo Experiência do Usuário. Nesse grupo os itens que mais destacaram durante o jogo foi o requisito Divertimento. Mesmo sendo uma atividade descontraída, os alunos não perderam a concentração em



resolver os enigmas, e dessa forma exercitando os conceitos.

Figura 21. Gráfico de Avaliação do Grupo Experiência do Usuário
 Fonte: Elaborado pela autora, baseado em Savi, (2014).

E na Figura 22 visualizamos o resultado do grupo Aprendizagem. Nesse caso praticamente todos os participantes colocaram como uma influencia positiva para aprendizagem a utilização do jogo.

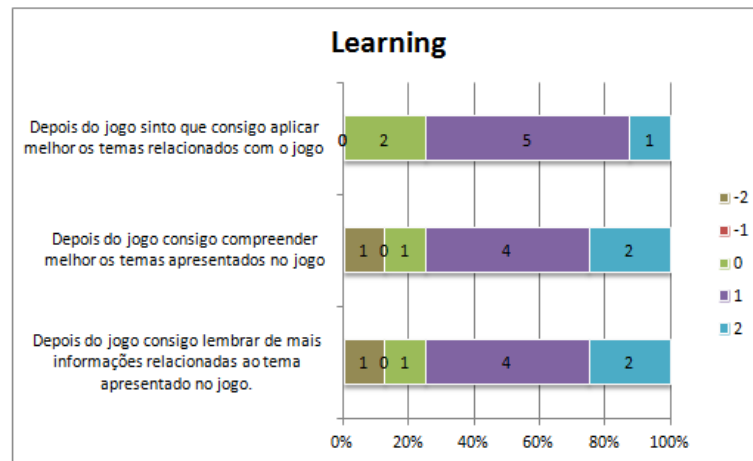


Figura 22. Gráfico de Avaliação do Grupo Aprendizagem
 Fonte: Elaborado pela autora, baseado em Savi, (2014).

Apesar do jogo Caça ao Tesouro está somente em uma versão de protótipo foi possível realizar todos os passos do experimento, pois seu objetivo está em explorar os conceitos e estruturas iniciais de algoritmos com pistas mapeadas com foco em colaboração, incentivando os alunos a trabalharem o raciocínio lógico.

Isso ficou bem visível no decorrer do jogo. Os alunos procuraram várias formas para resolver os enigmas, sozinhos e em grupo. Podemos inferir que a forma como o jogo e aula foi estrutura contribuíram para motivação dos alunos. Observamos que no papel de professor dos cursos introdutórios, o jogo foi uma forma do mesmo visualizar se o que foi planejado para disciplina está implicando em resultados ou é necessário modificar uma atividade ou intervir sempre que identificar alguma dificuldade que pareça intransponível para os grupos. Conseguimos notar essa situação na intervenção do participante cuja preferência é trabalhar sozinho.

No que diz respeito a análise do questionário de Avaliação para Jogos Educacionais, apesar de ter sido utilizado apenas em seu primeiro nível, forneceu dados importantes para acrescentarmos modificações no jogo, como por exemplo, a conseguir visualizar dicas e reduzir a quantidade de chances que o aluno tem para escolha da resposta, motivando assim a busca pela resposta em equipe e; que seja desenvolvido uma maneira que o aluno consiga visualizar sequência do nível de dificuldade dos enigmas .

4.4.3 Terceiro Estudo de Caso Exploratório

Dando continuidade na proposta do jogo Caça ao Tesouro, resolvemos aplicar com turmas que não tinham experiência com a disciplina Introdução a Computação, pois gostaríamos de identificar se a proposta de verificar se a proposta do jogo em trabalhar com o raciocínio lógico realmente influencia na aprendizagem. Para isso fizemos uma segunda experiência. Abaixo segue os tópicos com os resultados.

a. Participantes

Tal estudo foi realizado com 76 alunos do Instituto distribuídos em 3 turmas, com idades de 18 e 50 anos. Os cursos foram: Matemática, Estatística e Gás e Petróleo. A disciplina selecionada foi Introdução à Computação. A turma de Matemática contou com 20 alunos. Na turma de Estatística, também tomaram parte 20 alunos. Já na turma de Gás e Petróleo a presença foi de 36 alunos.

Os critérios para a seleção da amostra da população estudada dependeram de alguns fatores como: voluntários que já cursaram as disciplinas iniciais do curso de Ciência da Computação e áreas afins e interesse por parte destes em participar da pesquisa. Para identificação dos integrantes durante a análise será atribuído uma sequência alfabética (A, B, C, D, E, F, G, H).

A turma de Matemática contou com os participantes de A – T e sua referência será T1. Para a turma de Estatística também teremos os participantes de A – T, mas sua referência será T2. E os integrantes da turma de Gás e Petróleo de A – AJ, cuja identificação será T3.

b. Material

Para coleta de dados durante o experimento utilizamos os seguintes instrumentos: Questionário para Levantamento de Perfil, Questionário de Avaliação de Conhecimento, Questionário para Perspectiva Pessoal Pós-Jogo e o Questionário para Avaliação de Jogos Educacionais. O experimento contou também com a criação de um roteiro do jogo.

Para conhecermos o perfil dos participantes o questionário procurou abordar tópicos como: já cursou a disciplina Introdução a Computação, cursou algum curso técnico durante o ensino médio, como utilizam o computador nas horas vagas, se utiliza jogos, se já programou em alguma linguagem dentre outros.

O questionário para avaliar o conhecimento buscou identificar se os participantes tinham conhecimento com relação ao conteúdo da disciplina Introdução a Computação, mais especificamente nos conceitos e estruturas iniciais. Por isso, os itens abordados foram: Você sabe o que é um algoritmo? Você já escreveu algum programa? Você sabe o que são constantes e variáveis? Você sabe o que são estruturas condicionais? Você sabe o que são estruturas de repetição? Você sabe o que são vetores e matrizes? Você sabe o que é recursão? Como Você avalia o seu conhecimento em “Algoritmos e programação de computadores”?

No questionário para Perspectiva Pessoal Pós-Jogo foi solicitado que os alunos descrevessem sua visão após a participação no jogo. Nesse questionário os alunos deveriam informar se identificaram durante o uso os conceitos e estruturas iniciais de algoritmos. Essas questões foram previamente respondidas pelo avaliador do jogo para fins de comparação, ou seja, se os alunos conseguiram perceber esses itens conforme a visão do avaliador. Outro ponto que solicitado foi com relação à colaboração. Os alunos também deveriam descrever se identificaram durante o experimento a característica de seu um jogo colaborativo.

Já o questionário para Avaliação de Jogos Educacionais são apresentados na forma de afirmações para os alunos indicarem o quanto concordam com elas de acordo com uma escala *Likert* de 7 pontos variando de “discordo fortemente” até “concordo fortemente”, em uma escala que vai de -2 até +2.

c. Procedimento

Em cada uma das turmas o experimento ocorreu na mesma sequência de passos. O trabalho teve início com uma reunião para orientação e esclarecimentos do experimento, todos os participantes assinaram um Termo de Consentimento. Após autorizações, foi solicitado aos participantes o preenchimento do questionário de perfil e na sequência o questionário de avaliação do conhecimento. A partir desses passos os alunos foram divididos em equipes de 4 integrantes, para atender a estrutura do jogo. A divisão das equipes foi feita pelo avaliador que fez o papel de Professor, já que a proposta era simular uma atividade em sala de aula.

Para os testes com o jogo foi utilizado o método de prototipagem. A ferramenta *Moqups* foi utilizada para implementação de um protótipo funcional, ou seja, possui algumas funcionalidades e permite uma interação simulada destas funcionalidades. Com isso foi possível aos participantes realizarem os testes em laboratório, onde cada aluno ficou em um computador. Um segundo avaliador fez o papel das funcionalidades que não estavam implementadas, ou seja, ele era o Sistema.

No início do jogo, o Avaliador que fez o papel do Sistema distribuiu uma cor para cada integrante. A partir disso cada integrante começou a trabalhar com seu primeiro enigma. Quando eles resolviam a questão, a resposta devia ser dita para o Avaliador Sistema, pois ele estava com o gabarito das questões. Caso o raciocínio estivesse correto o participante poderia passar para o próximo enigma, senão ele continuava na mesma tela. Cabe ressaltar que os enigmas escolhidos para o teste estavam entre nível fácil e médio. Somente o último enigma era considerado nível difícil.

Sendo a proposta do jogo colaborativa, os integrantes das equipes podiam trocar ideias e apoio entre si, bem como o avaliador no papel do professor podia dar orientações no sentido de auxiliar na estrutura do raciocínio do aluno, mas não poderia fornecer respostas. As equipes atenderam ao procedimento do jogo, e, ao chegarem à resolução do quarto enigma todos compreenderam que

somente poderiam passar para o último enigma se os 4 membros tivessem resolvidos suas respectivas questões.

Durante a execução do jogo, o avaliador no papel de professor buscou tomar nota de algumas ações dos alunos para análise complementar posterior. A turma de Matemática finalizou o teste no tempo de 19 minutos. A segunda turma fechou os trabalhos em 16 minutos. Já a turma de Gás e Petróleo encerrou os desafios em 13 minutos.

d. Análise dos Resultados

Iniciamos a análise dos dados com a verificação do questionário de Levantamento de Perfil das três turmas. Destacamos a questão 10 do Levantamento de Perfil, a fim de chamar atenção para a característica colaborativa do jogo.

Na turma T1, dos 20 participantes, 04 marcaram a preferência por trabalhar sozinho. Entretanto, durante o experimento não houve interferência. Da mesma forma, 04 participantes da turma T2 também marcaram a preferência por trabalhar sozinho. Com relação a turma T3 notamos que essa opção foi mais acentuada. Dos 36 participantes, 16 optaram por trabalhar sozinho. Observe o gráfico geral na Figura 23.

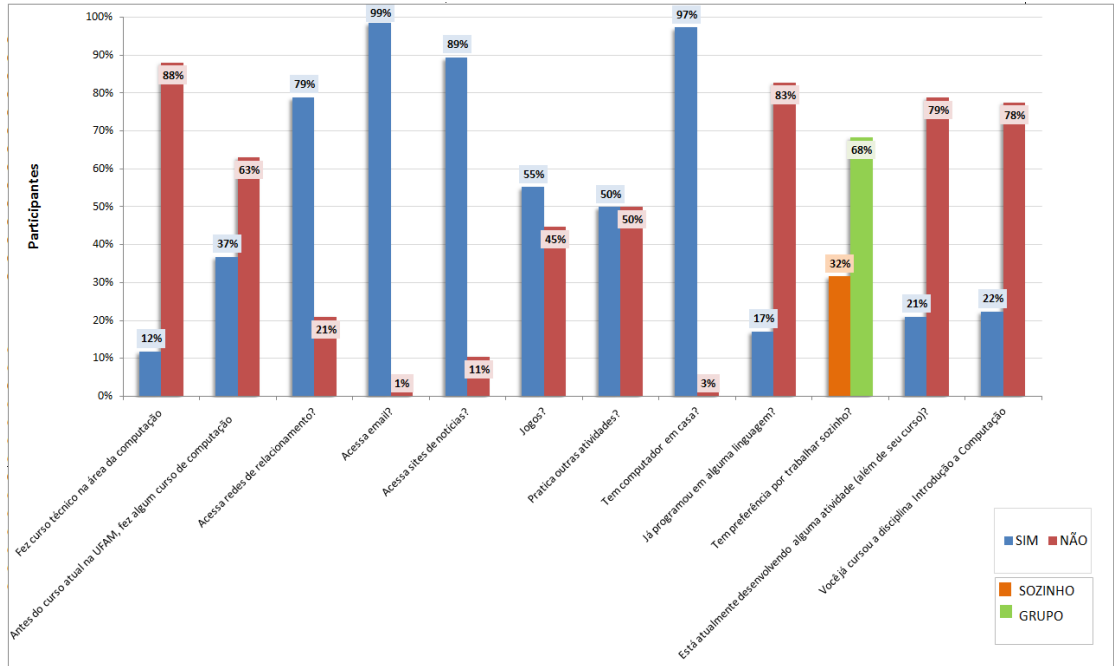


Figura 23. Levantamento de Perfil dos Participantes
 Fonte: Elaborado pela autora, (2014).

A seguir iniciamos a análise do questionário de Avaliação do Conhecimento com questões cujo conteúdo aborda conceitos de algoritmos. Na Figura 24 observamos o resultado geral das turmas T1, T2 e T3.

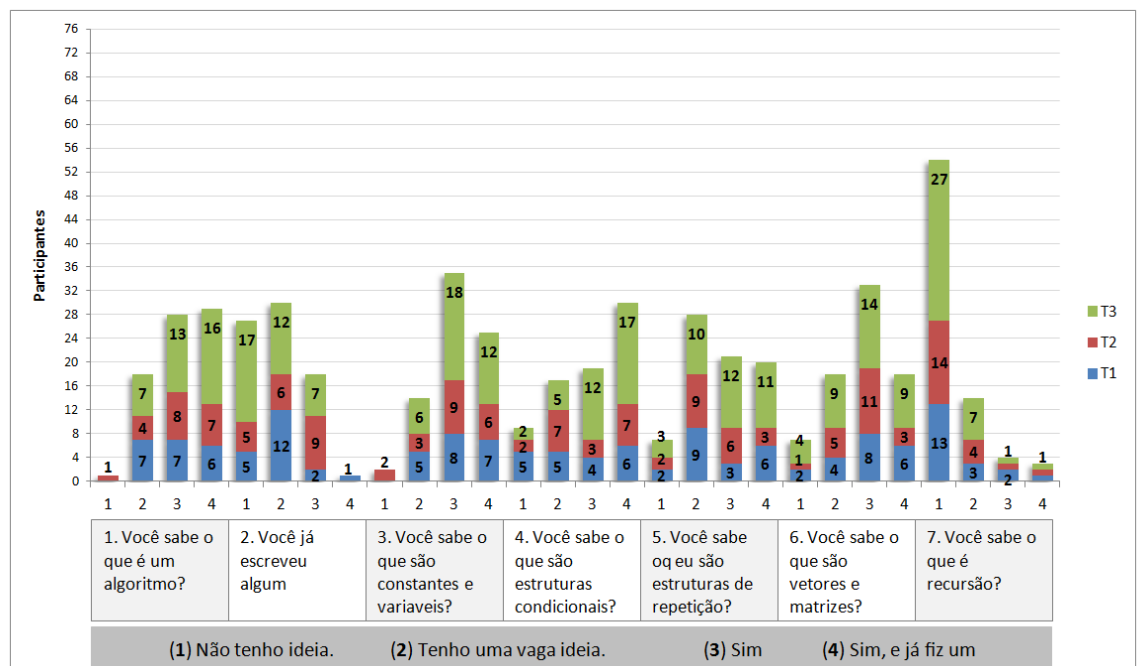


Figura 24. Avaliação de Conhecimento dos Participantes
 Fonte: Elaborado pela autora, 2014.

Após a execução do jogo, iniciamos o exame dos questionários com Perspectiva Pessoal Pós-Jogo. Este questionário foi respondido pelo avaliador Professor antes da execução do experimento, pois o objetivo é comparar a visão do Professor com a visão dos participantes, com relação aos conceitos de algoritmos abordados durante o jogo, bem como ao perfil colaborativo do jogo. Para exemplificar observe na Figura 25, as questões 6 e 7 na visão do Professor em comparação com a visão colaborativa dos integrantes B, F, O e S da turma T1. Essas opiniões foram destacadas, pois os 4 integrantes selecionaram, por preferência, trabalhar sozinho.

| Participante | 6. Do ponto de vista colaborativo, em alguma area do jogo voce identificou características que contribuíram para o trabalho em grupo? | 7. Do ponto de vista colaborativo, em qual momento do jogo você visualizou o papel do professor? |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| Gabarito | Duas oportunidades: - É livre para comunicação entre os membros da equipe. - O último desafio somente pode ser resolvido se todos chegaram no 4o enigma | Duas oportunidades: - Orientação com relação ao uso do jogo Orientação com relação a resolução dos enigmas |
| B | sim | em algumas etapas o professor se mostrou prestativo e atencioso |
| F | sim | sim, pois deu uma pequena introdução aos recursos que foram utilizados hoje |
| O | nenhum | nenhum |
| S | sim, pois quando um terminava podia ajudar o outro e na questao final | na hora de auxiliar os alunos em certas dificuldades e apoiando-os |

Figura 25. Instantâneo de Respostas de participantes cuja preferência é trabalhar sozinho - Questionário Pós-Jogo Turma de Matemática (T1)
Fonte: Elaborado pela autora, 2014.

Agora, observe na Figura 26 as respostas dadas a mesma pergunta. Porém, são de integrantes cuja preferência é trabalhar em grupo.

| | | |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| Participante | 6. Do ponto de vista colaborativo, em alguma area do jogo voce identificou características que contribuíram para o trabalho em grupo? | 7. Do ponto de vista colaborativo, em qual momento do jogo você visualizou o papel do professor? |
| Gabarito | Dois oportunidades: - É livre para comunicação entre os membros da equipe. - O último desafio somente pode ser resolvido se todos chegaram no 4o enigma | Dois oportunidades: - Orientação com relação ao uso do jogo Orientação com relação a resolução dos enigmas |
| C | sim, as pessoas que nos ajudaram a tirar duvidas | ajudando e filmando |
| D | sim, o tempo limitado e a divisão de tarefa pressionam a equipe em se ajudar | monitorou as atividades explicando as regras e se responsabilizar pela equipe de apoio aos alunos |
| E | Sim, quando não podia responder pois havia componentes atrasados no enigma | Sim, pois houve dicas em enigma que ocorreu duvidas |
| I | sim, alguns terminaram primeiro e foram ajudar quem ainda não tinha terminado | nas dificuldades encontradas ao longo do jogo. E nas perguntas feitas no próprio jogo |

Figura 26. Instantâneo de Respostas de participantes cuja preferência é trabalho em grupo - Questionário Pós-Jogo Turma de Matemática (T1)
Fonte: Elaborado pela autora, 2014.

Por fim, temos a utilização de uma planilha fornecida pelos autores do modelo Questionário para Avaliação de Jogos Educacionais viabilizou a análise qualitativa e quantitativa dos resultados.

Ao analisar a Figura 27 observamos que o experimento resultou em um efeito positivo nos alunos, pois de acordo com as frequências, vários itens dentro do grupo de Motivação receberam nota entre +1 e +2. O requisito Atenção, desse mesmo grupo também teve um bom resultado. Dentre os dois tópicos abordados houve somente uma atribuição a -1 e duas atribuições ao item Indiferente, cujo valor é zero.

Para o requisito Relevância também vale ressaltar o efeito positivo, pois diz respeito ao conteúdo da disciplina, e, como vimos no resultado do questionário de Avaliação do Conhecimento, todos os participantes já possuem familiaridade e prática com os conceitos.

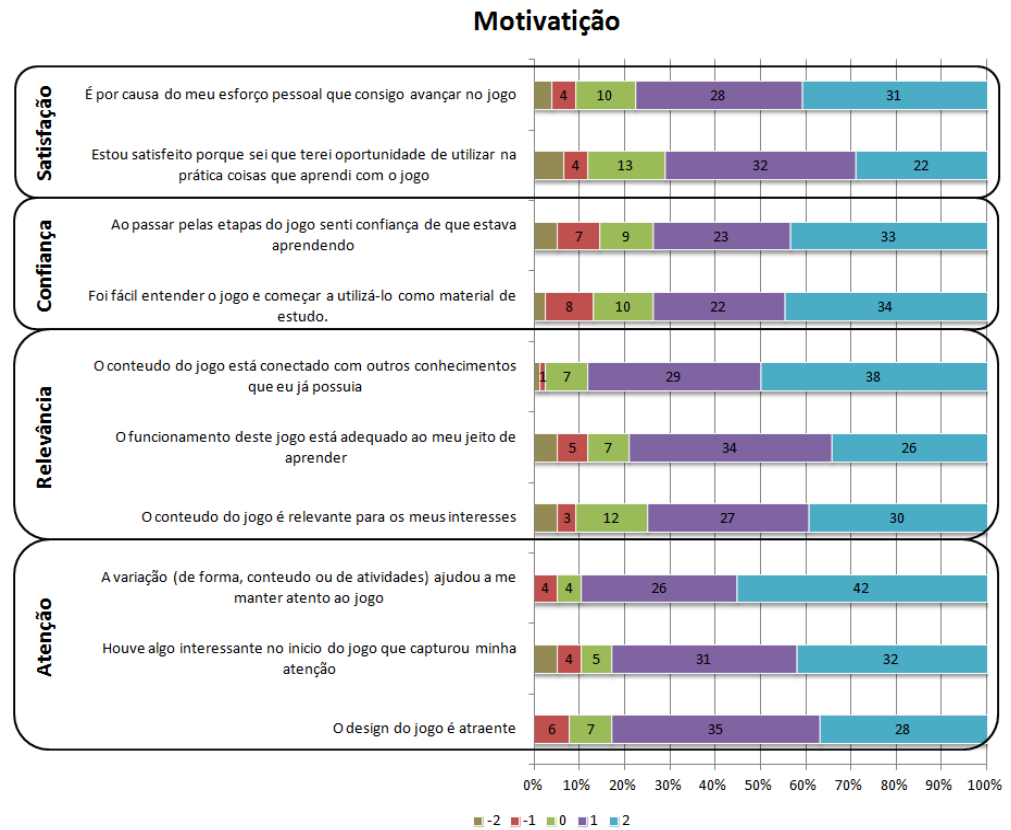


Figura 27. Grupo de Motivação

Fonte: Elaborado pela autora, baseado em Savi, (2014).

Na Figura 28 temos o resultado do grupo Experiência do Usuário. Nesse grupo os itens que mais destacaram durante o jogo foi o requisito Divertimento, com aprovação de praticamente 80% dos alunos. Mesmo sendo uma atividade descontraída, os alunos não perderam a concentração em resolver os enigmas, e dessa forma exercitando os conceitos.

Experiência do Usuário

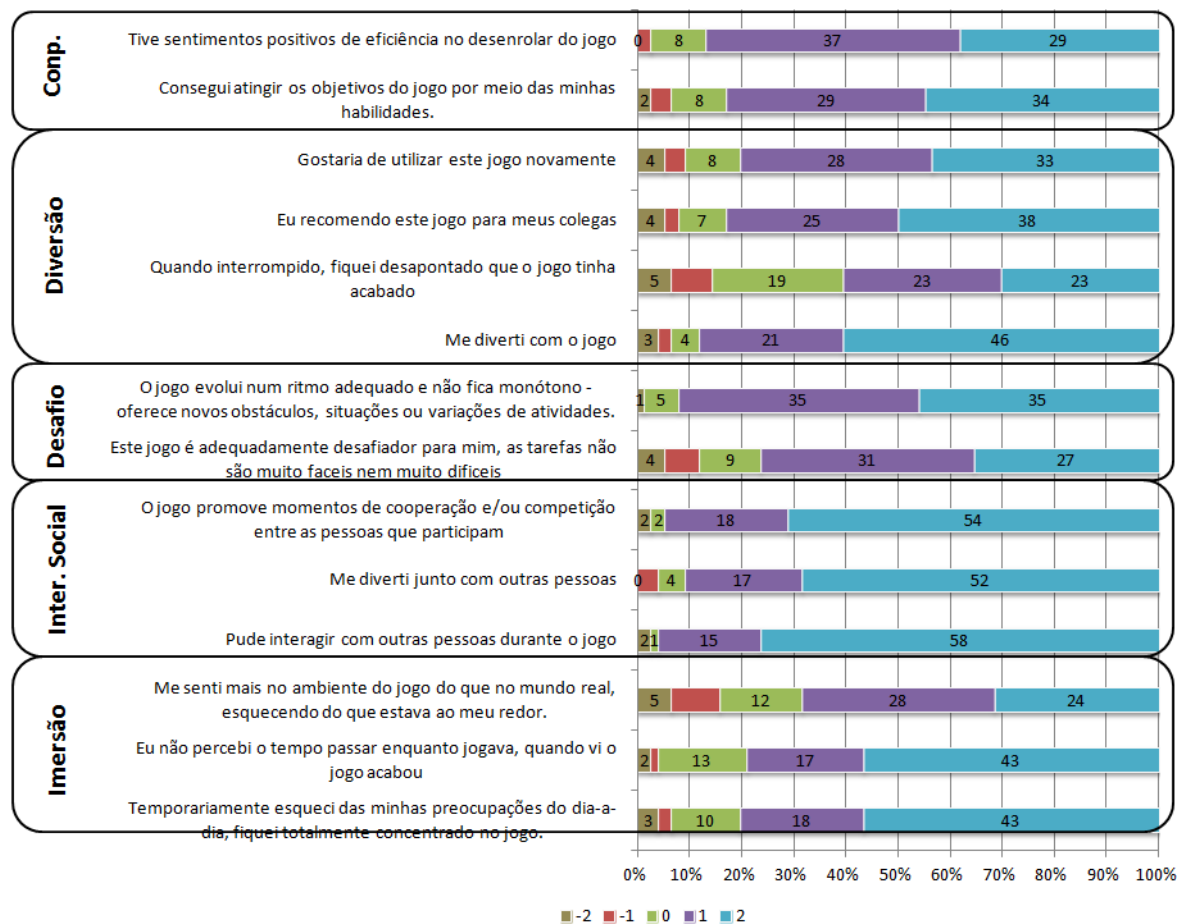


Figura 28. Grupo de Experiência do Usuário
 Fonte: Elaborado pela autora, baseado em Savi, (2014).

E na Figura 29 visualizamos o resultado do grupo Aprendizagem. Na percepção dos alunos o jogo contribuiu para a aprendizagem na disciplina. Podemos observar essa afirmativa na dimensão “O jogo contribuiu para a minha aprendizagem na disciplina” onde 60% dos alunos colocaram a pontuação entre +1 e +2. Esse destaque de porcentagem também pode ser observado nas outras duas dimensões.

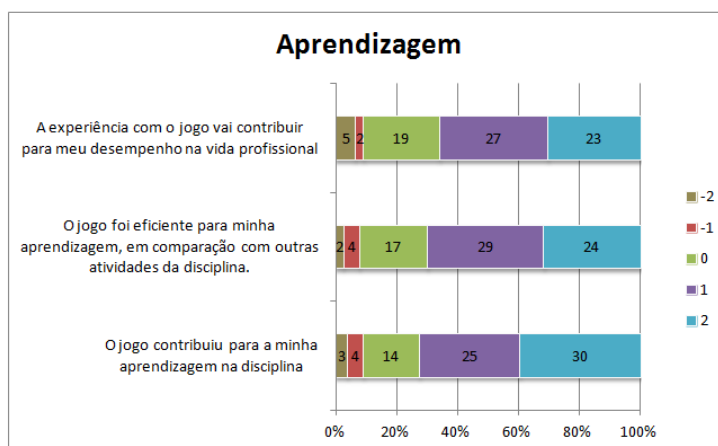


Figura 29. Grupo de Aprendizagem

Fonte: Elaborado pela autora, baseado em Savi, (2014).

Apesar do jogo Caça ao Tesouro está somente em uma versão de protótipo foi possível realizar todos os passos do experimento, pois seu objetivo está em explorar os conceitos e estruturas iniciais de algoritmos com pistas mapeadas com foco em colaboração, incentivando os alunos a trabalharem o raciocínio lógico.

Isso ficou bem visível no decorrer do jogo. Os alunos procuraram várias formas para resolver os enigmas, sozinhos e em grupo. Podemos inferir que a forma como o jogo e aula foi estrutura contribuíram para motivação dos alunos.

Observamos que no papel de professor dos cursos introdutórios, o jogo foi uma forma do mesmo visualizar se o que foi planejado para disciplina está implicando em resultados ou é necessário modificar uma atividade ou intervir sempre que identificar alguma dificuldade que pareça intransponível para os grupos. Conseguimos notar essa situação na intervenção do participante cuja preferência é trabalhar sozinho.

No que diz respeito a análise do questionário de Avaliação para Jogos Educacionais, apesar de ter sido utilizado apenas em seu primeiro nível, forneceu dados importantes para acrescentarmos modificações no jogo, como por exemplo, a conseguir visualizar dicas e reduzir a quantidade de chances que o aluno tem para escolha da resposta, motivando assim a busca pela resposta em equipe e; que seja desenvolvido uma maneira que o aluno consiga visualizar sequência do nível de dificuldade dos enigmas.

5 Conclusão

Esta dissertação descreve uma abordagem visual para aprendizagem colaborativa de programação estimulando as habilidades de resolução de algoritmos em um ambiente visual em conexão com um ambiente 3D, utilizando-se como estratégia central de colaboração as etapas iniciais de um esquema progressivo para aprendizagem de programação em grupo.

Esse esquema foi posto em prática em 2008 com uma turma real de alunos iniciante em Ciência da Computação e Engenharia da Computação. A necessidade surgiu como um modo de introduzir a colaboração nas práticas dos alunos de forma gradativa. A partir dessa experiência o esquema progressivo se mostrou eficaz para a introdução de práticas de colaboração à programação. Porém, surgiu a necessidade de se aplicar essa sistematização utilizando outro paradigma não imperativo.

Assim, iniciamos essa investigação realizando entrevistas com os coordenadores dos cursos de Ciência da Computação (CC) e Engenharia da Computação (EC) da Universidade Federal do Amazonas foi realizado um levantamento de notas e situação acadêmica dos últimos seis anos (2007- 2012). Com uma análise inicial foi observada a situação de reprovação de 41% dos alunos de Ciência da Computação e 40% dos alunos de Engenharia da Computação. Sendo que a reprovação por frequência nesses casos foi de 23% para o curso de CC e 25% para o curso de EC.

Em paralelo a essa linha de investigação também iniciamos a pesquisa bibliográfica, onde podemos ratificar por meio de vários estudos, como a aprendizagem de programação para alunos iniciantes é difícil pela necessidade de desenvolver raciocínio lógico, desenvolver processo de resoluções de problema e dificuldades em compreender abstrações. Além disso, a utilização de modelos tradicionais de aulas práticas dificulta o acompanhamento e a avaliação de todos os exercícios de todos os alunos em curto prazo. Ao trabalhar com muitos alunos ao mesmo tempo, o

professor geralmente mostra somente uma forma de resolver um determinado problema para todo o grupo, não podendo explorar e discutir outras formas de implementação possíveis.

Assim, torna-se imprescindível adotarmos estratégias que melhor se enquadre nessa situação de forma a prover os requisitos necessários para aguçar os alunos no processo de aprendizado, no sentido de que eles possam ser estimulados na comunicação e na busca pelo conhecimento. No entanto, para isso é necessário que exista um ambiente dotado de recursos em que o professor possa exercer a integração, o controle de atividades, bem como servir de mediador nesse novo espaço.

A investigação aqui relatada envolveu um Estudo Piloto realizada com a turma da disciplina Introdução a Computação do Curso de Ciência da Computação da UFAM. Definimos uma abordagem diferenciada utilizando aprendizagem colaborativa baseada no uso de um ambiente de programação visual e sua conexão com um ambiente 3D numa abordagem progressiva na complexidade (i) dos problemas, (ii) da plataforma (2D para 3D), e (iii) das articulações (atuação individual para grupos).

A implantação de quatro fases do esquema progressivo para aprendizagem de programação corroborou os resultados positivos, inclusive no que diz respeito ao acompanhamento e identificação de oportunidades de intervenção para auxílio aos grupos.

5.1 Contribuição

Visualizamos que as principais contribuições desta dissertação foram:

- Identificação de pontos positivos como percepção da programação de forma mais concreta – visual, de manipulação direta e posteriormente em realidade virtual.
- Facilidade para decidir sobre os produtos gerados (escolha da solução), pois a visualização favoreceu essa escolha. Essa característica visual foi

observada tanto no Scratch (2D) como no Opensimulator (3D), ambas ferramentas visuais.

- O benefício da interação que faz uma referência direta ao contexto de colaboração, pois a forma como foi distribuído o trabalho em equipe incentivou a comunicação entre os membros.
- Impacto positivo na motivação, possivelmente pelo tema dos desafios que envolviam estratégias de jogos.

5.2 Trabalhos Futuros

Como sugestões de trabalhos futuros, podemos descrever:

- A necessidade de uma avaliação contínua

A disciplina ministrada após a disciplina de IC, Estrutura de Dados I, também foi objeto de observação. Essa disciplina compõe parte da grade do curso que abrange o contexto de programação, e ao seu término foi realizada dois tipos de entrevistas. Uma com o professor da disciplina e outra com os alunos. O objetivo dessas entrevistas era verificar se em algum ponto a experiência vivenciada em IC contribuiu para Estrutura de Dados I.

A entrevista com o professor da disciplina procurou conhecer como a mesma foi planejada e executada. Essas informações irão contribuir para as novas tomadas de decisão referente ao projeto, onde observamos a necessidade de uma forma de avaliação contínua para os alunos. Isso se deve ao fato de que as ferramentas utilizadas na abordagem visual possuem um limite no contexto da programação. É necessário subir um

novo degrau, novas estruturas com complexidades que a ferramenta *Scratch* não suporta, como por exemplo, as estrutura de dados.

Essa visão pode ser reforçada com o estudo realizado com a turma de matemática durante a disciplina de Introdução a Computação. Mesmo sendo uma turma do curso de matemática, eles conseguiram visualizar que a ferramenta *Scratch* não seria o suficiente para compor todos os recursos que ele projetaram para suas aplicações.

- A utilização de outras ferramentas visuais colaborativas

Apesar dos benefícios no sentido de visualização, imersão e monitoramento, os mundos virtuais apresentam desafios tecnológicos que podem retardar sua implantação e aplicação. A arquitetura de um ambiente assim é cliente-servidor. No entanto, não existe um padrão, ou seja, diferentes mundos virtuais usam diferentes hardwares e softwares.

O desempenho da rede e a placa gráfica adotada também são fatores críticos. Se não temos uma rede adequada é necessário reduzir a quantidade de acessos bem como os dados que precisam trafegar. Além disso, também é necessário termos placas de vídeos que suportem resoluções maiores, do contrário o ambiente fica com retardo na visualização.

Uma proposta é utilizar ferramentas que possam ser implantadas em celulares, visto que atualmente esse é um dispositivo comum aos alunos e aos professores.

- Aprofundar no tema de desafios com contextos de jogos colaborativos

Os desafios que possuíam contexto de jogos foram os que mais chamaram atenção dos alunos. Os jogos podem estimular a criatividade e também podem ser usados como uma ponte para os aspectos abstratos da

programação. Por isso, a proposta é criar um ambiente que possa incentivar o raciocínio lógico para a resolução de problemas utilizando esses desafios.

Indicativos positivos foram constatados com os estudos realizados com o protótipo do jogo Caça ao Tesouro. Podemos observar que o fator motivacional se destacou nesses estudos, pois os alunos trabalhando em equipes em um contexto de desafios geraram indícios para continuarmos atacando essa linha.

5.3 Publicações de Resultados Parciais

A seguir listamos os relatos de resultados parciais deste trabalho.

- **Artigo resumido**
 - ***Uma abordagem para apoio a aprendizagem de lógica de programação em cursos de graduação utilizando mundos virtuais***
 - Workshop de Teses e Dissertações em Sistemas Colaborativos
 - Simpósio Brasileiro de Sistemas Colaborativos - 2012
- **Apresentação**
 - ***Uma abordagem para apoio a aprendizagem de lógica de programação em cursos de graduação utilizando mundos virtuais***
 - Wopi – Workshop de Pesquisa em Informática - 2012
- **Artigo resumido**
 - ***Uma Abordagem de Programação Visual para Aprendizagem de Programação em Grupo***
 - Workshop de Teses e Dissertações em Sistemas Colaborativos
 - Simpósio Brasileiro de Sistemas Colaborativos - 2013
- **Artigo completo**
 - ***Aprendizagem Colaborativa de Programação com Scratch e OpenSimulator***
 - Simpósio Brasileiro de Sistemas Colaborativos - 2013
- **Convite participação**
 - ***Apresentação PechaKucha***
 - Slactions 2013 (Conferência de Pesquisa em mundos virtuais).

Referências Bibliográficas

Aureliano, Viviane Cristina Oliveira; Tedesco, Patrícia Cabral de Azevedo Restelli. Ensino-aprendizagem de Programação para Iniciantes: uma Revisão Sistemática da Literatura focada no SBIE e WIE . Centro de Informática – Universidade Federal de Pernambuco (UFPE); Instituto Federal de Pernambuco – Campus Belo Jardim (IFPE). 2012. Disponível em: <<http://www.lbd.dcc.ufmg.br/bdbcomp/servlet/Trabalho?id=16365>>. Acesso em: 14 maio 2013

Altoé, Anair; Penati, Marisa Morales. O Construtivismo e o Construcionismo. Fundamentando a Ação Docente. Universidade Estadual de Maringá. 2005. Disponível em: <<http://www.dtp.uem.br/gepiaepde/constru.pdf>>. Acesso em: 03 julho 2014.

Alves, Daniele Guimarães; Cabral, Tathiane Dutra; Costa, Rosa Maria Esteves M. da. Ambientes Virtuais para Educação a Distância: uma estrutura de classificação e análise de casos. Universidade do Estado do Rio de Janeiro (UERJ). 2003. Disponível em: <<http://www.e-publicacoes.uerj.br/index.php/cadinf/article/viewFile/6616/4731>>. Acesso em: 26 novembro 2011.

Basili, V. R., Rombach, H. D. (1998). “The TAME Project:Towards Improvement – Oriented Software Environments”. IEEE Transactions on Software Engineering, v. 14.

Bento, João José Fernandes Gonçalves. Desenvolvimento e avaliação de um ambiente de aprendizagem 3D. Instituto Politécnico Escola de Bragança – Escola Superior de Educação. 2011. Disponível em: <<http://revistas.unipar.br/educere/article/viewFile/173/147>>. Acesso em: 26 novembro 2011.

Bini Elena Mariele; Koscianski, Andre. O Ensino de Programação de Computadores em um Ambiente Criativo e Motivador. Instituto..... 2009. Disponível em: <<http://posgrad.fae.ufmg.br/posgrad/viiienpec/pdfs/675.pdf>>. Acesso em: 07 julho 2014.

Campos, Dinah Martins de Souza. Psicologia da Aprendizagem. Petrópolis. Editora Vozes Limitada, 1972. Disponível em: <<http://www.cienciasecognicao.org/pdf/v05/m22542.pdf>>. Acesso em: 26 novembro 2011

Castañon, Gustavo Arja. Universidade Estácio de Sá. Construtivismo e ciências humanas. 2005. Disponível em: <<http://www.cienciasecognicao.org/pdf/v05/m22542.pdf>>. Acesso em: 07 julho 2014.

Castro, Thais Helena Chaves de; Castro, Alberto Nogueira de; Menezes, Crediné Silva de; Cury, Davidson. Arquitetura SAAP: Sistema de Apoio à Aprendizagem de Programação. Universidade Federal do Espírito Santo/Universidade Federal do Amazonas. 2002. Disponível em: <http://www.researchgate.net/publication/228782248_Arquitetura_SAAP_Sistema_de_Apoio_Aprendizagem_de_Programao>. Acesso em: 26 novembro 2011.

Castro, Thais Helena Chaves de; Júnior, Alberto Nogueira de Castro; Menezes, Crediné Silva de. Aprende – um Ambiente Cooperativo de Apoio à Aprendizagem de Programação. Universidade Federal do Amazonas/ Universidade Federal do Espírito Santo. 2004. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/308/294>>. Acesso em: 26 novembro 2011

Castro, Thais Helena Chaves de Castro; Júnior, Alberto Nogueira de Castro; Fuks, Hugo. Universidade Católica do rio de Janeiro / Universidade Federal do Amazonas. Aprendendo a Programar em Grupo. 2008. Disponível em: <<http://groupware.les.inf.puc-rio.br/public/papers/2008.SBSC.Castro.Programming.pdf>>. Acesso em: 25 agosto 2011.

Castro, Thais. Fuks, Hugo. Sistematização da Aprendizagem de Programação em Grupo. Instituto de Computação – Universidade Federal do Amazonas (UFAM); Programa de Pós-Graduação em Informática – Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio). 2011.

Disponível em: <<http://groupware.les.inf.puc-rio.br/public/papers/SistematizacaoGPL.pdf>>. Acesso em: 13 maio 2013

Escórcio, Dalila Silvestre. Formação Pedagógica Inicial de Formadores. Direção-Geral do Emprego e das Relações de Trabalho (DGERT). 2010. Disponível em: <<http://www.formate.com/>>. Acesso em: 26 novembro 2011.

Esteves, Maria Micaela Gonçalves Pinto Dinis. Uso do Second Life no Suporte à Aprendizagem Contextualizada de Programação. Universidade de Trás-os-Montes e Alto Douro. Escola de Ciências e Tecnologia. Departamento de Engenharias. 2010. Disponível em: <<http://pt.scribd.com/doc/52575406/Uso-do-Second-Life-no-Suporte-a-Aprendizagem-Contextualizada-de-Programacao>>. Acesso em: 22 novembro 2011.

Falckembach, Gilse A Morgental; Araujo, Fabricio Viero de. Aprendizagem de Algoritmos: Dificuldades na Resolução de Problemas. Universidade Luterana do Brasil/Faculdade Dom Alberto. 2005. Disponível em: <<http://periodicos.unesc.net/index.php/sulcomp/article/view/916/909>>. Acesso em: 26 novembro 2011.

Ferreira, Cláudia; Gonzaga, Flávio; Santos, Rodrigo. Um Estudo sobre a Aprendizagem de Lógica de Programação Utilizando Programação por Demonstração. Faculdade Governador Ozanam Coelho/ Universidade Federal de Alfenas/ Universidade Federal do Rio de Janeiro. 2009. Disponível em: <http://www.inf.pucminas.br/sbc2010/anais/pdf/wei/st06_03.pdf>. Acesso em: 23 setembro 2011.

Freire, Anderson; Rolim, Cledja; Bessa, Wladia. Criação de um ambiente virtual de ensino-aprendizagem usando a plataforma opensimulator. Instituto Federal de Educação, Ciência e Tecnologia de Alagoas. 2008. Disponível em: <<http://connepi.ifal.edu.br/ocs/index.php/connepi/CONNEPI2010/paper/viewFile/684/410>>. Acesso em: 22 novembro 2011.

Garcia, Fabiano Luiz Santos; Camargo, Fabio; Tissiani, Gabriela. Metodologia para criação de ambientes virtuais tridimensionais. Universidade Federal de Santa Catarina. 2001. Disponível em: <http://www.ppgia.pucpr.br/~fabricio/ftp/Aulas/Ciencia_da_Computacao/IHM/MetodologiaAmbientesVirtuais.pdf>. Acesso em: 26 novembro 2011.

Gomes, Alex Sandro; Junior, Elias Vidal Bezerra; Cabral, Mayara Kaynne Fragoso. REDU: Um Ambiente Virtual Colaborativo para Ensino de Programação Orientada a Objeto. Instituto Federal de Educação, Ciência e Tecnologia do Tocantins – IFTO, campus Porto Nacional. 2012. Disponível em: <<http://www.abed.org.br/congresso2012/anais/146c.pdf>>. Acesso em: 07 Julho 2014.

Haguenauer, Cristina; Mussi, Marcus Vinicius; Filho, Francisco Cordeiro. Ambientes Virtuais de Aprendizagem: Definições e Sis. 2009. Disponível em: <<http://www.latec.ufrj.br/revistas/index.php?journal=educaonline&page=article&op=view&path%5B%5D=112&path%5B%5D=298>>. Acesso em: 27 dezembro 2011.

Libaneo, José Carlos. Questões de Metodologia do Ensino Superior – A Teoria Histórico-Cultural da Atividade de Aprendizagem. Pontifícia Universidade Católica do Rio de Janeiro. 2003. Disponível em: <<http://www.inf.puc-rio.br/~francis/2006-Construtivismo.pdf>>. Acesso em: 07 Julho 2014.

Machado, Francis Berenger; Miranda, Luciana Lima. O Uso do Construtivismo e da afetividade nas metodologias de ensino à distância. Pontifícia Universidade Católica de Goiás. 2006. Disponível em: <http://www.ucg.br/site_docente/edu/libaneo/pdf/questoes.pdf>. Acesso em: 07 Julho 2014.

Miranda, Guilhermina Lobato; Bahia, Sara. Teorias da Aprendizagem. Manual de Apoio a Disciplina Psicologia da Aprendizagem. Universidade Católica Portuguesa. Instituto de Educação. Instituto de Ensino e Formação a Distância. 2007. Disponível em: <<https://cdp.wikispaces.com/file/view/ManualPs.pdf/31934693/ManualPs.pdf>>. Acesso em: 03 Julho 2014

Moresi, Eduardo (Organizador). Metodologia da Pesquisa. Universidade Católica de Brasília. 2003. Disponível em: <http://ftp.unisc.br/portal/upload/com_arquivo/1370886616.pdf>. Acesso em: 27 dezembro 2011.

Morgado, Leonel Caseiro; Sousa, Antonio; Barbosa, Luís. Ensaio de Utilização do Mundo Virtual Second Life no Ensino de Programação de Computadores. Universidade de Trás-os-Montes e Alto Douro. 2008. Disponível em: <<http://home.utad.pt/~leonelm/papers/ensaioutilizacaoSecondLifeensinoprogramacao.pdf>>. Acesso em: 21 setembro 2011.

Neto, Bernardo Lobato dos Santos. Proposta de um Laboratório Virtual para Aprendizagem de PROLOG. Universidade Federal do Pará. 2007. Disponível em: <<http://www.portaltcc.icen.ufpa.br/portaltcc/principal/Tcc/action.do;jsessionid=B1899088079A97FB8B03CFF4DB5093C7?act=download&id=106>>. Acesso em: 07 julho 2014.

Neto, Valter dos Santos Mendonça. A Utilização da Ferramenta Scratch como Auxílio na Aprendizagem de Lógica de Programação. Faculdade Pitágoras – São Luiz. 2013. Disponível em: <<http://www.br-ie.org/pub/index.php/wcbie/article/download/2675/2329>>. Acesso em: 07 julho 2014.

Neves, Maria de Fátima; Coelho, Juan Manuel Adan. OntoRevPro: Uma Ontologia sobre Revisão de Programas para o Aprendizado Colaborativo de Programação em Java. Faculdade de Engenharia de Computação, Pontifícia Universidade Católica de Campinas. 2006. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/496/482>>. Acesso em: 07 julho 2014.

Nobre, Isaura Alcina Martins, Menezes, Crediné Silva de. Suporte à Cooperação em um Ambiente de Aprendizagem para Programação (Samba). Universidade Federal do Espírito Santo. 2002. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/195/181>>. Acesso em: 07 julho 2014.

Nunes *et al.*. Diretrizes Curriculares dos cursos de Bacharelado em Ciência da Computação, Engenharia de Computação, Engenharia de Software, Sistemas de Informação e dos cursos de Licenciatura em Computação. Ministério da Educação. Conselho Nacional de Educação. Câmara de Educação Superior. 2003. Disponível em: <http://www.sbc.org.br/index.php?option=com_content&view=article&id=423:diretrizes-curriculares-dos-cursos-de-computacao&catid=65:destaques>. Acesso em: 26 novembro 2011.

Ostermann, Fernanda ; Cavalcanti, Cláudio José de Holanda Cavalcanti. Teorias de Aprendizagem. Texto Introdutório. Universidade Federal do Rio Grande do Sul – Instituto de Física. 2010. Disponível em: <http://www.ufrgs.br/uab/informacoes/publicacoes/materiais-de-fisica-para-educacao-basica/teorias_de_aprendizagem_fisica.pdf>. Acesso em: 03 Julho 2014.

Paula, Leda Queiroz; Paiva, Dilermando Junior; Freitas, Ricardo Luis. A Leitura e a Abstração do Problema no Processo de Formação do Raciocínio Lógico-Abstrato em Alunos de Computação. Faculdade de Tecnologia de Indaiatuba/Pontifícia Universidade Católica de Campinas. 2009. Disponível em: <<http://fatecindaiatuba.edu.br/reverte/index.php/revista/article/view/35/0>>. Acesso em: 18 novembro 2011.

Piaget, J. A Epistemologia Genética. RJ: Vozes, 1971. Tradução: Nathanael C. Caixeiro. L'épistemologie Génétique. Paris: Universitaires de France, 1970.

Pimentel, M., Fuks, H. (Org). Sistemas Colaborativos. Rio de Janeiro - RJ: Elsevier-Campus-SBC, 2011

Polya, George. A arte de resolver problemas: um novo aspecto do método matemático. Rio de Janeiro: Interciência, 1995. Disponível em: <<http://www.mat.ufmg.br/~michel/inicmat2010/livros/polya.pdf>>. Acesso em: 08 novembro 2011.

Pombo, Olga; Mosquito, Elisa; Inácio, Ricardo; Ferreira, Teresa; Cravo, Sara. "Aprender, ensinar e aprender a ensinar" (Tradução). Faculdade de Ciências da Universidade de Lisboa. 2008. Disponível em: <<http://www.educ.fc.ul.pt/docentes/opombo/seminario/polya/traducao.htm>>. Acesso em: 08 novembro 2011.

Rapkiewicz, Clevi Elena; Falkembach, Gilse; Seixas, Louise; Santos, Núbia dos Rosa; Cunha, Vanildes Vieira da; Klemann, Miriam. Estratégias Pedagógicas no Ensino de Algoritmos e Programação Associadas ao Uso de Jogos Educacionais. Universidade Estadual Norte Fluminense Darcy Ribeiro/ Centro Interdisciplinar de Tecnologia Educacional/ Universidade Federal do Rio Grande do Sul/ Fundação de Apoio à Escola Técnica do Estado do Rio de Janeiro/ Colégio Mauá. 2006. Disponível em: <<http://www.lume.ufrgs.br/handle/10183/22862>>. Acesso em: 26 outubro 2011.

Ribeiro, Romenig da Silva; Brandão, Leônidas de O.; Brandão, Anarosa A.F. Uma visão do cenário Nacional do Ensino de algoritmos e Programação: uma proposta baseada no Paradigma de Programação Visual. Universidade de São Paulo. 2012. Disponível em: <<http://www.lbd.dcc.ufmg.br/colecoes/sbie/2012/00127.pdf>>. Acesso em: 13 maio 2013

Sampaio, Augusto; Maranhão, Antonio. Conceitos e Paradigmas de Programação via Projeto de Interpretadores. Universidade Federal de Pernambuco. 2008. Disponível em: <<http://www.cin.ufpe.br/~in1007/transparencias/jai/Jai2008Augusto.pdf>>. Acesso em: 03 Julho 2014

Santos, Rodrigo Pereira dos; Costa, Heitor Augustus Xavier. Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos iniciantes em Computação e Informática. Universidade Federal de Lavras. 2006. Disponível em: <http://ufla.academia.edu/HeitorCosta/Papers/392048/Analise_De_Metodologias_E_Ambientes_De_Ensino_Para_Algoritmos_Estruturas_De_Dados_E_Programacao_Aos_Iniciantes_Em_Computacao_E_Informatica>. Acesso em: 26 novembro 2011.

Savi, Rafael, Wangenheim, Christiane Gresse von, Ulbricht, Vania, Vanzin, Tarcísio (2010). Proposta de um Modelo de Avaliação de Jogos Educacionais. Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento – EGC/UFSC. CINTED-UFRGS Novas Tecnologias na Educação. Disponível em: <http://www.br-ie.org/pub/index.php/sbie/article/view/2502/2161>. Acesso em: 22 abril 2014

Silva, Halysson F. A. , Santos, Otávio L., Rafalski, Jadson P.; Cury, Davidson; Menezes, Crediné S. Clic&Ação: Um Ambiente para a Construção Colaborativa de Micromundos. Universidade Federal do Espírito Santo (UFES). 2012. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/1703/1464>>. Acesso em: 03 Julho 2014.

Skinner, Burrhus Frederic. Sobre o behaviorismo. Trad. Maria da Penha Villalobos. Editora Cultrix, São Paulo - 1974. Disponível em: <http://minhateca.com.br/niltonvarela/Documentos/Ebooks/Freud*2c+Lacan*2c+Jung+e+outros++psicologia+e+psican*c3*a1lise+*5bAK*5d/SKINNER*2c+B.+F.+Sobre+o+behaviorismo,1338902.pdf> . Accessi em: 22 Agosto 2014.

Skinner, Burrhus Frederic (1980). Contingências do Reforço. Uma Análise Teórica. Trad. Rachel Moreno. Coleção os pensadores. São Paulo: Abril Cultural. Disponível em: <http://minhateca.com.br/PabloStuart/Documentos/An*c3*a1lise+do+Comportamento/Skinner*2c+B.+F.+Conting*c3*aancias+do+Refor*c3*a7o++Uma+An*c3*a1lise+Te*c3*b3rica,1985871.pdf> . Accessi em: 30 Setembro 2014.

Souza, Renato Rocha. Algumas considerações sobre as abordagens construtivistas para a utilização de tecnologias na educação. Universidade Federal de Minas Gerais. 2006. Disponível em: <<http://revista.ibict.br/liinc/index.php/liinc/article/viewFile/203/118>>. Acesso em: 03 Julho 2014.

Stahel, Gerry; Koschmann, Timothy ; Suthers, Dan. Traduzido por: Hugo Fuks, Tatiana Escovedo (Português do Brasil). Aprendizagem colaborativa com suporte computacional: Uma perspectiva histórica. Cambridge University Press. 2006. Disponível em: <http://gerrystahl.net/cscl/CSCL_Portuguese.pdf>. Acesso em: 03 Julho 2014

Tonini, Leonardo da Silva. O Modelo OOHDM Aplicado no Ensino da Programação Orientada a Objetos na UFLA. Universidade Federal de Lavras. 2013. Disponível em: <<http://www.bsi.ufla.br/wp-content/uploads/2013/10/O-MODELO-OOHDM-APLICADO-NO-ENSINO-DA-PROGRAMA%C3%87%C3%83O-ORIENTADA-A-OBJETOS-NA-UFLA-.pdf>>. Acesso em: 07 Julho 2014

Valentim, Henryethe. Um Estudo Sobre o Ensino-Aprendizagem de Lógica de Programação. Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Educação Científica e Tecnológica. 2009. Disponível em: <<http://posgrad.fae.ufmg.br/posgrad/viienepec/pdfs/137.pdf>>. Acesso em: 08 novembro 2011.

Vega, Katia Canepa; Pereira, Andreia; Carvalho, Gustavo Robichez de; Raposo, Alberto; Fuks, Hugo. Pontifícia Universidade Católica do Rio de Janeiro. Prototyping Games for Training and Education in Second Life: Time2Play and TREG. Disponível em: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5479113>>. Acesso em: 25 agosto 2011.

WATSON, John B. Clássico traduzido: a psicologia como o behaviorista a vê. Temas psicol., Ribeirão Preto, v. 16, n. 2, 2008. Disponível em <http://pepsic.bvsalud.org/scielo.php?script=sci_arttext&pid=S1413-389X2008000200011&lng=pt&nrm=iso>. Acesso em: 27 agosto 2014.

Yin, Robert K. trad. Daniel Grassi. Estudo de Caso. Planejamento e Métodos. Porto Alegre. Bookman, 2004.

Zhou, Ning-Ning; Deng, Yu-Long. Virtual reality: A state-of-the-art survey. School of Computing, Nanjing University of Posts and Telecommunications / State Key Laboratory of Software Development Environment. 2009. Disponível em: <<http://www.springerlink.com/index/10.1007/s11633-009-0319-9>>. Acesso em: 26 outubro 2011.

APENDICE A: Questionário de Perfil



Poder Executivo
Ministério da Educação
Universidade Federal do Amazonas
Instituto de Computação – IComp



INTRODUÇÃO A COMPUTAÇÃO – 2012/2 – TURMA 1

Professor: Alberto Castro

LEVANTAMENTO DE PERFIS DA TURMA – 02/01/2013

ALUNO(A): _____ Matrícula: _____

- 1) Fez curso técnico na área da computação () Sim () Não
- 2) Antes do curso atual na UFAM, fez algum curso de computação () Sim () Não
- 3) É familiarizado com scripts e outros recursos da Web () Sim () Não
- 4) Utiliza computadores nas horas vagas? () Sim () Não
 - a. Acessando redes de relacionamento () Sim () Não
 - b. Acessando email () Sim () Não
 - c. Acessando sites de notícias () Sim () Não
 - d. Jogando () Sim () Não
 - i. Quais jogos?

e. Outras atividades – Diga quais

- 5) Tem computador em casa? () Sim () Não
- 6) Já programou em alguma linguagem? () Sim () Não
 - a. Qual / durante quanto tempo?

7) Com respeito a trabalhos práticos, prefere trabalhar () sozinho () em grupo

8) Está atualmente desenvolvendo alguma atividade (além de seu curso)? Qual?

9) Se já cursou essa disciplina anteriormente, qual considera o principal motivo para a não aprovação?

10) Qual o seu curso e por que o escolheu?

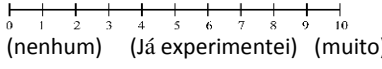
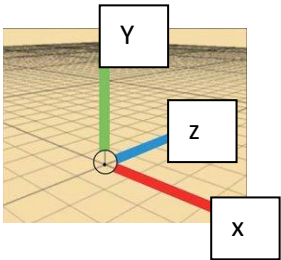
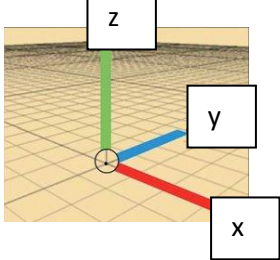
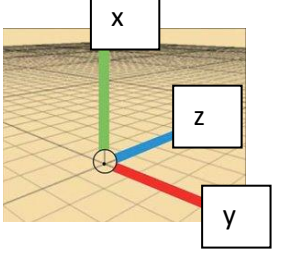
APENDICE B: Questionário sobre Ambiente 3D



Universidade Federal do Amazonas
Instituto de Ciência da Computação
Grupo de Sistemas Inteligentes



Questionário sobre Ambiente 3D

- i) Familiaridade com ambientes imersivos (por exemplo, jogos 3D) 
- 1) Realidade virtual é uma tecnologia de interação natural. Com esta definição o usuário pode entrar em ambiente em tempo real e pode aumentar a percepção de presença, ou seja, o usuário consegue criar uma realidade e o cérebro aciona a presença dos personagens ou objetos que o computador processa. ()V ()F
 - 2) Ambientes imersivos proporcionam uma imersão cada vez mais realística, em função da qualidade de seus elementos gráficos, capacidade de renderização e também, no caso dos games, por periféricos que permitem réplicas fidedignas dos movimentos do usuário no mundo virtual. ()V ()F
 - 3) O processo de ambientes virtuais consideram as seguintes características: imersão, navegação e iteração. ()V ()F
 - 4) Imersão para descrever a experiência única em que as pessoas estão completamente atraídas pela atividade e nela envolvidos, ou seja, os usuários têm a impressão de estar realmente interagindo com aquela realidade artificial. ()V ()F
 - 5) Programas utilizados para criações de cenários e objetos 3D: Sketup, Blender, 3Ds Max. ()V ()F
 - 6) Mundo Virtual é um ambiente simulado através de recursos computacionais destinado a ser habitado e a permitir a interação dos seus usuários via avatares. ()V ()F
 - 7) Avatar é a imagem que você escolhe - e que considera adequada - para a representação da sua identidade perante os outros. ()V ()F
 - 8) A melhor forma de compreendermos mundos virtuais é através de um conjunto mínimo de características. Dentre elas podemos citar: Representação física, Interatividade e Persistência. ()V ()F
 - 9) Persistência é o estado de seus objetos se preservam independente da presença do usuário. ()V ()F
 - 10) No caso da "2D" temos noção de tamanho (Largura x Profundidade) e no 3D, além de tamanho, temos a Altura. ()V ()F
 - 11) Qual figura esta correta:
 - a) 
 - b) 
 - c) 
 - 12) OpenSimulator, frequentemente chamado de OpenSim, é um jogo open source para hospedagem de mundos virtuais similar ao Second Life. ()V ()F
 - 13) Viewer são servidores de hospedagens para mundos virtuais. ()V ()F

APENDICE C: Questionário sobre a utilização do Ambiente 3D



Poder Executivo
Ministério da Educação
Universidade Federal do Amazonas
Instituto de Computação – IComp



INTRODUÇÃO A COMPUTAÇÃO – Turma 1 – 2012/2

Nome: _____

Prezado estudante,

Este questionário integra a unidade “Projeto de Pesquisa Cooperativo” da disciplina e tem por objetivo registrar suas impressões sobre a utilização do ambiente 3D. Embora o preenchimento do questionário seja obrigatório, suas impressões NÃO afetarão de qualquer modo sua nota e são fundamentais para a melhoria do processo.

1. De modo geral, você considera que o uso do ambiente 3D facilita a resolução de problemas em programação?
 Sim Na maioria das vezes Algumas vezes Não
2. Você teria interesse em desenvolver projetos mais complexos no ambiente 3D?
 Sim Não
3. O uso da abordagem de Polya contribuiu na resolução de problemas?
 Sim Na maioria das vezes Algumas vezes Não
4. A transferência dos scripts em *Scratch* para o ambiente 3D foi:
 Muito fácil Fácil OK Difícil Muito difícil
5. A experiência dos projetos no ambiente 3D, comparada com sua impressão no *Scratch*, ficou:
 Pior Igual Melhor
6. Ao longo de toda a unidade, no uso do ambiente 3D, cite:
 (a) Principais aspectos **positivos**

 (b) Principais aspectos **negativos**

7. A navegação no *Imprudence* para a consequente definição de comandos, foi:
 Muito fácil Fácil OK Difícil Muito difícil
8. A personalização do Avatar, foi:
 Muito fácil Fácil OK Difícil Muito difícil
9. A dificuldade em registrar as atividades no seu grupo foi:
 Baixa Média Alta
10. A interação no seu grupo foi:
 Baixa Média Alta
11. De modo geral, você acha que para a disciplina, o uso do 3D:
 Prejudica muito Prejudica Não influi Contribui Contribui Bastante

SUGESTÕES (use o verso da folha):