



VICTOR LELLIS COELHO

ATLAS DIGITAL DE ANATOMIA HUMANA 3D - ADAH3D

**LAVRAS-MG
2019**

VICTOR LELLIS COELHO

ATLAS DIGITAL DE ANATOMIA HUMANA - ADAH3D

Relatório Técnico apresentado à
Universidade Federal de Lavras, como
parte das exigências do Curso de
Graduação, para a obtenção do título de
Bacharel em Sistemas de Informação.

Prof. Dra. Ana Paula Piovesan Melchiori
Orientadora

LAVRAS – MG
2019

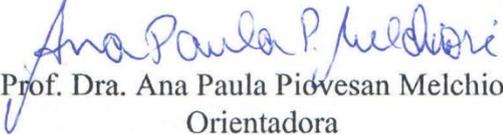
VICTOR LELLIS COELHO

ATLAS DIGITAL DE ANATOMIA HUMANA - ADAH3D

Relatório Técnico apresentado à
Universidade Federal de Lavras, como parte
das exigências do Curso de Graduação, para
a obtenção do título de Bacharel em Sistemas
de Informação.

APROVADA em 22 de Novembro de 2019

Dra. Ana Paula Piovesan Melchiori	UFLA
Dr. Joaquim Quinteiro Uchoa	UFLA
Dra. Giancarla Aparecida Botelho Santos	UFLA


Prof. Dra. Ana Paula Piovesan Melchiori
Orientadora

**LAVRAS – MG
2019**

RESUMO

Através do uso da realidade virtual na educação, pode-se auxiliar no processo de ensino e aprendizagem do aluno, estimulando o interesse em buscar mais conhecimento na área da saúde e também auxiliar os professores na ministração das aulas práticas e teóricas da Universidade Federal de Lavras. Com essa motivação, o projeto consiste no desenvolvimento de um Atlas Digital de Anatomia Humana 3D (ADAH3D), que possibilita a visualização do sistema ósseo humano em imagens tridimensionais. Para modelagem 3D deste sistema foi realizado uma digitalização de modelos anatômicos em resina, utilizando equipamentos e *softwares* de realidade virtual, como Blender 3D e Unity 3D. A partir dos modelos 3D finalizados foi possível identificar os acidentes ósseos específicos de cada um com a assistência de um monitor da disciplina de anatomia humana, para que estes pudessem ser identificados na aplicação de interação com o usuário. Esta que foi desenvolvida utilizando o *software* Unity 3D, possibilitando criar uma interface intuitiva e de fácil usabilidade, na qual o usuário poderá realizar estudos específicos de cada osso do corpo humano com seus respectivos conteúdos. Para que O em que o administrador do sistema poderá realizar um gerenciamento deste conteúdo através de um módulo web de gerenciamento de todos os ossos, conteúdos e controle de acesso dos usuários. Como resultado do trabalho pretende-se ofertar mais uma alternativa para estudantes da disciplina de anatomia humana.

Palavras-chave: Realidade Virtual. Anatomia Humana. Atlas. 3D. Ensino e Aprendizagem. Computação Gráfica.

LISTA DE FIGURAS

Figura 1 - Scanner Artec 3D 7	11
Figura 2 - Tela inicial do Blender 3D.....	12
Figura 3 - Diagrama de Caso de Uso da cena inicial do ADAH 3D	15
Figura 4 - Diagrama de Caso de Uso- Visualização específica de um modelo ósseo 3D	16
Figura 5 - Modelo do crânio posicionado para ser digitalizado.....	17
Figura 6 - Régua do medidor de distância do Artec Studio 11	18
Figura 7 - Página do 3D View do Artec Studio 11	19
Figura 8 - Imagem com o rastreamento perdido.....	19
Figura 9 - Modelo de uma vértebra com marcadores para ser digitalizada	21
Figura 10 - Processo de alinhamento manual	21
Figura 11 - Modelo anatômico em resina do crânio	23
Figura 12 - Modelo virtual 3D	23
Figura 13 - Modelo 3D do crânio exportado do Artec Studio e importado no MeshLab	24
Figura 14 - Área do acidente ósseo Face Patelar do Fêmur selecionado.....	25
Figura 15 - Acidente ósseo Face Patelar do Fêmur identificado e nomeado.....	25
Figura 16 - Espelhamento do Fêmur realizado.....	26
Figura 17 - Modelo do Esqueleto humano completo estruturado no Blender 3D	27
Figura 18 - Tela inicial do Unity 3D	28
Figura 19 - Gerenciamento de cenas no Unity3D.....	29
Figura 20 - Painel exibindo a identificação com o mouse do fêmur	31
Figura 21 - Lâmpadas posicionadas na cena para visualização dos modelos 3D.....	31
Figura 22 - Objeto vazio para rotação da câmera e suas configurações	32
Figura 23 - Câmera posicionada para visualização da cena com o objeto.	33
Figura 24 - Botões para seleção de vistas dos modelos	36
Figura 25 - Cena com os botões para seleção da divisão do esqueleto	36
Figura 26 - Cena de Ajuda do ADAH3D	37
Figura 27 - Tabelas do Banco de Dados Relacional do ADAH 3D	38
Figura 28 - Tela de Login do ADAH 3D	39
Figura 29 - Página de cadastro de Osso do módulo de gerenciamento	40
Figura 30 - Página de cadastro e gerenciamento de conteúdo dos modelos.....	40
Figura 31 - Página com o modal para cadastro de conteúdo.....	41
Figura 32 - Cena inicial do ADAH 3D	42
Figura 33 - Cena para visualização individual do modelo 3D do Fêmur na vista Ventral.....	43
Figura 34 - Cena do esqueleto Axial com o mouse sobre o crânio	43
Figura 35 - Cena do esqueleto Apendicular com o mouse sobre a Escápula Esquerda.....	44
Figura 36 - Página Inicial do módulo de gerenciamento do administrador	44

SUMÁRIO

RESUMO	3
1. INTRODUÇÃO	7
2. CONCEITOS E FERRAMENTAS UTILIZADAS	9
2.1. Conceitos e tecnologias de Computação Gráfica e Realidade Virtual	9
2.1.1. Conceitos e ferramentas para modelagem tridimensional	10
2.2. Conceitos, ferramentas e softwares utilizados para o desenvolvimento do módulo de gerenciamento de conteúdo	13
2.2.1. Arquitetura do módulo de gerenciamento	14
3. METODOLOGIA DE DESENVOLVIMENTO	15
3.1. Levantamento de requisitos	15
3.2. Processo de digitalização e modelagem 3D	16
3.3. Retopologia dos modelos	23
3.4. Identificação dos acidentes ósseos	24
3.4.1. Técnica de Espelhamento	26
3.4.2. Estrutura do esqueleto completo no Blender 3D	26
3.5. Desenvolvimento da aplicação no Unity 3D	27
3.5.1. UI do ADAH3D	29
3.5.2. Processo de Rotação, Translação e <i>zoom</i> da câmera	32
3.5.3. Cena para visualização de modelos e conteúdos específicos	35
3.5.4. Esqueleto Axial e Apendicular	36
3.5.5. Cena de Ajuda	37
3.5.6. Execução e Disponibilização do ADAH3D	37
3.6. Desenvolvimento do módulo de gerenciamento de conteúdo	37
3.6.1. Banco de Dados	38
3.7. Acesso e controle de permissões do ADAH 3D	38
3.8. Cadastros de Ossos, Conteúdo e Usuários	39
4. RESULTADOS	42
5. CONCLUSÕES E TRABALHOS FUTUROS	45

1. INTRODUÇÃO

Para obter um bom resultado no processo de ensino-aprendizagem na disciplina de anatomia humana, além das metodologias tradicionais, pode-se utilizar outras tecnologias disponíveis que agregam na aquisição de conhecimento. Isso se torna possível pelo avanço tecnológico e o uso da informática na educação, auxiliando tanto o aluno, quanto o professor neste processo de ensino-aprendizagem. Ainda nesse contexto, a utilização de realidade virtual aplicada em aula, possibilita ao docente estimular o aluno progressivamente a buscar o que precisa para ampliar o conhecimento na área e não se limitar em suas pesquisas.

Pelo fato do conteúdo desta disciplina ser tão complexo e com várias particularidades essenciais que o aluno deve compreender, pode haver um desestímulo para buscar o conhecimento, passando a ser uma difícil tarefa para o docente ao utilizar os métodos tradicionais de ensino utilizando materiais como livros, atlas impressos e 2D. E ainda alguns destes materiais, podem ter o acesso limitado pela biblioteca da instituição, por ter um número reduzido de exemplares. Tem-se também o uso de materiais 3D impressos que são usados em aulas práticas, mas que não podem deixar o recinto para estudos fora da sala de aula, deixando o aluno com tempo limitado para os estudos, e estes modelos que também sofrem com o desgaste com o tempo, perdendo a qualidade com o uso excessivo.

O objetivo deste projeto consiste na criação de um Atlas Digital de Anatomia Humana 3D - ADAH 3D, que possibilita a visualização de simulações em 3D de modelos específicos de ossos do sistema esquelético humano. Dessa forma o usuário pode interagir com o ambiente virtual e obter acesso aos conteúdos e imagens tridimensionais de cada modelo individualmente, e poder identificar os acidentes ósseos de cada um.

Além do objetivo geral acima exposto, os objetivos específicos deste trabalhos são:

- desenvolvimento de uma plataforma pedagógica para ensino de anatomia humana;
- desenvolvimento de uma plataforma com fácil usabilidade;
- suprir a carência de materiais da instituição;
- evitar desgastes de modelos anatômicos 3D impressos em resina e de humanos.

O que nos motiva neste projeto é que iremos contribuir para o estudo da anatomia, tendo um potencial para suprir a carência de materiais didáticos em instituições, estimulando o usuário para participação ativa no software de realidade virtual, podendo atingir áreas nas quais os métodos tradicionais de ensino de Medicina estejam falhando. Contudo, já existem Atlas Digitais de Anatomia Humana disponíveis para acesso na internet ou instalados no localmente no computador, mas estes são pagos e muitos alunos não possuem condições para aquisição deste tipo de *software* para estudos, e não trazem as identificações específicas dos acidentes ósseos com seus respectivos conteúdos. Com isso, a criação do Atlas irá possibilitar trazer esses detalhes reais para o mundo virtual, no qual o aluno poderá acessar de qualquer lugar através de um computador com acesso à internet para realizar seus estudos, sem precisar estar presente no laboratório da instituição.

2. CONCEITOS E FERRAMENTAS UTILIZADAS

Antes de iniciar o desenvolvimento do projeto, foram realizadas pesquisas e estudos para buscar novas técnicas e tecnologias para melhor simulação virtual do sistema esquelético, estudando os conceitos, *softwares*, equipamentos e o processo de desenvolvimento a ser utilizado.

Existem diversos tipos de processos de desenvolvimento de *software*, mas segundo Sommerville (2011), todos estes processos devem ter quatro atividades fundamentais, na qual foram adotadas para o desenvolvimento do ADAH3D. A Especificação do *Software*, na qual serão definidas as funcionalidades da aplicação com levantamento de requisitos. A fase de Implementação, na qual é realizado o desenvolvimento da aplicação. A fase de Testes, para melhoria de usabilidade e correções no sistema. E a Evolução do *Software*, para incremento de novos módulos.

2.1. Conceitos e tecnologias de Computação Gráfica e Realidade Virtual

Realidade Virtual (RV) é uma experiência imersiva e/ou interativa baseadas em imagens 3D, ou seja uma simulação do mundo real (Rodrigues e Porto, 2013). Aplicações de realidade virtual, permite que o usuário navegue e interaja com o ambiente virtual com imagens tridimensionais que são originadas pelo computador, estas que são pontos que renderizam a imagem no monitor.

O ADAH3D é uma aplicação não imersiva e interativa. A Interação que está integrada à competência do computador de detectar as entradas do usuário e alterar-se em tempo real o mundo virtual e as ações sobre esse ambiente, levando em consideração o nível de envolvimento do usuário com o ambiente virtual, podendo ser relacionado como participante ativo e passivo.

Para que isso aconteça com maior qualidade de desempenho, é imprescindível a uso de determinados dispositivos como mouses, teclados, *joysticks* e outros. A partir destes é possível transformar a aplicação em um ambiente interativo, acompanhando as ações e movimentos realizados pelo usuário.

Para Manssour e Cohen (2006) a capacidade de manusear e modificar interativamente as propriedades dos modelos tridimensionais que integra o ambiente virtual, é uma das aplicabilidades mais importantes de computação gráfica. Estas que constitui-se em funções matemáticas aplicadas sobre os pontos do plano, transformando o aspecto do modelo tridimensional.

Os principais tipos de transformações dos modelos geométricos são: translação, rotação e escala. Este processamento que é representado por um conjunto de pontos (x, y, z).

A translação é utilizada para determinar a posição do modelo geométrico na cena, esta transformação resume-se em somar constantes e deslocar todos os pontos do plano, isto é, mudar o modelo tridimensional de posição.

O processo de rotação no desenvolvimento do atlas, que consiste em rotacionar um modelo 3D em torno de um eixo. Conforme Manssour e Cohen (2006) a operação aplica um arranjo de cálculos usando o seno e cosseno do ângulo de rotação a todas as coordenadas dos pontos que compõem o modelo, onde deve-se definir em qual eixo será rotacionado.

A partir do modelo *Screen-based*, é possível aplicar o processo de rotação e translação centrado na câmera. Segundo Castier, Martha e Gattass (1994), este modelo age no posicionamento da imagem de um objeto na cena em associação aos eixos que têm direções paralelas à tela. Essas que caracterizam as orientações do sistema de coordenadas do olho, que é equiparável aos planos x e y do olho humano. Contudo, os sentidos de rotação e translação do *game object* no modelo *Screen-based* são: vertical, horizontal e axial da câmera.

E também a escala, que é utilizada para aumentar ou diminuir o tamanho do modelo para ser exibido na cena, modificando os pontos, na qual é multiplicado por um valor todas as coordenadas dos pontos, que poderá ser ampliado ou reduzido. Este processo pode ser utilizado para realizar o *zoom* do modelo, porém para o desenvolvimento do projeto foi utilizado outra técnica para que possa melhorar o processamento da aplicação, conforme explicado no item 3.5.

2.1.1. Conceitos e ferramentas para modelagem tridimensional

Modelagem 3D é um processo de criação de algum objeto do mundo real ou inanimado através de uma reprodução matemática, por meio de *softwares* específicos, obtendo um modelo tridimensional como objeto final. Este modelo é composto por vértices, arestas e faces, na qual é criada uma malha dando formato ao objeto. Para realização do processo de modelagem 3D existem diversos tipos de técnicas e softwares que ajudam da melhor forma para criação do modelo desejado.

Para a criação dos modelos dos ossos humanos 3D foi utilizado a técnica de digitalização através de um *scanner* 3D e também a técnica de modelagem por imagem de

referência para ossos de escala menor, utilizando fotos dos modelos de resina para apoio a criação da modelagem no software

Para o processo de digitalização dos modelos anatômicos em resina do Atlas foi utilizado o Scanner Artec 3D® 7 e do software Artec Studio 11, com a licença adquirida com a aquisição do *scanner*, ilustrado na Figura 1, para assim digitalizar todos os modelos do esqueleto axial e apendicular do lado direito.

Figura 1 - Scanner Artec 3D 7



O funcionamento do *scanner* decorre pela projeção de pontos de luz para o modelo a ser digitalizado, e este interpreta uma baixa porcentagem destes pontos outra vez através da óptica. Por meio dessa técnica, os *scanners* 3D reproduzem o modelo tridimensional, produzindo uma imagem com o conhecimento da distância de cada ponto em relação ao objeto projetado.

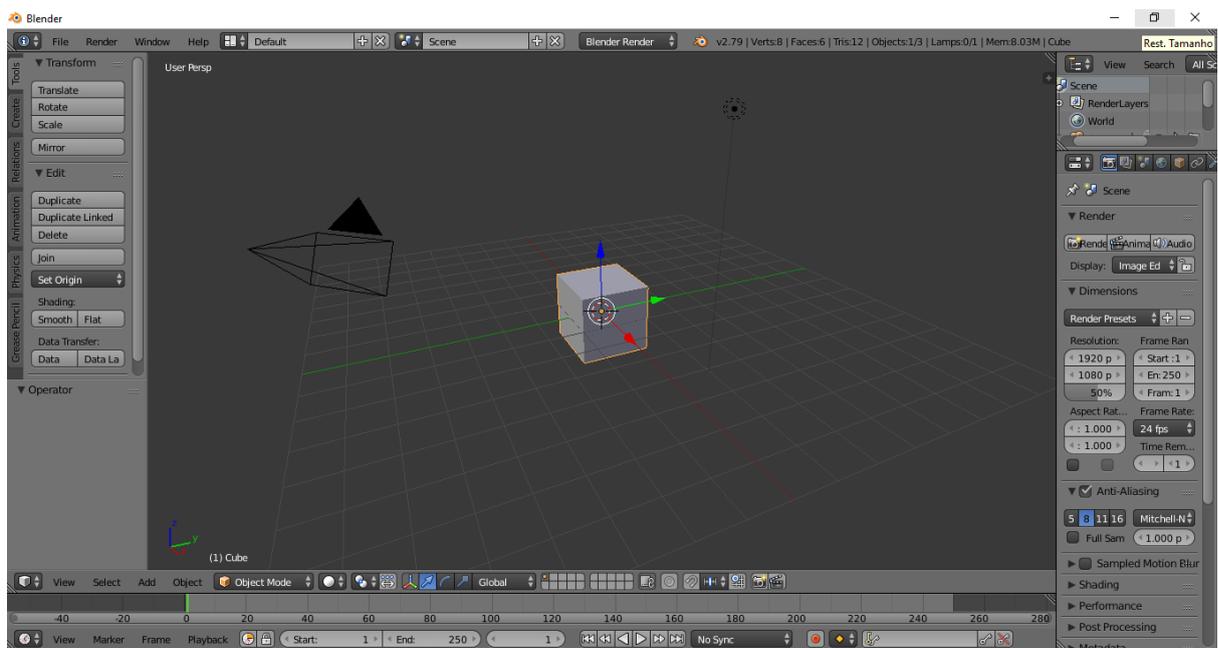
Com a digitalização dos modelos finalizada, ainda não é possível obter uma malha perfeita, é preciso passar por ajustes. Para isso foi utilizado o software MeshLab, de código aberto para realizar o processamento e tratamento de malhas triangulares em 3D. Através de algumas ferramentas disponibilizadas é possível editar, limpar, renderizar e realizar a retopologia do modelo, que é o principal foco, diminuindo o número de arestas e faces, e também o preenchimento de buracos na malha, reconstruindo a superfície.

Para a fase de identificação dos acidentes ósseos utilizou-se o software Blender 3D. Este que é um programa de código aberto, distribuído sob a licença GNU-GPL (*General Public Licence*). Para o desenvolvimento desse projeto foi utilizado a versão 2.57 e depois atualizada para a versão 2.79a. O programa possui ferramentas para proporcionar um processo de modelagem facilmente e também desenvolvimento de jogos, por possuir sua

própria *Game Engine*. Além de ser um programa gratuito de código aberto, é também de fácil usabilidade e uma vasta comunidade para obtenção de informações e conhecimento sobre a utilização do mesmo, por isso, o escolhemos para a fase de mapeamento dos acidentes ósseos e estruturação do esqueleto completo.

Na cena inicial do Blender 3D com a inicialização do programa é possível visualizar alguns objetos similares a outros *softwares* (Figura 2): o plano cartesiano com os três eixos (X, Y e Z) em plano quadriculado para melhor percepção do objeto 3D, uma câmera que pode ser movimentada com o clique do *mouse*, uma luz para iluminar a cena, e também um cubo que deve ser excluído ao iniciar um novo projeto.

Figura 2 - Tela inicial do Blender 3D



Para o desenvolvimento da aplicação de interação com usuário foi utilizado o *software* Unity 3D, versão 2018.3 *Personal*, na qual a licença é gratuita mas com limitações em comparação às versões pagas. Criado pela empresa Unity Technologies, é utilizado para desenvolvimento de jogos ou aplicações em 3D ou 2D, na qual possui sua própria *game engine*. Esta versão mesmo gratuita possui muitas ferramentas e API para o desenvolvimento do projeto, dando suporte para *WebGL* no Windows e Xbox 360, *OpenGL ES* no Android e iOS e *WebGL* na internet. No Unity também podem ser utilizadas as linguagens *C#* e *JavaScript*. Para a implementação dos *scripts* foi utilizada linguagem *C#*, sendo codificados com o *software* Visual Studio 2017, com licença incorporada do sistema operacional Windows da Microsoft.

Todo o projeto foi desenvolvido em um computador Dell XPS 8500 com processador

Intel® Core™ i7- 3770 (3.4GHz, 6MB Cache, 8 Threads) e 16 GB de RAM e sistema operacional Windows 10 64 bits.

2.2. Conceitos, ferramentas e softwares utilizados para o desenvolvimento do módulo de gerenciamento de conteúdo

O módulo de gerenciamento de conteúdo é um sistema web, onde é feito os cadastros de conteúdos que serão apresentados no Atlas e controle de acesso dos usuários. Este será gerenciado pela Prof.^a Dra. Giancarla Aparecida Botelho Santos e os monitores da disciplina de Anatomia Humana da Universidade Federal de Lavras.

Para o desenvolvimento deste módulo, foi necessário utilizar algumas tecnologias para o melhor funcionamento e arquitetura do projeto. Foram utilizadas as linguagens HTML para criação da estrutura das páginas *WEB*, CSS para definir o estilo das páginas do *frontend*, *JavaScript*, sendo utilizada para manipular o comportamento do HTML e do CSS na página. Para facilitar o desenvolvimento também foram utilizados *frameworks*, ajudando assim na reutilização de códigos e deixando-o mais limpo possível, sendo então utilizado Codeigniter 3.1.11 e Bootstrap na versão 3.7.

O *CodeIgniter* é um *framework* PHP aplicando o padrão de projeto M.V.C. para arquitetura dos projetos. Com uso deste, tem a possibilidade de facilitar o desenvolvimento de sistemas, possuindo uma vasta biblioteca para reutilização de código. Este *framework* dispõe de uma documentação bem estruturada e de fácil entendimento, também com uma grande comunidade para suporte à problemas encontrados.

Já o *Bootstrap* é um *framework* para o desenvolvimento do *front-end* do sistema, no qual é compatível com HTML5, CSS3 e também *JavaScript*, que ajuda na aplicação de alguns recursos, como a responsividade nos diversos tamanhos de tela. Também podem ser utilizados facilmente padrões de botões, barra de menus, tabelas e imagens. E também assim como o *CodeIgniter*, possui uma documentação completa e estruturada para fácil entendimento e uma grande comunidade de desenvolvedores.

Para o Sistema de Gerenciamento de Banco de Dado (SGBD) utilizado no desenvolvimento do ADAH 3D foi o MySQL e modelado utilizado o software Workbench 6.3, este que é de código aberto aplicado em grande parte das aplicações para gerenciar as bases de dados. Este SGBD usa a linguagem SQL (*Structure Query Language*), na qual é uma linguagem comum para inserir, consultar, excluir e atualizar todo o conteúdo armazenado em um banco de dados.

2.2.1. Arquitetura do módulo de gerenciamento

O desenvolvimento do módulo de gerenciamento de conteúdo e de controle de usuários foi empregado o padrão de projeto Model-View-Controller (M.V.C.), sendo um dos mais utilizados atualmente para o desenvolvimento de sistemas. A utilização deste modelo se dá em três etapas, sendo elas:

- *Model* (Modelo): esse pacote é a ligação entre as camadas *View* e *Controller*, constitui-se nos dados da aplicação, regras de negócios, lógica e funções. Para que os dados possam ser consultados, gravados e, exibidos. É responsável por tudo que a aplicação vai fazer com os dados a partir dos comandos enviados pela camada de controle.
- *View* (Visão): esse pacote é a saída das exibição dos dados no nosso projeto em uma página do navegador web. Os dados que são solicitados do modelo são enviados para a *view* para serem exibidos. No pacote de Visão também é possível realizar interações com o usuário, no qual envia os comandos para o pacote de Controle.
- *Controller* (Controlador): O pacote Controle, faz acontecer intermediação da entrada, convertendo-a em comandos para o modelo ou visão. É onde são manipulados os dados que o usuário insere ou atualiza. O foco do Controle é a ação do usuário.

3. METODOLOGIA DE DESENVOLVIMENTO

Neste capítulo serão apresentados os métodos, práticas, e processos utilizados para o desenvolvimento do Atlas Digital de Anatomia Humana 3D - ADAH3D, detalhando todo o processo desde a modelagem dos ossos, identificação dos acidentes ósseos, retopologia da malha, desenvolvimento da aplicação e o módulo de gerenciamento de conteúdo.

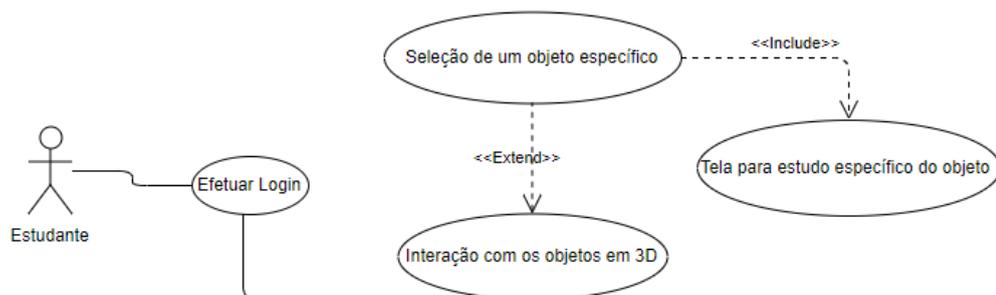
3.1. Levantamento de requisitos

Para a especificação do projeto de *software*, foi realizada uma análise da necessidade dos usuários finais através de entrevistas para fazer o levantamento de requisito com a Prof.^a Dra. Giancarla Aparecida Botelho Santos, do Departamento Ciências da Saúde (DCS) e com monitores da disciplina de Anatomia Humana da Universidade Federal de Lavras.

Para algumas funções principais especificadas foram elaborados casos de uso, mostrando as principais funcionalidades que serão executadas pelo usuário final, a fim de possibilitar a compreensão do comportamento do sistema, para que não gere dúvidas sobre suas necessidades, com isso podendo ser consultado durante todo o processo de desenvolvimento.

O diagrama de caso 1, conforme mostra a Figura 1, demonstra a funcionalidade principal do sistema. Após o *login* efetuado, o usuário será direcionado para a página inicial que terá a opção para acessar o ADAH 3D. Ao ser redirecionado para a cena principal, o usuário visualiza o esqueleto humano completo, podendo rotacioná-lo em 360° e obter os nomes de cada osso através da colisão do *mouse* com a malha do objeto, e com a opção de selecionar algum modelo para estudo individual.

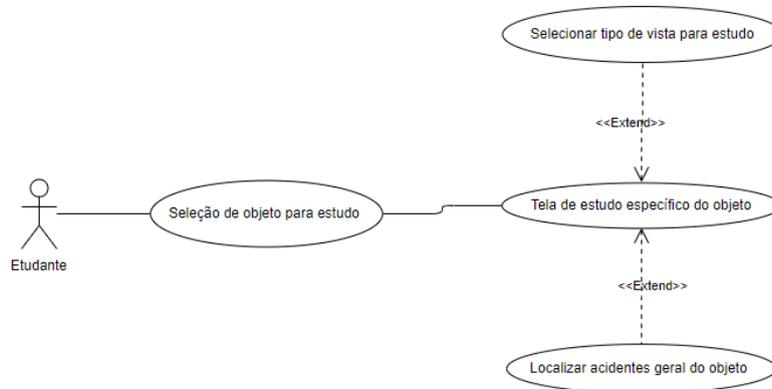
Figura 3 - Diagrama de Caso de Uso da cena inicial do ADAH 3D



O segundo diagrama de caso de uso, possui outra funcionalidade importante do sistema, conforme a Figura 2. A partir do modelo selecionado para estudo individual, o

usuário, além da possibilidade de rotacionar o modelo em 360° livremente, é possível também selecionar visões distintas do modelo já definidas e exibindo textos específicos para cada uma.

Figura 4 - Diagrama de Caso de Uso- Visualização específica de um modelo ósseo 3D



Outra necessidade é o controle de acesso ao ADAH3D, onde o usuário deverá realizar o cadastro antes de fazer o *login* caso ainda não tenha. Após o cadastro, o usuário poderá acessar o atlas com seu *e-mail* e senha, conseguindo então acessar a aplicação do atlas.

E para o gerenciamento de conteúdo foi levantado a necessidade de exibir um conteúdo por vista, sendo já definidas as vistas: frontal, dorsal, superior, inferior, lado esquerdo e lado direito. Para acesso a esse gerenciamento somente aos usuários que possuem permissão de administrador.

Também foram relacionados os requisitos não funcionais, destacando que o sistema deverá ser *web* disponível para computadores e dispositivos móveis, a partir do uso de um navegador, sendo responsivo em ambas as proporções de telas. O sistema deverá ter total disponibilidade e confiabilidade, para que os alunos e professores possam acessar a aplicação de qualquer lugar e a qualquer momento, a fim de propiciar mais tempo e comodidade para os estudos.

3.2. Processo de digitalização e modelagem 3D

A primeira parte do desenvolvimento do projeto consiste na modelagem da estrutura óssea do corpo humano em 3D através do Scanner Artec 3D® 7, conforme mencionado no item 2.1.1.

O processo de digitalização, foi realizado no Departamento de Ciência da

Computação no Laboratório de Educação Conectada. E para auxiliar neste processo foi necessário a utilização de um suporte, produzido no formato de uma caixa preta em MDF, dispondo de uma base giratória e uma haste fixa superior com ajustamento para estabilizar os modelos em diversas posições. Com os modelos dos ossos fixados e apoiados na base, é realizado uma rotação manual de 360°. Na qual o *scanner* em posição adequada e fixa faz a captura das imagens de acordo com o movimento da base giratória empurrando-a manualmente, girando o modelo em seu próprio eixo. Esta rotação deve ser precisa e devagar para que o objeto final esteja o mais fiel possível a forma original, por isso durante o processo manual de rotação não deve permitir que o modelo se mova fora do eixo.

Para que seja capturado de forma mais próxima possível do real é necessário realizar várias vezes o processo de digitalização conforme ilustrado na Figura 5, capturando as imagens de diversas posições, para que ao exportar o objeto 3D esteja mais fácil de manipulá-lo. Por isso, é necessário capturar no mínimo uma vez o modelo na posição vertical, e horizontal.

Figura 5 - Modelo do crânio posicionado para ser digitalizado



Com o *scanner* conectado ao computador, o *software* Artec Studio o identifica automaticamente. Este pode estar em três modos, sendo eles: o modo inativo (*Idle*), no qual indica que o *scanner* foi identificado, porém não está capturando ou exibindo imagens. O modo visualizar (*Preview*), na qual irá apenas exibir as imagens capturadas ao vivo, mas não está gravando na memória. Com este modo é possível posicionar o *scanner* e o modelo antes de iniciar a gravação, para que a captura seja a melhor possível. E por último o modo de

gravação (*Recording*), iniciando a gravação das imagens capturadas, podendo gravar até 15 frames por segundo e armazenando na memória RAM. Ao final de cada modelo capturado, deve-se realizar uma limpeza da memória, descarregando as verificações realizadas, e melhorando o processamento da máquina para próxima digitalização.

Antes do início da captura das imagens tridimensionais deve-se deixar configurado para o *scanner* utilizar o modo *geometric and texture*, capturando a geometria e também a textura do objeto, facilitando o processamento e edição da imagem no *software* Artec Studio 11. Porém, ao final do processo é exportado somente a forma geométrica, visto que a textura do modelo em resina também não é o real, sendo assim aplicou-se então uma cor neutra, melhorando também o processamento das imagens durante a execução da aplicação.

Para iniciar a captura, coloca-se o *scanner* em modo *Preview* para que possa ver a distância adequada do objeto. O Artec Studio possui um medidor de distância do lado esquerdo conforme exibido na Figura 6, demonstrando em tempo real a melhor distância. Para uma maior qualidade da captura, a janela *3D View*, conforme é mostrado na Figura 7, mostra as superfícies que se enquadram na faixa ideal para o *scanner*, deixando a distância pelo menos entre 520 a 880 do modelo conforme ilustrado na Figura 6, após posicionado da melhor forma clicar *Recording* para começar a gravação.

Figura 6 - Régua do medidor de distância do Artec Studio 11

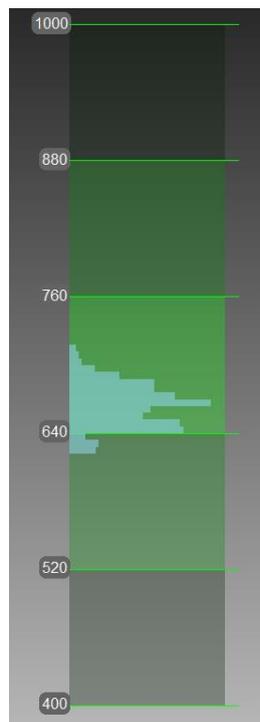
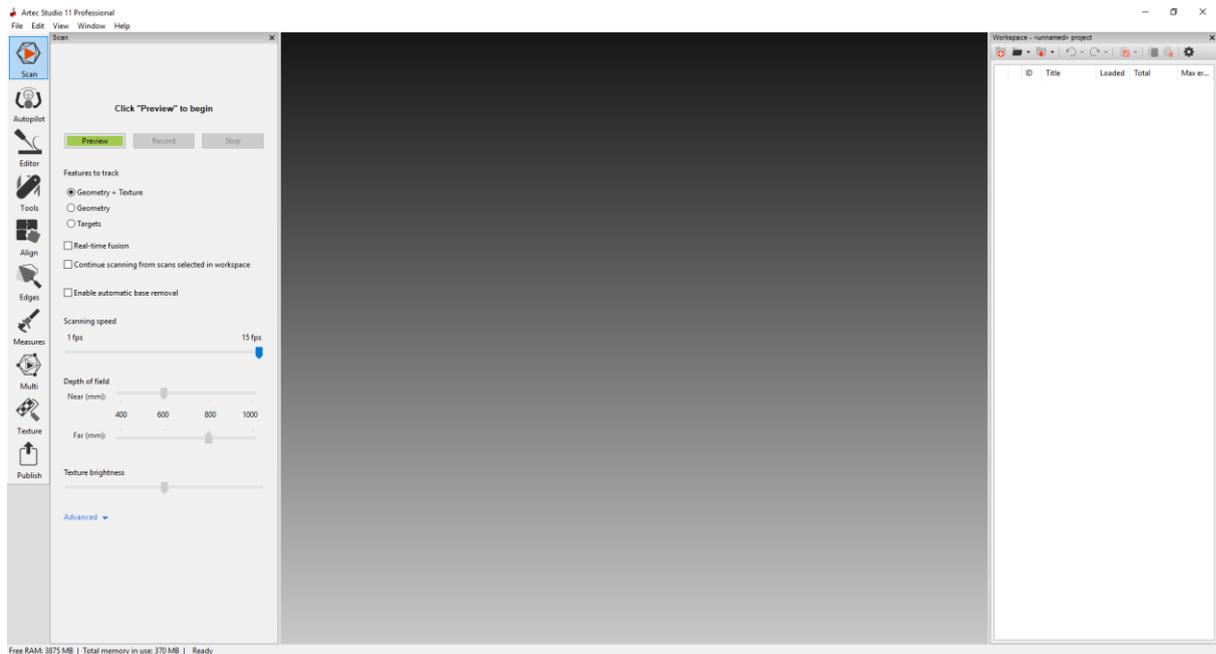


Figura 7 - Página do 3D View do Artec Studio 11



O scanner captura os *frames* de acordo com a frequência configurada antes do início da gravação. Quanto mais rápido melhor, para que o *software* possa sobrepor os *frames* à medida que o modelo vai se movendo, realizando automaticamente a junção destes, não deixando o modelo digitalizado com grandes falhas. O Artec Studio grava superfícies adjacentes, portanto ele faz o rastreamento em tempo real, reconhecendo as superfícies comuns. No entanto, o rastreamento pode se perder (Figura 8), devido a uma rápida movimentação manual da base giratória, ou também por ser modelo com dimensão muito pequena, o que leva o *scanner* a parar a captura da cena.

Figura 8 - Imagem com o rastreamento perdido



Para captura de alguns modelos grandes não era obtido qualidade apropriada para gravação a partir da distância adequada que o *scanner* propõe para abranger todo o modelo. Com isso foi necessário refazer por mais vezes o processo de digitalização. Como o fêmur por exemplo, foi digitalizado primeiramente na posição vertical (vistas dorsal e ventral) na metade de baixo do modelo, e depois invertendo-o, foi capturado a outra metade de cima. E para finalizar foi digitalizado com o osso na posição horizontal (vistas superior e inferior).

Outros ossos são pequenos e difíceis de serem capturados, por isso era necessário posicionar objetos maiores como caixas, próximo ao modelo, para tentar capturar alguma imagem referente a este posicionado à frente. Mesmo utilizando técnica acima, não era possível realizar a captura de alguns modelos, como as falanges proximais, os ossos metacarpos e os ossos metatarsos. Sendo assim, foi necessário realizar a modelagem do osso por imagem de referência conforme já explicado no item 2.1.1.

Com o término de cada digitalização, era gerado um modelo geométrico, estes criados em sequência e identificados por cores definidos automaticamente pelo Artec Studio, podendo abrir cada uma no painel de trabalho, rotacionando, movendo e aplicando zoom para melhor visualização. Neste painel é possível visualizar o número de *frames* capturados, o tamanho do arquivo em MB e o número máximo de erros registrados, verificando assim qualidade que ficou a digitalização. Portanto, quanto menor esse valor, melhor para fazer o alinhamento ao final de todas as digitalizações de cada modelo.

Durante a captura de imagens, o *scanner* grava algumas superfícies que não pertencem ao modelo do osso, como a base da caixa de apoio, a haste de fixação e alguns ruídos. Por isso, antes de criar o objeto geométrico foi necessário realizar alguns ajustes nas digitalizações, deixando o objeto mais próximo possível do modelo real. O *software* realiza automaticamente a limpeza destes ruídos com um algoritmo, porém não consegue detectar todas as partes que precisam ser apagadas. Para aplicar este algoritmo é necessário antes realizar o *Register Global* no Artec Studio, transformando as superfícies em único sistema de coordenadas. Esse processo de remoção se dá em um cálculo de distância média de pontos de acordo com um determinado número de conglomerado de pontos, e também o desvio padrão de cada um. Os pontos em que as médias são maiores que um intervalo pré-determinado são considerados como ruídos e são excluídos.

Após apagar todas as partes desejadas era necessário realizar o alinhamento digitalizações, fazendo a junção de todas as partes, a fim de gerar o modelo geométrico. Desse modo, ao iniciar o processo de alinhamento, selecionamos todas as digitalizações

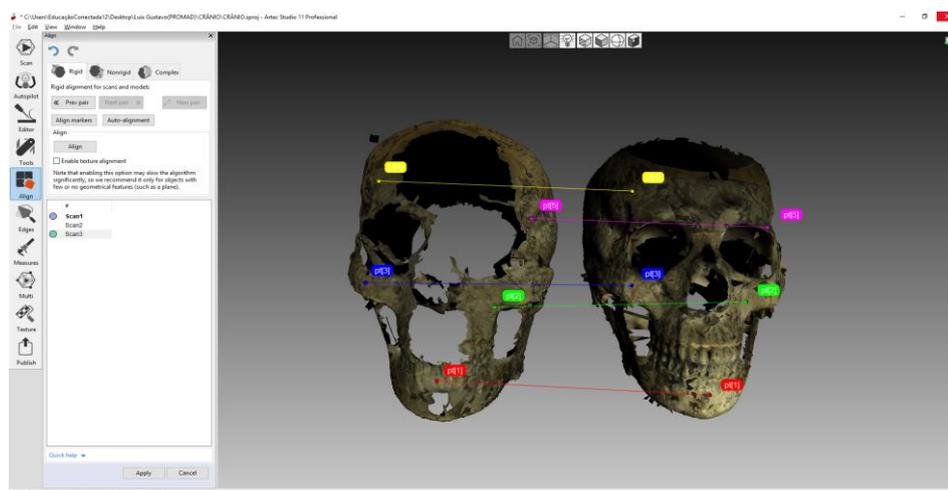
realizadas para formar um modelo, deixando todas como visíveis. O *software* oferece vários tipos de alinhamento, sendo o mais utilizado o alinhamento automático (*Auto-Alignment*), na qual realiza a junção das digitalizações selecionadas automaticamente. Porém, muitos ossos não tinham como obter pontos de referência devido a sua forma anatômica e textura homogênea, sendo necessário afixar em determinados pontos alguns marcadores com fita adesiva branca e pincel colorido conforme ilustrado na Figura 9, facilitando para o *software* realizar o alinhamento automático.

Figura 9 - Modelo de uma vértebra com marcadores para ser digitalizada



Mesmo com o uso dos marcadores, o alinhamento automático não era correto o suficiente, tendo a necessidade de realizá-lo manualmente por pares de pontos, no qual se marca um ponto em uma superfície e outro no mesmo local na segunda superfície digitalizada, conforme mostra a Figura 10, após realizar quantas marcações necessárias, executa o algoritmo e o *software* realiza o alinhamento seguindo as marcações.

Figura 10 - Processo de alinhamento manual



Posteriormente à realização do alinhamento de todas as digitalizações, é necessário aplicar o registro global novamente, selecionando as digitalizações alinhadas e aplicar o algoritmo. Ao executar esse processo é consumido muitos recursos, portanto o processamento de um número grande de digitalizações pode levar um tempo maior, exigindo grande parcela da memória RAM.

No processo final criou-se os modelos 3D utilizando a ferramenta de fusão (*fusion*), transformando em um modelo poligonal, solidificando as superfícies, obtendo um resultado final muitas das vezes satisfatório, pois é a fase na qual visualizamos o objeto 3D desejado. O *software* disponibiliza alguns tipos de fusão: a fusão rápida (*fast fusion*), a fusão suave (*smooth fusion*) e fusão nítida (*Sharp Fusion*). Durante o processo utilizou-se a fusão suave para processar os movimentos leves dos modelos e a fusão nítida para reconstruir e obter as superfícies mais finas.

Após a aplicação da ferramenta de fusão, é possível que o modelo ainda possui alguns erros e imperfeições, como ruídos e buracos na malha, as quais são corrigidas automaticamente pelo Artec Studio, para isso utilizou-se a ferramenta (*Small-object filter*) para remoção desses ruídos. Além disso, o modelo pode ficar com superfícies grosseiras, podendo utilizar a ferramenta de suavização (*Smoothing*) para suavizar as faces do modelo.

O *software* também oferece a opção *Auto Pilot*, na qual selecionando todas as digitalizações, é possível realizar todo esse processo automaticamente, economizando tempo e simplificando o processamento das digitalizações.

Após ter processado todas as digitalizações e deixado o modelo em forma poligonal, exportamos o objeto 3D em formato *.obj*, para que seja importado no *software* MeshLab.

Na Figura 11 é possível visualizar o modelo anatômico do crânio em resina, e na Figura 12 o modelo 3D virtual antes de ser exportado para a fase de retopologia e identificação dos acidentes ósseos.

Figura 11 - Modelo anatômico em resina do crânio



Figura 12 - Modelo virtual 3D

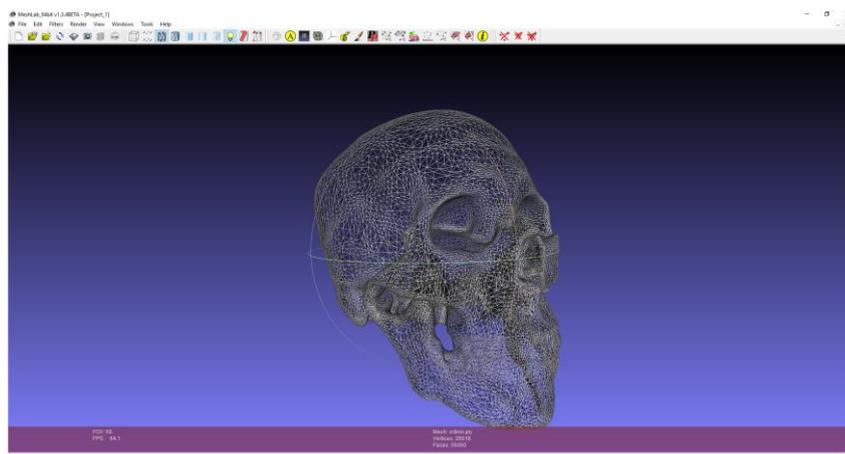


3.3. Retopologia dos modelos

Ao exportar o objeto do Artec Studio, ainda era perceptível um alto número de vértices e desalinhados, arestas duplicadas e faces de diversos tamanhos, arestas duplicadas, exigindo um alto processamento da máquina.

Com o uso do *software* MeshLab citado no item 2.1.1, foi possível realizar uma redução do número de arestas e vértices significativo, diminuindo ainda mais o tamanho do arquivo e suavizando o objeto, sem perder a qualidade e detalhes do modelo 3D. E ainda é possível, o MeshLab também deixa os polígonos de tamanhos uniformes, de forma que facilita posteriormente o processo de identificação dos acidentes ósseos no *Blender 3D*.

Figura 13 - Modelo 3D do crânio exportado do Artec Studio e importado no MeshLab



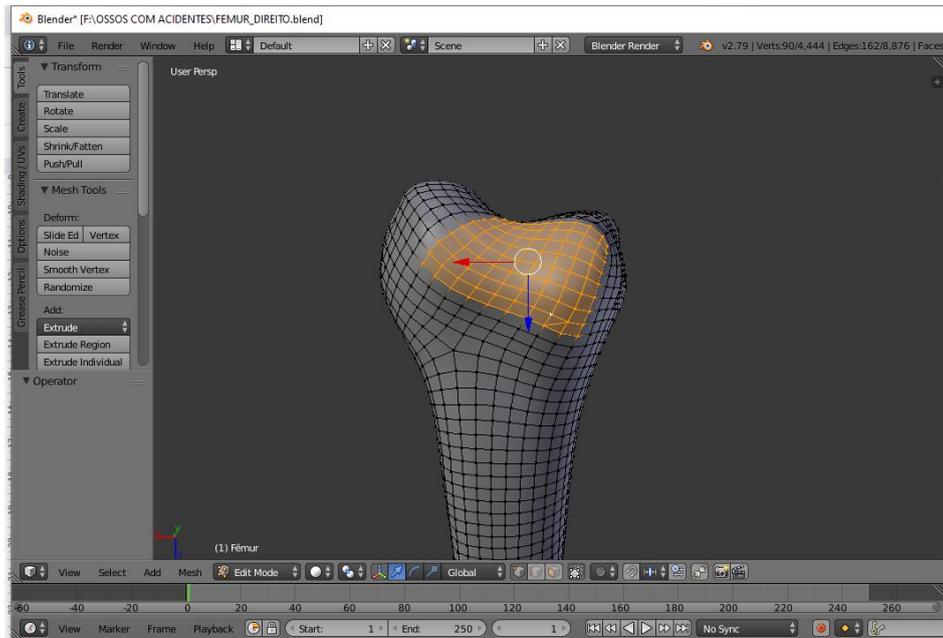
A Figura 13 apresenta o modelo antes da redução dos polígonos. Depois de ser exportado do Artec Studio o modelo estava com 28.018 vértices e 56.060 faces. Após o processo de redução de polígonos no *MeshLab*, houve uma redução para 9.988 vértices e 20000 faces.

3.4. Identificação dos acidentes ósseos

Após realizar todos o processo de redução de polígonos, foi importado os objetos no *software* Blender 3D citado no item 2.1.1, para realizar o mapeamento dos acidentes ósseos. Antes de começar a manipulação, foi necessário centralizar o modelo no *Viewport*, para que o objeto seja rotacionado e movimentado no seu próprio eixo.

Com a orientação de um monitor da disciplina de anatomia humana e com o livro Atlas de Anatomia Humana Sobotta, (PUTZ, PABST 2002) como material de apoio, visualizando melhor os acidentes nas imagens 2D. Para manipular o objeto primeiramente é necessário deixar o objeto em modo de edição (*Edit Mode*), visualizando os vértices, arestas e faces selecionar e manipulá-los. Utilizando técnicas e ferramentas para seleção, houve um trabalho em conjunto, na qual o monitor apontava as áreas para identificação, e com essas técnicas era possível abranger toda a área estipulada.

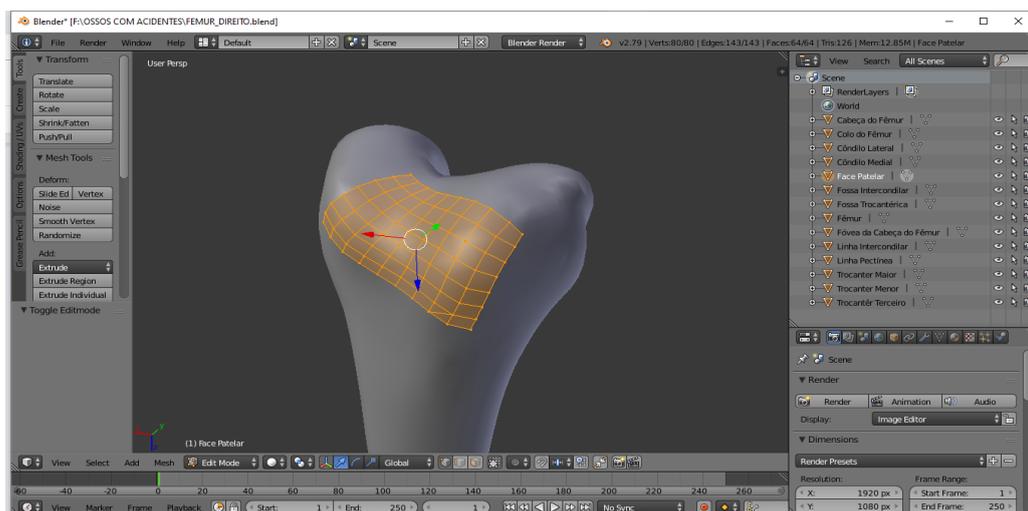
Figura 14 - Área do acidente ósseo Face Patelar do Fêmur selecionado



Após ter selecionado toda a área conforme é ilustrado na Figura 14, duplicou-se todos os vértices deixando na mesma posição, mas um pouco elevado em relação ao objeto maior, formando uma espécie de casca, com o intuito de exibir na tela o nome do acidente, quando o *mouse* colidir com a área identificada.

Com os vértices duplicados, estes eram separados do objeto maior e deixados como objeto filho. E com o gerenciamento de objetos em cena do Blender conforme mostra na Figura 15, ao lado direito superior da tela, sendo possível possível renomear todos os objetos duplicados com o nome científico de cada acidente, para que sejam exibidos na tela do Atlas durante a execução.

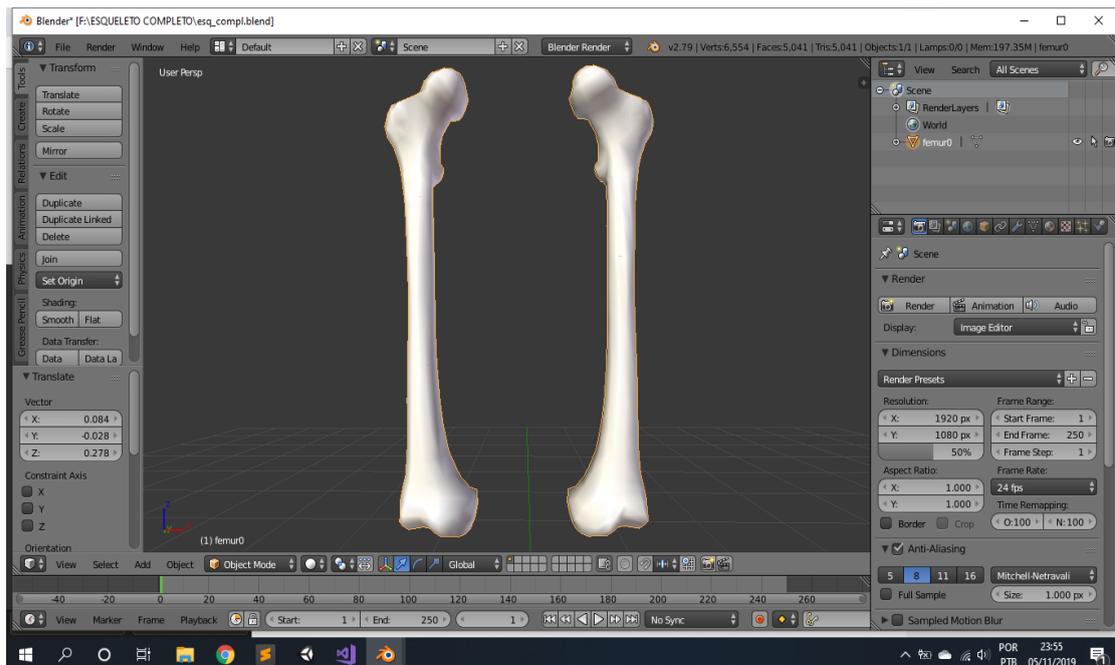
Figura 15 - Acidente ósseo Face Patelar do Fêmur identificado e nomeado



3.4.1. Técnica de Espelhamento

O esqueleto apendicular compõe-se pelos ossos do membro superior e inferior, como por exemplo o fêmur, a tíbia, clavícula e outros, que são osso que estão do lado direito e esquerdo do corpo humano. Para que não tenha um retrabalho e facilitar o processo de modelagem do esqueleto, para cada um desses tipos de ossos foi utilizado a técnica de espelhamento (*Mirror modifier*) no Blender,. Com essa ferramenta foi possível duplicar os ossos 3D com os lado invertidos como ilustrado na Figura 16.

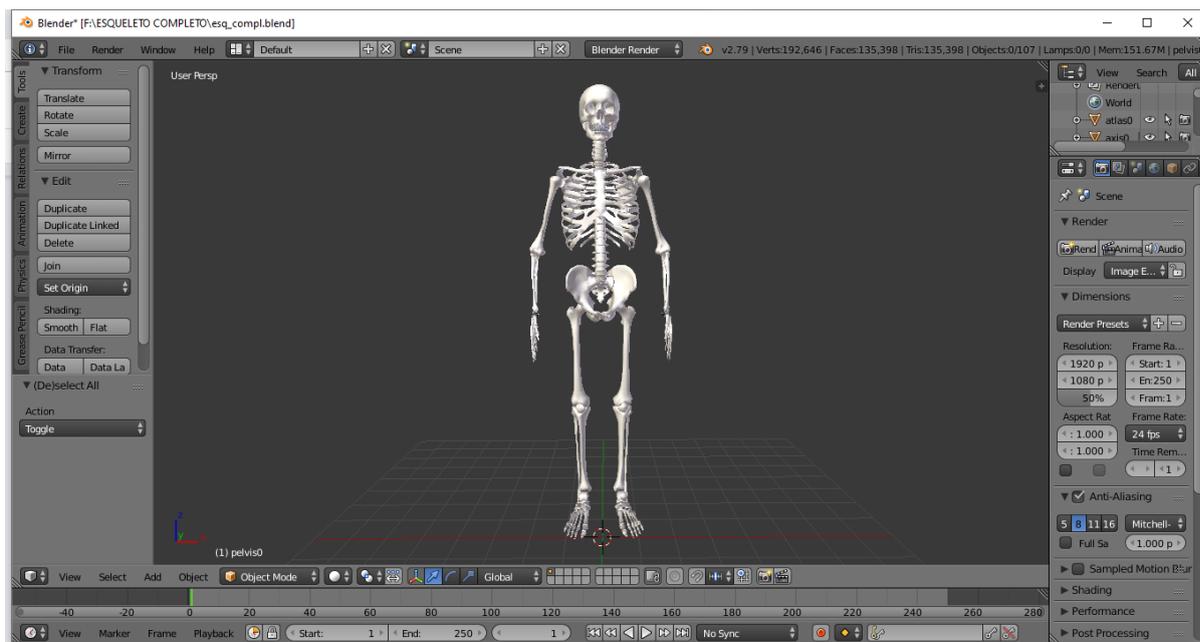
Figura 16 - Espelhamento do Fêmur realizado



3.4.2. Estrutura do esqueleto completo no Blender 3D

De acordo com os requisitos iniciais era necessário visualizar o esqueleto montado completo na página inicial, por isso foi estruturado o esqueleto humano completo a partir das modelagens realizadas (Figura 17), com as orientações de tamanhos e posições do monitor. Porém, o esqueleto completo foi montado a partir dos modelos sem as identificações ósseas, sendo apenas nomeados cada osso para exibição do nome na tela inicial do atlas.

Figura 17 - Modelo do Esqueleto humano completo estruturado no Blender 3D



Com a etapa de identificação de acidente ósseos finalizadas, todos os modelos 3D foram exportados separadamente, gerando os arquivos para que pudessem ser importado no *software* Unity 3D.

3.5. Desenvolvimento da aplicação no Unity 3D

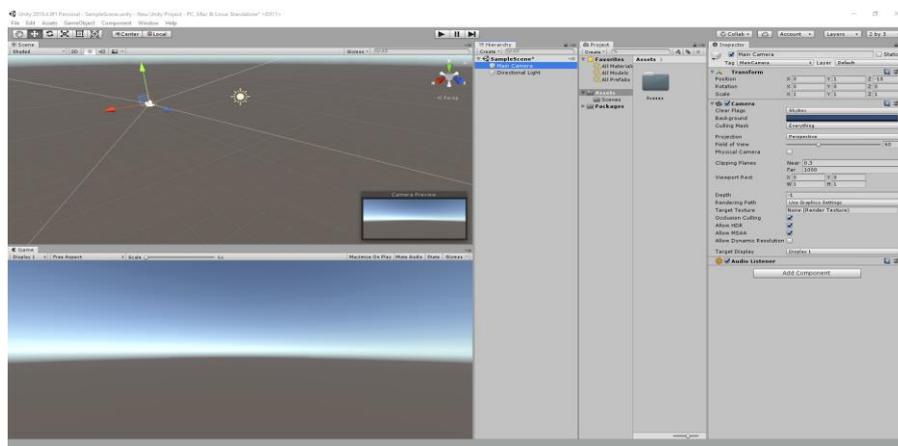
Para o desenvolvimento da aplicação de interação com o usuário foi utilizado o software Unity 3D, conforme mencionado no item 2.1.1, na qual é bem intuitivo para o desenvolvimento de aplicações 3D utilizando metodologias de computação gráfica, com elemento e funções próprias disponibilizadas para facilitar o processo de implementação. Ao executar o Unity 3D, o mesmo já é iniciado com uma cena conforme é ilustrado na Figura 18, sendo exclusiva e de nível único, na qual pode ser inserido qualquer *game object*, componentes, com isso, construindo a aplicação desejada. A cena inicial já possui alguns elementos padrões instanciados, como a câmera (*Main Camera*) e uma lâmpada (*Light*).

O *software* possui alguns *layouts* que ajudam no desenvolvimento, onde já é iniciado com o *layout* 2 por 3, como na Figura 18, dividindo o lado esquerdo em duas partes. Na aba *Scene* na parte de superior esquerda, é possível visualizar a cena completa com todos os elementos, podendo manipulá-los como rotacionar, mover, aumentar ou reduzi-los. A janela *Game* é para execução do projeto, oferecendo uma prévia de como vai se comportar os elementos e suas configurações, ou seja, é a janela na qual vai executar a aplicação podendo realizar testes e modificações em tempo de execução, sem ter que construir as cenas todas as

vezes para testar uma nova função desenvolvida ou adição de um componente em algum *game object*.

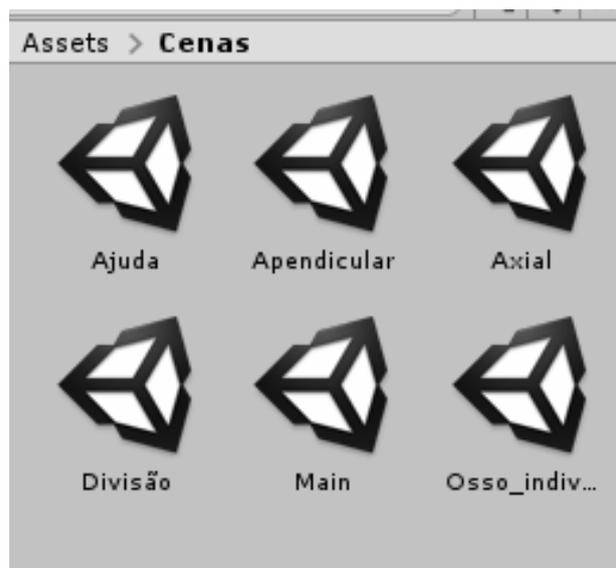
No lado direito da tela é dividido em três partes, na qual poderá visualizar a lista de todos os elementos ativos e inativos da cena na aba *Hierarchy*, na qual pode se configurar a hierarquia dos objetos. Já na aba *Inspector* é listados todas as configurações e componentes adicionados ao elemento selecionado, onde podem ser alterados em tempo de execução, facilitando a fase de desenvolvimento e testes. Na aba *Project* é possível visualizar e manipular todas as pastas e arquivos (*Assets*) do projeto, tais como objetos 3D, materiais, *scripts*, efeitos de áudio e texturas.

Figura 18 - Tela inicial do Unity 3D



Com isso, o desenvolvimento se dá a partir das cenas criadas, que podem ser gerenciadas (Figura 19) a partir da aba *Project*, estas que são ligadas umas às outras, através de botões ou ações desenvolvidas via script, na qual é possível criar, importar qualquer objeto 3D, botões, texturas, materiais em cada cena distinta.

Figura 19 - Gerenciamento de cenas no Unity3D



Ao analisar todos os requisitos funcionais levantados no início do projeto, definimos a partir desse sistema de controle de cenas e quais e quantas seriam necessárias para a aplicação, sendo elas:

- *Scene Main*: esta cena é a inicial, na qual o usuário irá visualizar o esqueleto completo.
- *Scene Osso Individual*: esta cena irá capturar o osso selecionado pelo usuário e trazer todo o conteúdo referente a ele.
- *Scene Divisão*: esta cena dá a opção para o usuário selecionar qual esqueleto deseja estudar, axial ou apendicular.
- *Scene Axial*: cena para estudo individual do esqueleto axial.
- *Scene Apendicular*: cena para estudo individual do esqueleto apendicular.
- *Scene Ajuda*: cena que contém as informações necessárias para que o usuário utilize o *software*.

3.5.1. UI do ADAH3D

O Unity possui um sistema de UI próprio e intuitivo para construção da interface do usuário. É a partir de um *canvas* adicionado a cena, que pode-se adicionar todos os elementos da parte visual da aplicação, e todos os elementos da parte de UI deve ser objetos filho do *canvas*. Portanto, foi construído um padrão de cenas para toda a aplicação, como cores, fontes, botões e painéis de exibição, sendo recriadas em todas as cenas posteriores.

O *canvas* é adicionado como um retângulo que pode ser visualizado na janela *Scene*,

ajudando a visualizar o posicionamento dos elementos na tela, De acordo com essa configuração de hierarquia criamos o padrão para visualização dos elementos nas cenas criadas, com o plano de fundo da cor preta definida na configuração da câmera e os textos e botões com a cor branca. As cenas possuem no canto superior direito o nome do projeto, no campo superior esquerdo tem-se o campo para exibição dos ossos em 3D, campo para exibição dos conteúdos que são buscados do banco de dados, para exibição dos nomes dos ossos e acidentes ósseos, botões com os tipos de vistas, barra de menus, que é visualizada em todas as cenas com o mesmo formato e com as seguintes opções respectivamente:

- *Home* - botão que carrega a cena principal
- *Divisão* - cena para seleção de qual esqueleto deseja visualizar;
- *Ajuda* - cena com as informações necessárias para utilização da aplicação;
- *Sair* - botão para finalizar a execução.

Após ter criado a interface do usuário, foi importado primeiramente na cena principal o modelo do esqueleto completo, posicionando-o em (0, 0, 0), para que seja exibido no centro da tela. Os ossos que compõem o esqueleto humano são objetos filhos do *game object* Sistema Esquelético. Sendo o mesmo para os modelos com os acidentes ósseos identificados, na qual as malhas identificadas e nomeadas são objetos filhos de cada osso 3D.

Para que aconteça a interação dos usuários com os objeto, rotacionando, movimentando, acionando alguma ação necessária para os elementos da cena a partir dos dispositivos de entrada, foi necessário adicionar ao projeto um *EventSystem*, responsável por enviar para o objeto a ação solicitada.

É preciso também adicionar alguns componentes aos objetos inseridos na cena, para a execução da aplicação. Portanto, Para que o objeto possa ser manipulado via *script* foi preciso adicionar o *Mesh Filter*, obtendo o acesso todos os componentes e atributos do objeto, como por exemplo ativar e inativar na cena via *script*. O *Mesh Renderer*, para renderizar as malhas adicionadas pelo *Mesh Filter*. E também para que seja identificado o nome do osso ou do acidente ósseo é necessário que tenha o componente *Mesh Collider*, que rastreia se o ponteiro do *mouse* está posicionado sobre o objeto. Todos esses componentes devem ser adicionados aos objetos de interação da cena.

O *script* *AcidentesGeral*, foi adicionado para todos os objetos filhos dos ossos 3D, para que realize a exibição do nome do osso ou do acidente ósseo para os modelos identificados, caso o *mouse* esteja sobre a malha do modelo 3D. Com o método

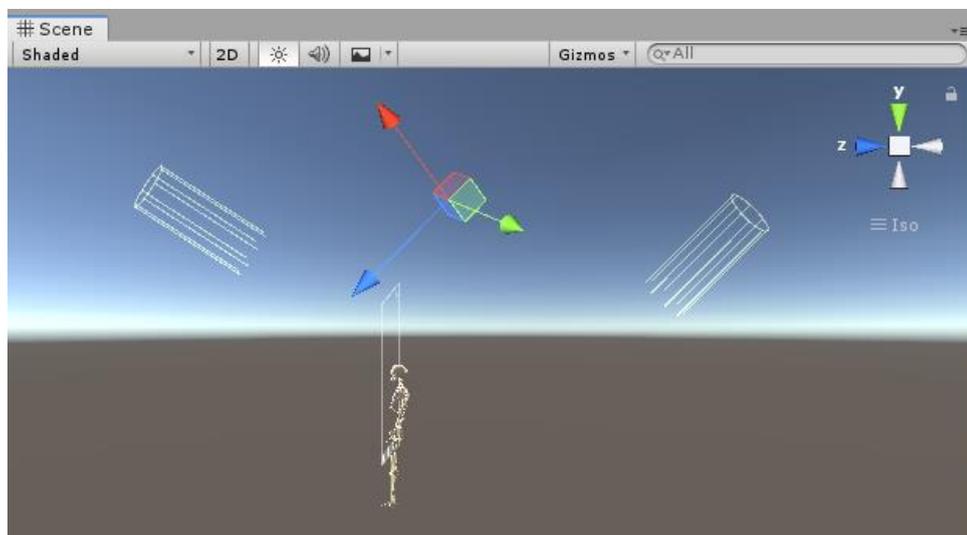
OnMouseOver que ao ser chamado é criando um objeto da classe *NomeDoOsso* do tipo *string* que é atribuído a câmera. O nome do osso na qual o ponteiro do *mouse* está sobre a superfície, é exibido em um painel na cena conforme ilustrado na Figura 20. Quando o *mouse* sai de cima da malha do objeto, é chamado o método *onMouseDown* também da classe *AcidentesGeral*, na qual atribui-se vazio à essa *string*.

Figura 20 - Painel exibindo a identificação com o mouse do fêmur



Além de toda a estrutura de interface programada com as ferramentas de UI do Unity a partir do *canvas*, foram adicionadas duas lâmpadas (*Directional Light*) nas cenas que contém modelos de ossos 3D para interação, uma na frente do objeto e outra atrás, para que a iluminação da cena seja uniforme, como é ilustrado na Figura 21. Melhorando assim, a visualização dos modelos.

Figura 21 - Lâmpadas posicionadas na cena para visualização dos modelos 3D



3.5.2. Processo de Rotação, Translação e *zoom* da câmera

Para que o usuário tenha uma visualização 360° do *game object*, foi desenvolvido um mecanismo de manipulação da câmera, na qual o objeto fique imóvel e somente a câmera rotacione em todos os ângulos, com o objetivo de diminuir o custo de processamento, utilizando o método *Screen-based*, conforme foi explicado no item 2.1.

Para que a rotação aconteça corretamente em torno do modelo dos ossos, foi criado um *game object empty* (objeto vazio) denominado como “CameraRotacao”, sendo invisível na cena, contendo apenas as configurações de posição, rotação e escala, sendo adicionado somente o *script* Rotação, para que o mesmo seja rotacionado em seu próprio eixo.

O *game object* CameraRotacao e os modelos ósseos 3D importados na cena foram posicionados no plano com a posição (0, 0, 0) conforme é ilustrado na Figura 22, e a câmera adicionada como objeto filho do objeto CameraRotacao, foi fixada em 225 no eixo z, deixando-a com uma distância razoável do modelo 3D, para que este seja exibido inteiramente na tela de acordo com a Figura 23. Portanto, quando o objeto vazio realizar o processo de rotação, a câmera irá alterar sua posição nos três eixos acompanhando o objeto pai, deixando que a câmera realize o movimento de câmera orbital sobre o objeto 3D que está instanciado na cena, propiciando a impressão de rotação do modelo.

Figura 22 - Objeto vazio para rotação da câmera e suas configurações

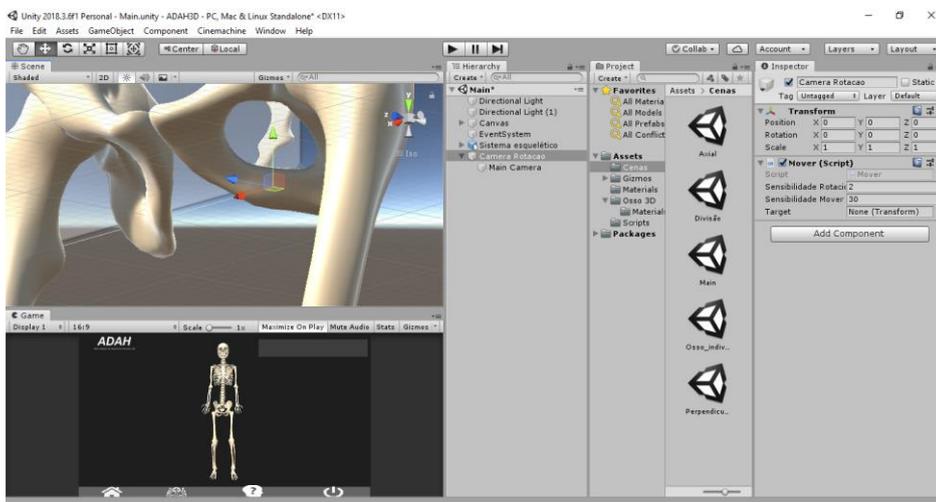
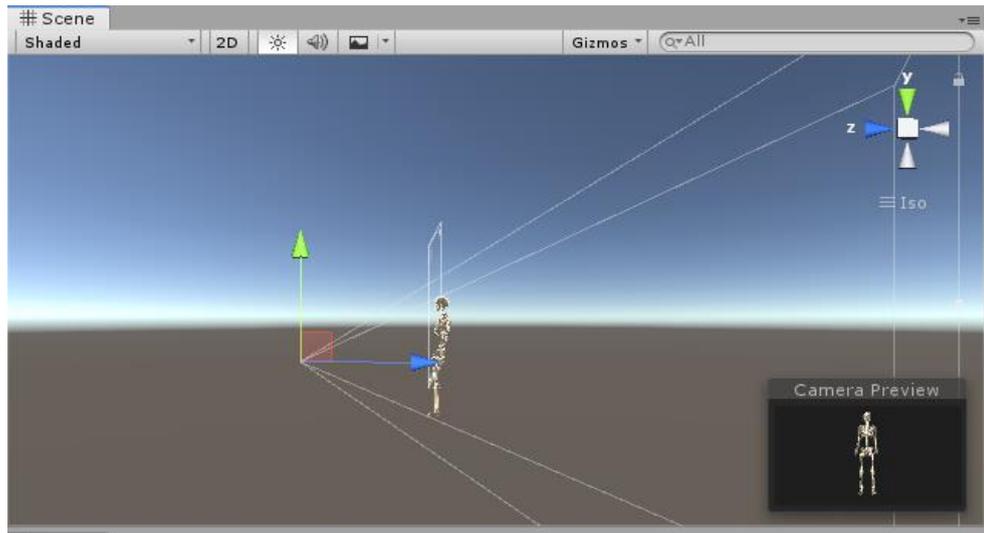


Figura 23 - Câmera posicionada para visualização da cena com o objeto.



Para que este processo de rotação mencionado acima aconteça corretamente, o script Rotacionar, é executado durante todo o processo de execução da aplicação, instanciando a sensibilidade rotacionar e as posições do *mouse* em relação aos eixos x e y. Com o botão esquerdo do *mouse* pressionado, ao movê-lo na horizontal obtém-se a rotação em torno do eixo vertical e na vertical obtendo a rotação sobre o eixo horizontal, na qual tem direções coincidentes à tela (x e y).

Com o uso de API, foi utilizado o método *Input.GetAxis*, na qual retorna o valor da posição do eixo virtual calculada pela posição do *mouse*, onde retornará o valor entre -1 e 1. Porém, este valor é multiplicado pela sensibilidade que foi instanciada no início do código com o valor de 30.0f, para que a rotação tenha maior velocidade e assim melhorando a interatividade do usuário.

Foi criado um script para ser aplicado somente a câmera para aproximação do modelo 3D, ou seja, aplicar o *zoom* ao objeto, alterando somente a posição do eixo z da câmera, pois o objeto pai, CameraRotacao, deve permanecer no centro do objeto.

Para que o *zoom* funcione corretamente é calculado a distância de acordo com a rolagem do *scroll* do *mouse* através do método *Input.GetAxis*. Porém, este *zoom* deve ter um limite, para que não ultrapasse o objeto que está em cena. Por isso foi definido um valor máximo e mínimo de *zoom*. Com o método *Math.Clamp* específico da Unity, facilita este processo de bloqueio da posição, define-se a variável que deve ser restringida, no caso, o *zoomAtual* e também o valor mínimo e máximo de *zoom*. Desse modo, a função não permite que a posição Z ultrapasse o valor definido nas variáveis controladoras.

Ao realizar o processo de *zoom* na câmera mencionado acima, o usuário necessita de movimentar a câmera na horizontal e vertical para poder visualizar outras partes do osso com o *zoom* aplicado. Por isso é aplicado na câmera o *script* Mover, para que somente a câmera se mova horizontalmente e verticalmente, ou seja, somente nos eixos x e y que são coincidentes a tela do dispositivo, deixando o objeto pai CameraRotacao imobilizado no centro. Para que o movimento da câmera seja acionado, o usuário necessita manter pressionado o botão do *scroll* do *mouse*, e ao movimentar o *mouse* a câmera detecta a posição do eixo virtual, identificando os valores correspondentes aos eixos, x e y, e aplicando através do método *transform.position*, instanciando um *Vector3*. É passado o valor de alteração dos eixos correspondentes, multiplicando por “*senbilidadeMover*”, para que possa aumentar a velocidade do movimento da câmera, e mantendo o valor de Z igual a 0.0f.

Conforme Castier, Martha e Gattass (1994), espera-se que com o movimento horizontal do *mouse*, para o lado esquerdo, o objeto também se movimente para a esquerda, e caso movimente para o lado direito, o objeto também siga para o lado direito. E o mesmo pode ser esperado na vertical. Caso o usuário o mova para cima, espera-se que o objeto movimente-se para cima, assim também para o sentido inverso. Mas como não é o objeto que está sendo movimentado e sim a câmera, e a mesma está posicionado a frente do objeto, como um espelho, é preciso realizar o movimento inverso para que o movimento natural esperado aconteça. Para isso, invertamos o sentido do eixo da câmera com o eixo virtual, invertendo os valores do sinal no *script*.

O movimento da câmera também pode ser acionado quando pressionadas as teclas a seguir:

- *Up Arrow* - move a câmera para cima;
- *Down Arrow* - move a câmera para baixo;
- *Right Arrow* - move a câmera para direita;
- *Left Arrow* - move a câmera para esquerda.

Para a ação de mover a câmera também é necessário que tenha um limite das posições dos eixos que estão sendo manipulados, *para que o objeto não deixe o campo de visão da câmera*. Por isso, durante todo o processo de execução é verificado a posição da câmera nos eixos x e y. No eixo x, ultrapasse de 35 e menor que -35, através do método *transform.position* atualizamos a posição de x para 35 e -35 respectivamente, e as posições de y e z são permanecidas com o valor atual. E no eixo Y é verificado se é maior que 100 e menor que -100, caso esteja fora desse intervalo o eixo y é atualizado para 100 e -100,

respectivamente, deixando a posição do eixo x e z na posição atual.

Para todas as cenas que possuem manipulação de um modelo 3D foi replicado o objeto *CameraRotacao* e a câmera principal, com todos os *scripts* aplicados como componentes destes objetos, e configurações realizadas, pois o funcionamento deste processo será a mesma para todas as cenas da aplicação.

3.5.3. Cena para visualização de modelos e conteúdos específicos

A cena *Osso_Individual*, para estudos específicos de cada osso, foi importado todos os elementos que compõem a cena principal, como o canvas, câmera e lâmpadas, exceto o esqueleto completo. Com todos os elementos e configurações duplicados para a nova cena. Foi importado também todos os modelos 3D com os acidentes ósseo identificados, sendo que todos devem estar inativados. Estes deverão ser ativados somente quando o usuário selecionar o objeto desejado para estudo, que estão na posição $(x=0, y=0, z=0)$ permitindo a visão Ventral inicial, e com uma escala de $(x = 1, y=1, z=1)$.

Para que o objeto selecionado pelo usuário na cena inicial seja ativado na cena *Osso_individual*, para estudos específicos, foi adicionado em cada objeto filho do esqueleto completo da cena principal o *script* *MudarCena*, na qual é identificado qual modelo o usuário selecionou. Com o nome do osso selecionado é buscado na cena um objeto com o mesmo nome, e ao acessar os componentes do mesmo é possível setar o atributo *setActive* como *true*, exibindo o modelo desejado na cena.

Com a cena *Osso_Individual* em execução com um modelo ativo, na qual o usuário selecionou, é preciso exibir na tela todo o conteúdo referente ao mesmo. Para isso, é aberto uma conexão com o banco de dados, que é explicado na seção 3.3, buscando na tabela de conteúdo textual qual o texto para o respectivo modelo e vista na qual está selecionada. Ao buscar o texto, este é instanciado em uma variável *conteudoOsso* e enviado para a cena para ser exibido no painel abaixo do painel de identificação dos nomes dos ossos e acidentes.

E para a manipulação do osso, o usuário tem a possibilidade de rotação e movimentação livre com o *mouse*, da mesma forma que na cena principal. Mas nesta cena também é possível visualizar o modelo a partir de algumas vistas já definidas na aplicação:

- Dorsal;
- Ventral;
- Superior;

- Inferior;
- Lateral Esquerdo;
- Lateral direito.

A partir do script Vistas foram definidas posições para cada botão disponível na cena, sendo um para cada vista, conforme é ilustrado na Figura 24. Ao selecionar uma destas, a câmera é translada para a posição desejada, exibindo o modelo 3D conforme a seleção e é exibido o texto referente a esta vista.

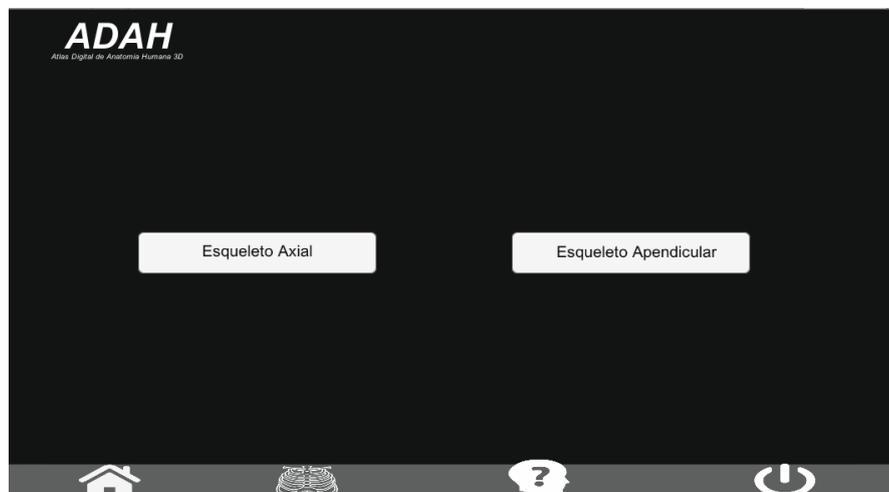
Figura 24 - Botões para seleção de vistas dos modelos



3.5.4. Esqueleto Axial e Apendicular

Na barra de menus, ao pressionar a segunda opção é carregada a cena Divisão, na qual o usuário tem a possibilidade de selecionar através de um botão qual esqueleto deseja abrir especificamente conforme ilustrado na Figura 25, podendo visualizá-los separadamente.

Figura 25 - Cena com os botões para seleção da divisão do esqueleto



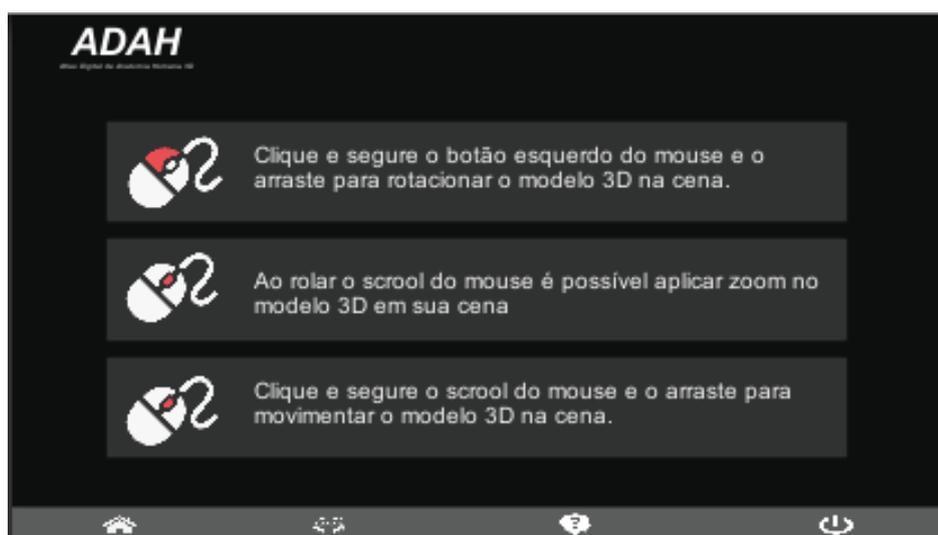
A partir da opção selecionada através de um botão, é carregada a cena específica para contendo somente os osso do esqueleto selecionado. Também é possível a partir dessa cena selecionar os ossos para estudos específicos, e também a opção de livre rotação e zoom do

modelo ativo na cena.

3.5.5. Cena de Ajuda

Caso o usuário tenha alguma dúvida em relação a utilização da aplicação, ele conta com uma cena somente para explicações de como usar o ADAH 3D, que pode ser acessada de qualquer cena, clicando no botão de ajuda na barra de menu, com o ícone de ponto de interrogação. Nessa tela contém textos e imagens de explicação do manuseio do *mouse* para manipulação dos modelos.

Figura 26 - Cena de Ajuda do ADAH3D



3.5.6. Execução e Disponibilização do ADAH3D

Para que toda a aplicação seja executada e exportada, gerando o arquivo executável, é necessário construir a aplicação com todas as cenas desejadas. Durante a fase de implementação e testes, a aplicação foi executada para a plataforma *Universal Windows*, para que possa testar todas as funcionalidades sem ter que abrir um navegador, facilitando o processo de implementação. Mas ao ADAH3D será disponibilizado na web para os usuários finais sendo possível acessar através de um navegador, por isso a plataforma foi executada e gerada o arquivo final para a plataforma WebGL, que permite executar e visualizar imagens 3D, através de um *canvas*, executando em um navegador web o projeto criado no Unity

3.6. Desenvolvimento do módulo de gerenciamento de conteúdo

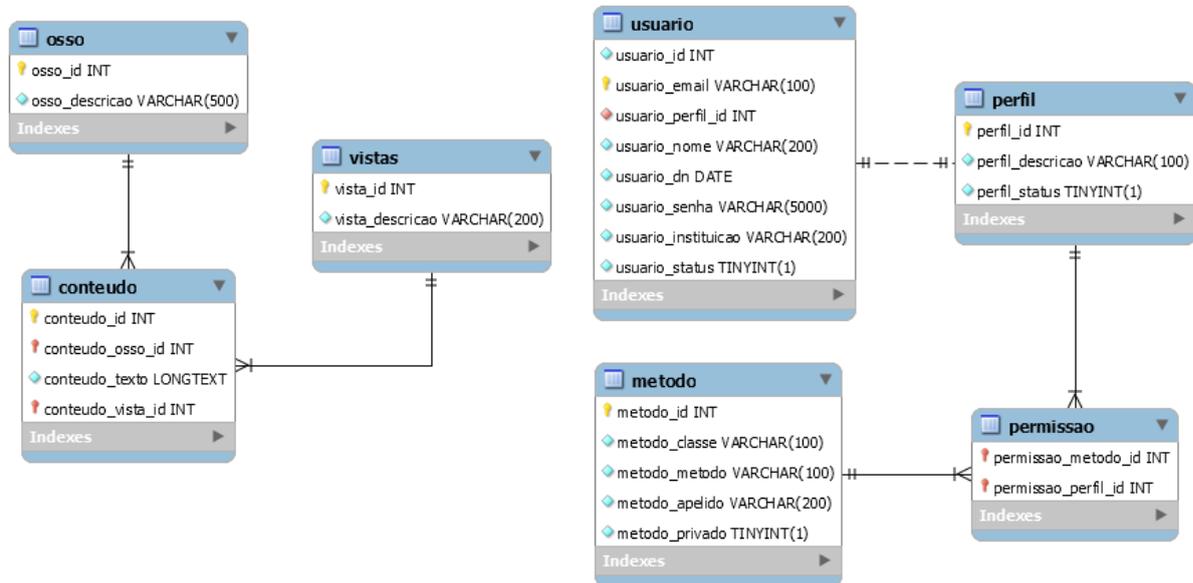
Para que o *software* esteja atualizado é necessário que tenha um módulo de gerenciamento do conteúdo que compõe o ADAH3D, para que o administrador da aplicação,

não tenha uma dependência do desenvolvedor para cadastrar ou atualizar algum conteúdo.

3.6.1. Banco de Dados

Foram criadas três tabelas para gerenciar o conteúdo da aplicação e quatro tabelas para gerenciar os usuários e suas permissões de acessos, conforme é ilustrado na Figura 27.

Figura 27 - Tabelas do Banco de Dados Relacional do ADAH 3D



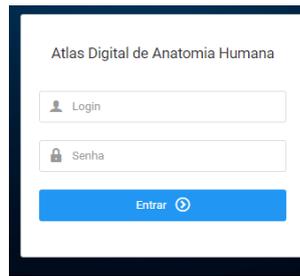
Para inserir um conteúdo no sistema, é preciso ter no mínimo um cadastro de vista, e outro de osso. A partir dos ID's destes é possível cadastrar na tabela conteúdo, todo o texto referente ao osso e vista selecionado.

E para o cadastro de usuário é necessário os dados pessoais, tais como nome, e-mail, senha, data de nascimento, instituição, e é relacionado ao usuário uma chave estrangeira do tipo de perfil. Este pode ser Administrador ou Usuário comum, na qual na tabela permissões serão cadastrados quais métodos e classe este perfil tem permissão para executar.

3.7. Acesso e controle de permissões do ADAH 3D

Para acessar o Atlas Digital de Anatomia Humana é preciso acessar o link <http://www.adah.ufla.br>, que irá ser redirecionado para a página de *login*, na qual o usuário que já é cadastrado poderá logar com seus dados como é ilustrado na Figura 28. E caso ainda não tenha um registro, o usuário será redirecionado para tela de cadastro de usuário, na qual deverá preencher os campos com seus dados e finalizar o cadastro, após isso o mesmo já poderá acessar o Atlas.

Figura 28 - Tela de Login do ADAH 3D

A imagem mostra a interface de login do sistema ADAH 3D. No topo, há o título "Atlas Digital de Anatomia Humana". Abaixo dele, há dois campos de entrada: o primeiro é rotulado "Login" com um ícone de usuário e o segundo é rotulado "Senha" com um ícone de cadeado. Abaixo dos campos, há um botão azul com o texto "Entrar" e um ícone de seta para a direita.

Em todas as páginas do sistema de gerenciamento há um script, que é chamado sempre quando uma classe do *Controller* é instanciada, na qual no construtor da classe verifica se possui uma sessão iniciada de algum usuário no navegador. Caso possui uma sessão válida, o método da classe é executado.

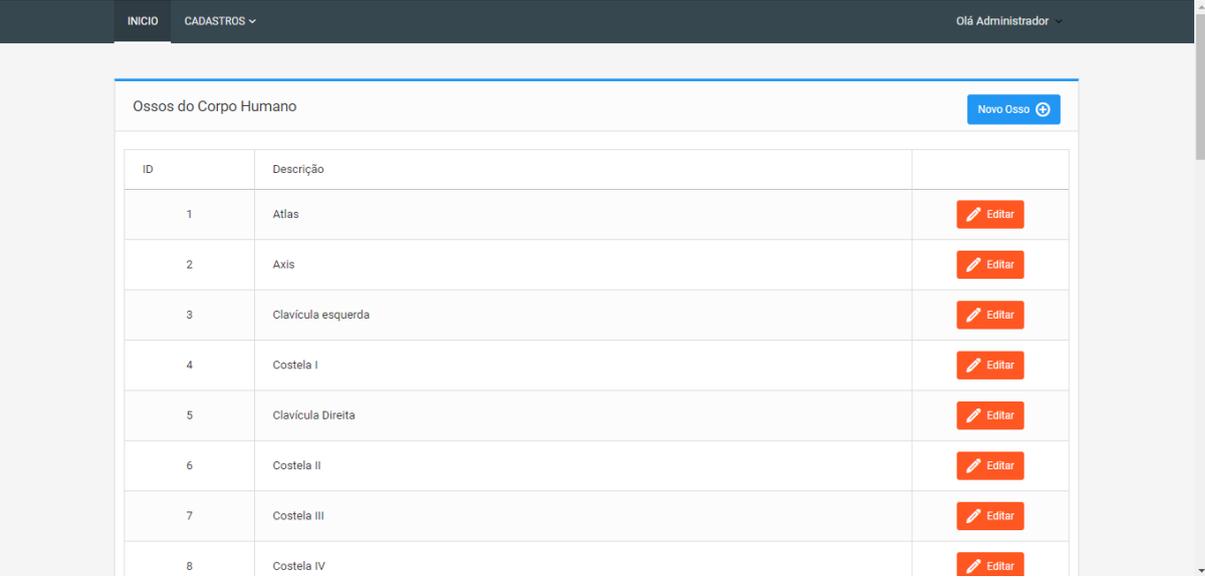
Porém, antes do método ser executado, é verificado se o usuário possui permissão. Para isso é chamado um método da classe *Auth*, que verifica se possui liberação da classe e método na tabela permissão para o perfil do usuário logado. Caso tenha a permissão, o usuário é redirecionado para a página solicitada, senão voltará para a página inicial.

O usuário que possui o perfil de Administrador terá mais funcionalidades que um usuário comum, onde poderá gerenciar todos os cadastros de conteúdos, ossos e também o controle de usuários. Antes que esses menus sejam exibidos na página do administrador, é verificado por uma função PHP, buscando no banco de dados se possui a permissão para tal funcionalidade. Caso tenha é retornado o valor 1 e o menu com link será exibido para ser acessado. Caso contrário, não será exibido na tela para o usuário.

3.8. Cadastros de Ossos, Conteúdo e Usuários

Na página inicial de cadastro de ossos, é exibido todos os ossos já cadastrados (Figura 33), e caso não tenha nenhum é exibido uma tabela vazia. Com o botão de cadastro de “Novo Osso”, localizado no canto superior direito (Figura 33), é possível realizar o cadastro em um modal exibido na parte superior da tela, preenchendo apenas a descrição do osso e selecionando o status.

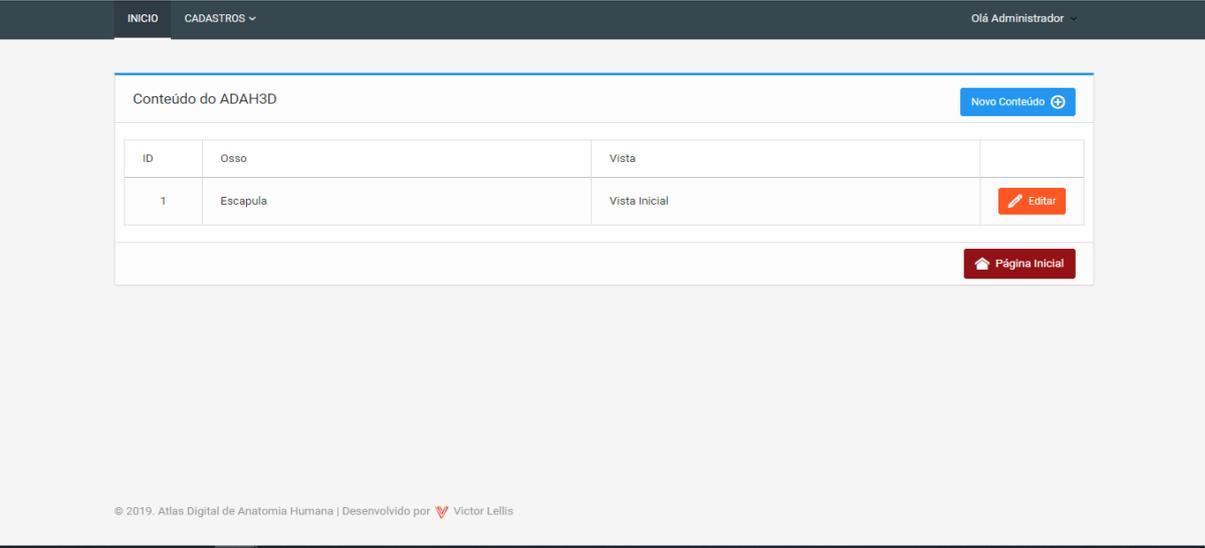
Figura 29 - Página de cadastro de Osso do módulo de gerenciamento



ID	Descrição	
1	Atlas	Editar
2	Axis	Editar
3	Clavicula esquerda	Editar
4	Costela I	Editar
5	Clavicula Direita	Editar
6	Costela II	Editar
7	Costela III	Editar
8	Costela IV	Editar

A página de cadastro de conteúdo (Figura 34) é semelhante a de cadastros de ossos, sendo exibido uma tabela com os cadastros já realizados. Para realizar um novo cadastro, o usuário tem disponível também um botão, que ao ser clicado é aberto um modal de cadastro, conforme ilustrado na Figura 36, onde o usuário precisa selecionar o osso, a vista, e descrever o conteúdo no campo descrição.

Figura 30 - Página de cadastro e gerenciamento de conteúdo dos modelos

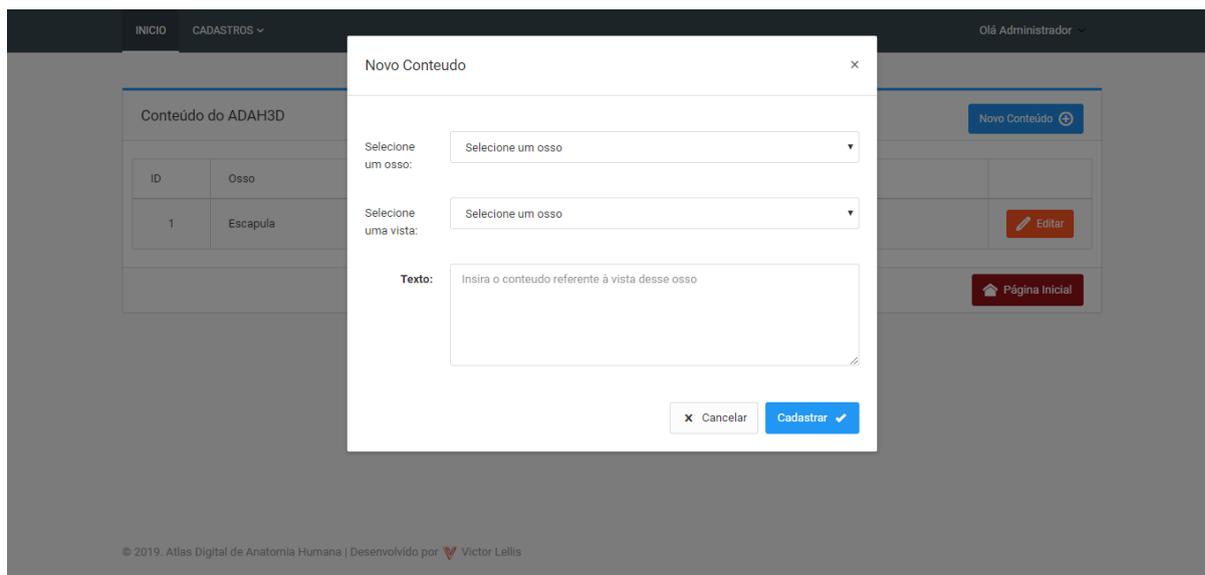


ID	Osso	Vista	
1	Escapula	Vista Inicial	Editar

[Página Inicial](#)

© 2019. Atlas Digital de Anatomia Humana | Desenvolvido por  Victor Lellis

Figura 31 - Página com o modal para cadastro de conteúdo



Para ambos os tipos de cadastros também é possível realizar a edição dos dados cadastrados. As opções de editar são exibidas para cada linha de dados exibido na página, passando para o modal de edição o ID do cadastro selecionado e preenchendo os campos enviados para o modal para serem editados.

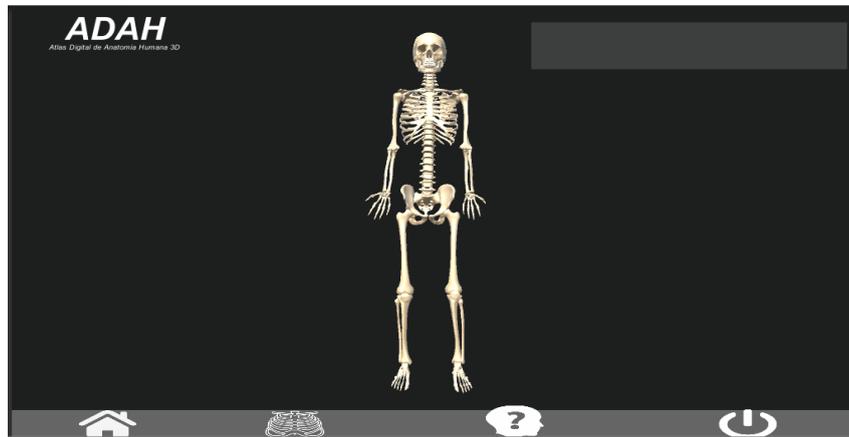
O usuário administrador também poderá realizar o cadastro de usuários pelo módulo de gerenciamento de conteúdo, podendo inserir novos usuários e editar os usuários que já são cadastrados. Para o cadastro e alteração de senha, o sistema envia para o *controller* via *post*, o conteúdo do *input* `usuario_senha`, enviado para o método de criptografia própria do codeigniter a partir de uma chave definida nas configurações, na qual o sistema descriptografa utilizando esta chave para verificar a autenticidade da senha antes do *login*.

4. RESULTADOS

O Atlas Digital de Anatomia Humana 3D proporciona uma interface para o usuário bem intuitiva com uma fácil navegação, possibilitando o estudo do sistema esquelético do corpo humano com conteúdos específicos e acidentes ósseos de cada modelo.

Ao executar, a aplicação é iniciada com a cena Principal conforme ilustrado na Figura 32, onde o usuário visualiza o esqueleto completo e tem a possibilidade de selecionar algum modelo para estudo individual, assim como foi solicitado no levantamento de requisitos. O usuário possui várias possibilidades de visualização do modelo 3D, podendo rotacionar e mover livremente com o uso do *mouse* e do teclado, E também podendo navegar pela aplicação com os botões da barra de menus.

Figura 32 - Cena inicial do ADAH 3D



Conforme é ilustrado na Figura 33, a cena de estudos específicos do modelo 3D do osso, é exibido apenas o osso que foi selecionado na cena principal. No canto superior direito é exibido acima o nome do acidente ósseo onde o *mouse* está posicionado (na imagem o mouse está posicionado sobre a cabeça do fêmur). Logo abaixo, está o texto referente à vista Ventral que foi selecionada nas opções de tipo de vistas.

Figura 33 - Cena para visualização individual do modelo 3D do Fêmur na vista Ventral



É possível também através da barra de menus acessar a cena de divisão do esqueleto humano, sendo eles: o esqueleto axial (Figura 34) e esqueleto Apendicular (Figura 35). Em cada cena é possível visualizar apenas os ossos de cada um e também sendo possível interagir com o modelo assim como nas outras cenas. A partir do modelo carregado na cena o usuário também pode selecionar algum osso 3D para estudo específico, sendo encaminhado para a cena conforme ilustra na Figura 36.

Figura 34 - Cena do esqueleto Axial com o mouse sobre o crânio

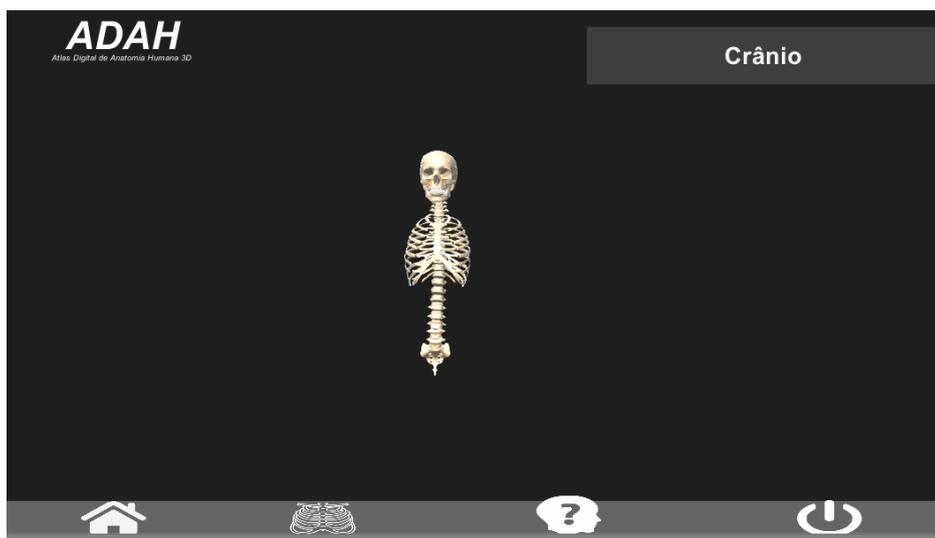
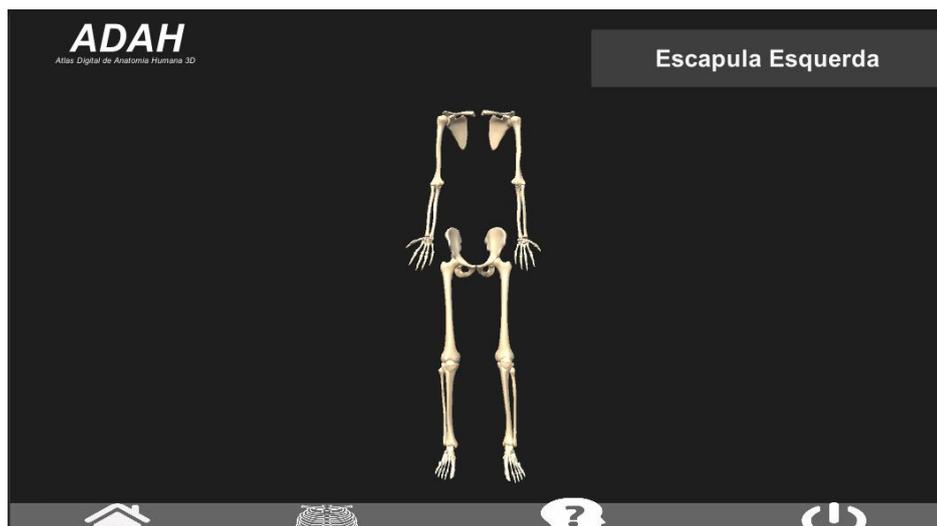
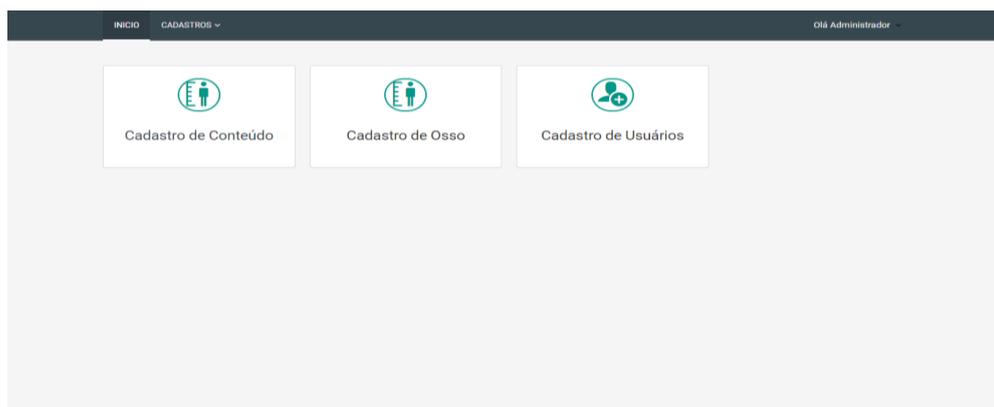


Figura 35 - Cena do esqueleto Apendicular com o mouse sobre a Escápula Esquerda



E para o controle de conteúdo, ossos e de usuários foi possível criar o sistema web com todas as características solicitadas no levantamento de requisitos, podendo ser acessado de qualquer dispositivo com acesso à internet. O usuário administrador, ao entrar no sistema de gerenciamento terá acesso às opções de gerenciamento, conforme ilustrado na Figura 36.

Figura 36 - Página Inicial do módulo de gerenciamento do administrador



Com isso foi possível a disponibilização de uma aplicação 3D para que o usuário possa realizar um estudo completo do sistema esquelético do corpo humano, interagindo com os modelos tridimensionais de forma mais real possível.

5. CONCLUSÕES E TRABALHOS FUTUROS

Como resultado final foi desenvolvido uma aplicação para estudo de anatomia humana com modelos tridimensionais, por meio da utilização de *software* livre e de códigos abertos, atendendo os requisitos inicialmente levantados.

Foram digitalizados todos os ossos do esqueleto axial e todos os ossos do esqueleto apendicular do lado direito, sendo que o lado esquerdo foi aplicado a técnica de espelhamento. O processo de digitalização, retopologia da malha do modelo, identificação dos acidentes ósseos e estruturação do esqueleto completo no Blender 3D, foram necessários 2 anos e 3 meses, sendo 12 horas semanais, totalizando aproximadamente 1.400 horas para o processo total de modelagem. Com isso, foram modelados um total de 206 ossos do sistema esquelético do corpo humano.

Com essa aplicação é possível efetuar os estudos necessários para os alunos da disciplina da Anatomia Humana da UFLA, dentro e fora da sala de aula, auxiliando também os professores nas ministrações de suas aulas práticas ou teóricas, pois o sistema pode ser acessado de qualquer computador que tenha acesso a internet.

Através da realidade virtual, foi possível desenvolver uma aplicação 3D, na qual é permitida a interação do usuário com a aplicação, tendo mais uma opção diferenciada para estudo de anatomia humana, podendo ser mais atrativo para o aluno ajudando-o na exploração de conteúdo de anatomia.

E com o término do desenvolvimento do projeto, foi possível perceber a importância de utilizar um processo de desenvolvimento a ser seguido, com base em uma metodologia, pois assim pode-se obter uma maior qualidade do *software*, de acordo com o gerenciamento das atividades do processo definido.

Em relação ao conteúdo do ADAH3D, ainda está em fase de produção pela monitora da disciplina de anatomia humana juntamente com a Dra. Giancarla Aparecida Botelho Santos. Devido a este motivo a fase de testes de usabilidade ainda não foram aplicados aos usuários finais, alunos da disciplina de anatomia humana.

Para trabalhos futuros, propõe-se a continuidade da implementação dos outros sistemas do corpo humano, como sistema muscular, nervoso, cardiovascular, entre outros, realizando a modelagem 3D destes sistemas e incrementando nas cenas já disponibilizadas e o cadastros dos conteúdo necessários, pode ser desenvolvido também uma funcionalidade de perguntas e respostas, como um *quiz* para melhor fixação do conteúdo.

REFERÊNCIAS

- SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Education do Brasil, 2011.
- PUTZ, R.; PABST, R. **Sobotta, Atlas de Anatomia Humana**. Vol. 1 e 2. 22 ed. Rio de Janeiro: Guanabara Koogan, 2002.
- CASTIER, B.; MARTHA, L. F.; GATTASS, M. **Uma Taxonomia para Manipulação Interativa e Visualização de Objetos 3D**, Sibgrapi' 94 - VII Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens, Curitiba, PR, pp. 149-156, Nov. 1994.
- MANSSOUR, I. H.; COHEN, M.; **Introdução à Computação Gráfica**, RITA. 13. 43-68, 2006.
- BLENDER, Blender Reference Manual. 2017. Disponível em: <<https://docs.blender.org/manual/pt/2.79/index.html>>. Acesso em 03-julho-2017
- ARTEC STUDIO, Artec Studio Documentation. 2017. Disponível em: <<http://docs.artec-group.com/as/11/en/index.html>>. Acesso em 05-março-2017.
- PHP. Manual Oficial do PHP - Prefácio. 2017. Disponível em: <http://www.php.net/manual/pt_BR/preface.php/>. Acesso em: 09-outubro-2018.
- BOOTSTRAP. Bootstrap. 2017. Disponível em: <<http://getbootstrap.com/>>. Acesso em: 10-outubro-2018.
- JQUERY. Jquery. 2015. Disponível em: <<http://jquery.com/>>. Acesso em: 13-outubro-2018.
- CODEIGNITER, CodeIgniter User Guide. 2015. Disponível em: <https://codeigniter.com/user_guide/index.html>. Acesso em 20-novembro-2018.
- UNITY 3D, Unity User Manual. 2018. Disponível em: <<https://docs.unity3d.com/2018.3/Documentation/Manual/>>. Acesso em 05-maio-2018.