

Iron Ears: Educational Game for Teaching and Learning of Data Structure

Alexandre da Rosa
Setor de Educação Profissional e
Tecnológica
Universidade Federal do Paraná
Curitiba, Brasil
alexandrerosa@ufpr.br

Romualdo Viana Filho
Setor de Educação Profissional e
Tecnológica
Universidade Federal do Paraná
Curitiba, Brasil
romualdo.viana@ufpr.br

Gabriel Vaz Igarashi
Setor de Educação Profissional e
Tecnológica
Universidade Federal do Paraná
Curitiba, Brasil
gabriel.igarashi@ufpr.br

Vinicius Struginski Pereira
Setor de Educação Profissional e
Tecnológica
Universidade Federal do Paraná
Curitiba, Brasil
struginskipereira@ufpr.br

Rafael Hitoshi de Oliveira
Setor de Educação Profissional e
Tecnológica
Universidade Federal do Paraná
Curitiba, Brasil
rafaelhitoshi@ufpr.br

Andreia de Jesus
Setor de Educação Profissional e
Tecnológica
Universidade Federal do Paraná
Curitiba, Brasil
andreia.jesus@ufpr.br

Abstract— Alongside the technological evolution, education also evolves and adapts, thus appearing the educational software. Between the educational software are the educational games, which distinguish of common games as well on the goals, that would be the promotion of teaching using the ludic and generate entertainment, as the way that are produced. Considering the beneficial effects that educational games owns, it was decided to use it to help the teaching of data structure. Therefore, the proposed game is called “Iron Ears: Data Structure” and is aimed to Analisis and Development Systems students and similar, approaching the content of linked lists, queues, stacks and trees.

Keywords—higher education, data structure, educational game, serious game

I. INTRODUÇÃO

O ato de educar se adapta e se transforma naturalmente ao estado atual da sociedade. Hoje, as tecnologias digitais conquistaram o dia a dia das pessoas de diversas formas, trazendo novas possibilidades de interações e, como resultado, adentrou o mundo da educação, viabilizando que o processo de ensino-aprendizagem se beneficie desta. Um exemplo é o uso de softwares educacionais, cujo o intuito é o de auxiliar no processo educacional.

Softwares educacionais podem ser classificados de diferentes formas e uma delas é o jogo. Os jogos educacionais são ferramentas muito eficientes quando se trata de ensinar, pois enquanto o conhecimento é transmitido, eles divertem, motivam e desafiam, aumentando a capacidade de absorção do conteúdo e o desenvolvimento de capacidades intelectuais do jogador/aluno ([1]). Notando os benefícios advindos desta categoria de software, pensou-se como eles poderiam ajudar no ensino-aprendizagem de Estruturas de Dados, que é um tema obrigatório para qualquer estudante da área da Computação e da Tecnologia da Informação e que, por vezes, acaba sendo um dos mais complexos dessas áreas. Então, para fundamentar a importância de contribuir para a melhora do processo de ensino-aprendizagem na disciplina de Estrutura de Dados, foram coletadas informações referentes ao rendimento de duzentos (200) estudantes do curso superior de Tecnologia em Análise e Desenvolvimento de Sistemas (TADS) da Universidade Federal do Paraná dos anos de 2016 (1o. e 2o. semestre) e 2017 (1o. Semestre) (Fig. 1).

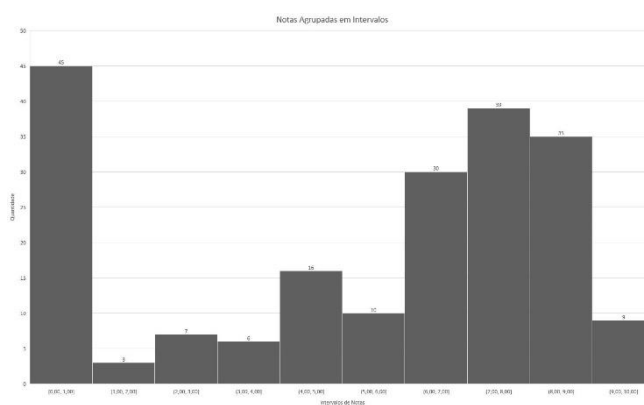


Figura 1. Quantidade de notas por intervalo

Na figura 1 é possível ver as notas coletadas, agrupadas em intervalos de 1 em 1, iniciando com a primeira coluna, da esquerda para a direita, representado o intervalo entre 0 e 1 e finalizando com a última coluna representado o intervalo entre 9 e 10. De acordo com os dados coletados, analisou-se que: apenas 54% dos estudantes foram aprovados sem necessitar de exame final; 12% foram aprovados com exame; 6% foram reprovados em exame; 20,5% foram reprovados por falta; 7,5% foram reprovados por nota. Mesmo com mais da metade dos estudantes aprovados, a média das notas fica em 7,97 e apenas 9 de 200 alunos conseguiram uma nota superior a 9. Então, tendo como base a análise desse conjunto de dados, nasce a ideia de um jogo educacional voltado para a disciplina de Estrutura de Dados. O público-alvo são os estudantes do TADS e quaisquer outros indivíduos, que terão a oportunidade de complementar o seu processo de aprendizagem nessa disciplina com uma ferramenta lúdica, a qual contribuirá para uma compreensão mais fácil dos conceitos abstratos referentes a Estrutura de Dados. Logo, o objetivo deste artigo é apresentar a fundamentação e o processo de desenvolvimento desse jogo para desktop.

II. METODOLOGIA APLICADA NO DESENVOLVIMENTO DO JOGO

O modelo que foi escolhido para a execução da metodologia de desenvolvimento do jogo tem base no Processo Unificado (UP). Este modelo é intitulado de Modelo de Processo para o Desenvolvimento de Serious Games, sendo

este descrito por [2] como "uma terminologia que foi estabelecida na década de 1970 e caracteriza jogos como dispositivos educacionais para quaisquer faixas etárias e situações diversificadas". Assim sendo, este modelo se enquadra melhor neste projeto, pois o objetivo aqui é a concepção de um jogo educacional. Para a execução do Processo Unificado, o qual foi adaptado, aplicou-se o modelo proposto por [3], que é constituído de um conjunto de disciplinas que proporcionam diretrizes para definir as tarefas e atribuir responsabilidades em um projeto. Neste são definidas quatro fases, sendo elas, concepção, pré-produção, produção e pós-produção. Um projeto que aplica estas fases terá no final elaborado uma geração do jogo. Esta passagem é conhecida como ciclo de desenvolvimento ou marco. Além das fases de desenvolvimento também existem sete processos de criação e três de apoio, que são tratados cada um como uma disciplina. Sendo que a ênfase em cada uma destas disciplinas varia conforme o andamento do projeto (Fig. 2).

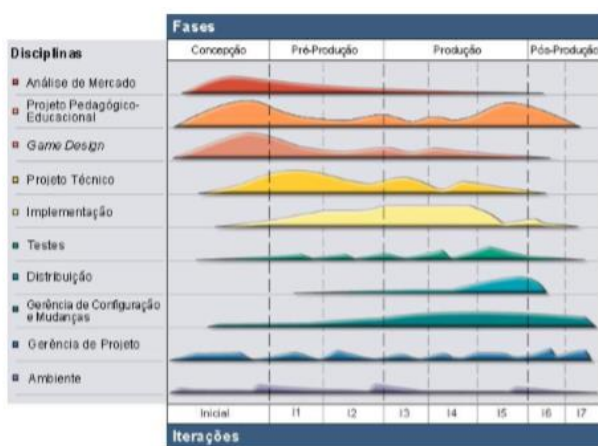


Figura 2. Estrutura do Modelo de Processo para o Desenvolvimento de Serious Games [2].

Os documentos modelados durante este processo foram divididos em Análise Essencial e Modelagem de Jogos. A Análise Essencial foi composta pelas seguintes ferramentas: Diagrama de Contexto, Lista de Eventos, Diagrama de Fluxo de Dados, Dicionário de Dados, Diagrama de Fluxo de Telas e Diagramas de Transição de Estados. Já a Modelagem de Jogos foi composta por: Página-única, Dez-páginas, Gráfico de Ritmo, Game Design Document (GDD).

A tela de introdução é a primeira tela do sistema, onde nela é mostrado um vídeo de introdução sobre a história que permeia a narrativa do jogo. A tela de menu é a principal tela do sistema em relação à navegabilidade, quase todas as telas são ligadas a ela e é através dela que o jogador poderá ir para outros espaços do jogo.

A tela de extras concentra os caminhos para se chegar a tela de créditos, cutscenes e arte conceitual. A tela de créditos é responsável por exibir as informações referentes a todos que contribuíram com o desenvolvimento do projeto. A tela de cutscenes permite acessar todas as cutscenes presentes no jogo e que aparecem quando há o desbloqueio de um novo mundo no mesmo. Por último, a tela de arte conceitual apresenta toda a arte criada para se chegar ao artefato final deste projeto.

Em relação à jogabilidade, a tela de gameplay é a tela onde o jogo de fato irá acontecer, no entanto para se chegar a esta tela deve-se passar pelas telas de save, de escolha de mundos e de fase. A tela de save é responsável por escolher e registrar o progresso do jogador. A tela de escolhas de mundo permite que o jogador escolha o mundo em que ele irá jogar, sendo que cada mundo representa um tipo de estrutura de dados, sendo um para listas em geral, um para pilhas e filas e outro para árvores. Por fim, tem-se a tela de escolha de fases, que conterá as fases referentes ao mundo escolhido, sendo 10 fases para cada mundo. Agora com relação a modelagem de jogos, devido ao alto grau de especificidade, é importante compreender o mínimo das funções de cada documento durante o desenvolvimento. Portanto, uma breve descrição destes documentos é apresentada a seguir.

O Página-única [4] é o primeiro documento que ajuda a compor o GDD. O objetivo dele é dar uma visão geral do jogo e tem o intuito de ser lido por todas as partes interessadas ao jogo, não só a equipe de desenvolvimento, por isso deve ser escrito de maneira que possa ser interessante lê-lo por todos. Enquanto o Página-única funciona como uma espécie de rascunho do jogo, o Dez-páginas [4] é um documento mais amplo que define a espinha dorsal do jogo. Seu objetivo é fazer com que os leitores possam compreender o básico do produto, sem entrar em muitos detalhes.

O Gráfico de Ritmo é responsável por descrever os elementos das fases do jogo e leva este nome por permitir que os designers de jogos consigam perceber e modelar um senso de progressão no jogo. Segundo [4, pp. 100-102], "você pode, então, identificar deficiências no design e começar a mover os elementos. Preenchendo buracos aqui, limpando lugares inchados ali." (p. 100, 102).

Já um GDD contém tudo que estará presente no jogo. É nele que toda a equipe irá se referenciar durante o desenvolvimento do jogo. Normalmente GDDs giram em torno de trezentas páginas, mas esse valor pode variar bastante a depender do jogo. Devido a inexistência de um formato oficial para este documento, a equipe deste projeto se baseou no formato apresentado por Scott Rogers em seu livro "Level Up: um guia para o design de grandes jogos".

Para o desenvolvimento do jogo, com esta metodologia, os integrantes dividiram esforços para se concentrar em três subequipes diferentes, arte, design e desenvolvimento, sendo que a liberação dos recursos de uma subequipe acarretaria na alocação em outra subequipe.

Por fim, o jogo foi implementado na plataforma Unity, utilizando C# como linguagem de desenvolvimento e aplicando o paradigma orientado à eventos, o que justifica o uso da Análise Essencial como parte da documentação do jogo.

III. APRESENTAÇÃO DO IRON EARS

O jogo em si possui uma história, contada através de cenas de corte que simulam uma história em quadrinhos (Figura 3), de forma a intercalar com os momentos de gameplay, para que haja um maior entretenimento e engajamento do jogador. O trecho de texto a seguir é referente a contextualização da história do jogo: "Em Todanim, diversas espécies, antes consideradas sem inteligência, dominam. Tendo os golfinhos

como governantes máximos dos reinos aquáticos, bem como uma coalizão para reger os povos voadores, além de uma ordem de sábios guardiões, formados por corujas, que impulsionaram o desenvolvimento intelectual dos outros povos, inclusive lobos ferozes que constituem a Grande Alcateia Lupus, um dos povos com maior extensão territorial. Porém, o povo com pleno desenvolvimento tecnológico é sem dúvida a República Parlamentarista de Terralepo, formada por uma nação de coelhos humanoides, que com seus Mechas formam o exército dos Orelhas de Ferro, que enfrentam a impetuosa alcateia em sua busca por dominância mundial. Sendo a árdua tarefa de gerir a produção dos Mechas atribuída ao recém-chegado Albus, que deverá encontrar maneiras inovadoras, baseando-se em conhecimentos criados por humanos a centenas de anos, para conseguir manter uma linha de produção de robôs em alto nível. Faça seus neurônios se contorcerem para ajudar na gestão do exército destes coelhos impetuosos e impeça o domínio dos lobos raivosos. ”

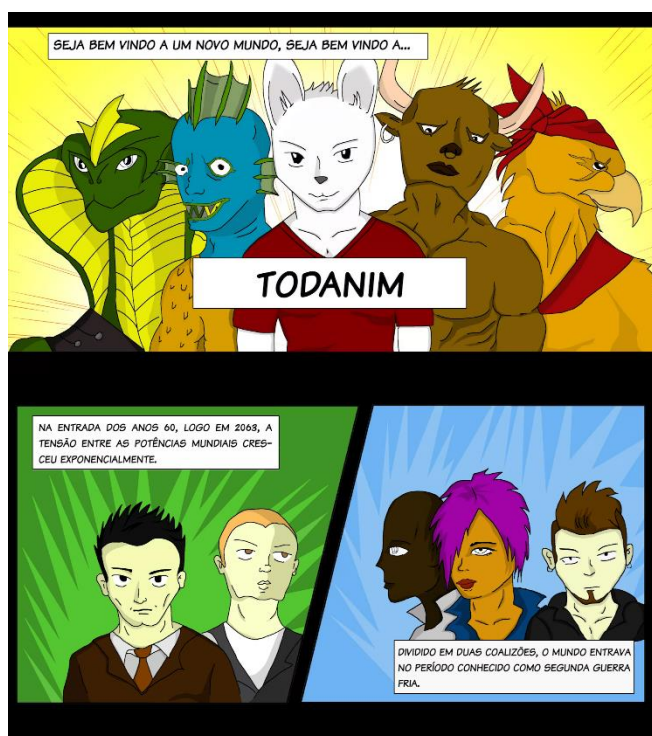


Figura 3. Cenas de Corte – Introdução.

Em Iron Ears: Data Structure, que é um jogo de quebra-cabeça e simulação em 2D, o jogador será desafiado a ajudar na gestão de linhas de produção de mechas dos queridos leporídeos de Terralepo, para que estes consigam se manter firmes contra os constantes ataques provenientes da Alcateia. O jogador deverá utilizar de seus conhecimentos em Estruturas de Dados para criar diversos Tipos de Dados Abstratos (TDA), representados por estações de trabalho, ao mover comandos referentes às operações destes TDAs para que uma linha de produção seja criada. Em cada mundo será usado Estruturas de Dados diferentes para que o jogador alcance os objetivos da fase e possa ajudar Albus e os Orelhas de Ferro.

Quanto a jogabilidade, o controle do jogador consiste em movimentos de arrastar e soltar ‘comandos’, utilizando o mouse, para a planilha de execução. Além de preenchimentos

de campos específicos a partir do teclado e seleção de opções a partir de escolhas drop-down (Figura 4). O jogo é dividido em 3 ‘mundos’, contendo cada um, 10 fases, nas quais o jogador deverá cumprir com um objetivo em específico, informado no início da mesma. Tendo que utilizar conhecimentos em Estruturas de Dados para realizar as tarefas.

O primeiro ‘mundo’ irá necessitar de conhecimentos em listas lineares das diferentes formas (dinâmica, estática, encadeada), já o segundo ‘mundo’ irá requisitar conhecimento em filas e pilhas, e finalmente o terceiro ‘mundo’ demandará compreensão de árvores.

Este ritmo completo de jogabilidade pode ser sequenciado em quatro passos. Iniciando-se pelo passo 1, quando a fase é aberta e seu objetivo é apresentado ao jogador. Na sequência, a tela com os comandos e a opção de executar sua solução é liberada ao jogador para que ele possa alcançar o objetivo proposto. Assim que se sinta seguro, o mesmo pressiona o botão de Play que irá movimentar a linha de produção de acordo com os comandos inseridos. Por fim, com a solução correta, o jogo exibe a tela de sucesso com a pontuação do jogador de uma a três cenouras, sendo esta baseada no número de comandos utilizados na solução.

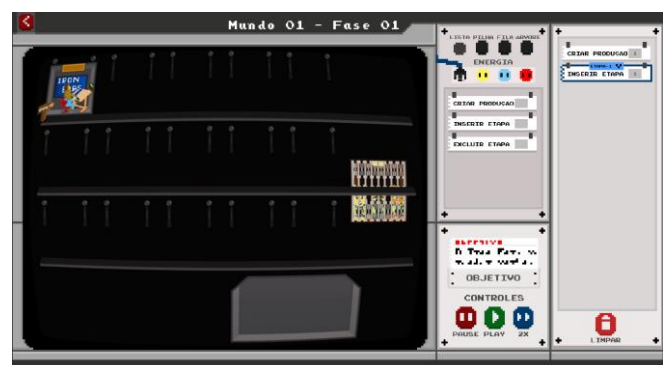


Figura 4. Tela do jogo

IV. POTENCIAL DE APLICAÇÃO

O principal objetivo do Iron Ears é a de possibilitar que o ensino-aprendizagem de Estruturas de Dados possa ser descomplicado e, mais que isso, servir como um incentivo para o seu aprendizado fazendo uso do lúdico, como já mencionado.

A ideia aqui não é a de remover a necessidade de se ter um responsável por ensinar ou instruir sobre os princípios básicos do uso de tipos de dados abstratos, mas pelo contrário, o Iron Ears surge como uma ferramenta a ser utilizada em conjunto com outras formas de ensino de Estrutura de Dados, principalmente nos cursos de desenvolvimento, como TADS e outros.

Como parte do que propõe a utilização do Iron Ears, uma versão prévia do jogo será utilizada nas aulas de Estruturas de Dados do curso de TADS da UFPR, sendo possível assim, validar seu uso para que o seu desenvolvimento possa ser ajustado e melhorado, além de poder assim analisar de

maneira mais concreta os benefícios de utilização da ferramenta nesse processo de ensino-aprendizagem.

V. CONSIDERAÇÕES FINAIS

Muito ainda se espera alcançar com o desenvolvimento do Iron Ears, que no escopo atual, está longe de atingir todo o potencial pensado pelos autores deste projeto. Ainda assim, busca-se, através da versão final desta etapa do projeto, possibilitar aos educadores e estudantes de programação, uma ferramenta funcional e realmente útil na área de programação.

Pensa-se, ainda, que futuramente, o jogo possa também tratar de grafos entre as estruturas de dados que ele aborda. Além disso, permitir também, através de uma plataforma externa, um local para que, por exemplo, um professor possa ver estatísticas de seus alunos envolvendo as soluções dos problemas apresentados pelo jogo.

REFERÊNCIAS

- [1] Freire, P. Educação e mudança. São Paulo: Paz e Terra, 2003 apud Schiffer, 2008. Paginação irregular.
- [2] Lemes, D.O. Serious Game: jogos e educação. Disponível em <http://www.abrelivros.org.br/home/index.php/bienal-2014/resumos-efotos/5647-1-primeiro-resumo>. Acesso em 01 set. 2017.
- [3] Rodrigues, H. F.; Machado, L. S.; Valença, A. M. G. Definição e Aplicação de um Modelo de Processo para o Desenvolvimento de Serious Games na Área de Saúde. Disponível em http://www.de.ufpb.br/~labteve/publi/2010_wim2.pdf. Acesso em: 31 ago. 2017.
- [4] Rogers, S. Level Up: um guia para o design de grandes jogos. São Paulo: Blucher, 2014